



HAL
open science

Approche organisationnelle basée sur le paradigme agent pour la synthèse et la réutilisation des connaissances en ingénierie collaborative

Hind Darwich Akoum

► To cite this version:

Hind Darwich Akoum. Approche organisationnelle basée sur le paradigme agent pour la synthèse et la réutilisation des connaissances en ingénierie collaborative. Gestion et management. Université de Lorraine, 2014. Français. NNT : 2014LORR0224 . tel-01751365

HAL Id: tel-01751365

<https://hal.univ-lorraine.fr/tel-01751365>

Submitted on 29 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

Ecole doctorale RP2E (Ressources, Procédés, Produits, Environnement)

Université de Lorraine

ERPI (Equipe de Recherche sur les Processus Innovatifs)

THESE PRESENTEE EN VUE DE L'OBTENTION DU GRADE DE

DOCTEUR

DE

L'UNIVERSITE DE LORRAINE

Spécialité : GENIE DES SYSTEMES INDUSTRIELS

PAR

Hind DARWICH AKOUM

Approche organisationnelle basée sur le paradigme agent pour la synthèse et la réutilisation des connaissances en ingénierie collaborative

Soutenue publiquement le 10 octobre 2014, devant le jury d'examen :

Abdelaziz BOURAS, Professeur des Universités, Université Lyon II, Rapporteur

Emmanuel CAILLAUD, Professeur des Universités, Université de Strasbourg, Rapporteur

Jean-Pierre BOUREY, Professeur des Universités, Ecole Centrale de Lille, Président

Patrick CHARPENTIER, Professeur des Universités, CRAN, Université de Lorraine, Examineur

Eric BONJOUR, Professeur des Universités, ERPI, Université de Lorraine, Directeur de thèse

Davy MONTICOLO, Maître de conférences, ERPI, Université de Lorraine, Co-directeur de thèse

Thierry BEAUJON, Dirigeant, TDC Software, Examineur invité

A ma Maman...

Remerciements

Le 19 octobre 2010, j'ai passé un entretien pour une embauche dans le cadre d'une thèse CIFRE avec la société TDC Software (<http://www.tdc.fr/>) et le laboratoire ERPI de l'Université de Lorraine. C'est à ce moment-là que j'ai tourné la page de ma vie étudiante pour acquérir, quelques mois plus tard, une première expérience dans le monde de recherche industrielle en France.

Ainsi, je tiens à exprimer tout d'abord mes sincères remerciements à M. Thierry Beaujon, directeur de TDC Software, pour m'avoir accordé cette opportunité et de m'avoir accueillie au sein de l'équipe de développement de sa société basée dans le Doubs. Je le remercie pour sa disponibilité, son écoute et ses avis qui ont permis de rythmer l'avancement de mon travail. Toute ma gratitude va également à M. Fabien Barret, M. Hakim Baradi, M. Yannick Blondeau et Mme. Mélanie Caredda, pour leur participation à ce travail, nos nombreux échanges, et pour leur soutien.

Ce travail de thèse a été effectué dans le cadre d'une collaboration de recherche avec l'équipe de recherche ERPI de l'Université de Lorraine. Je souhaite remercier Mme le Professeur Laure Morel de m'avoir accueillie au sein de son équipe, et pour le soutien du laboratoire au plan logistique et matériel qui ne fait jamais défaut aux doctorants.

Un grand merci à M. le Professeur Eric Bonjour pour avoir tenu le rôle de directeur de thèse, et aussi pour ses qualités pédagogiques, son expertise pluridisciplinaire dans les domaines liés au développement des produits et de la gestion des connaissances, ses encouragements continus, sa patience et son soutien humain.

J'adresse également mes grands remerciements à mon co-directeur de thèse M. Davy Monticolo pour sa précieuse participation à l'encadrement de ces travaux, ses conseils avisés, et pour avoir partagé son expertise sur les systèmes multi-agents et la gestion de connaissances.

Un très grand merci à M. Abdelaziz Bouras, professeur de l'Université Lyon II et M. Emmanuel Caillaud, professeur à l'Université de Strasbourg, pour m'avoir fait l'honneur de rapporter sur ce travail de thèse. J'adresse également mes remerciements à M. Jean-Pierre Bourey, professeur de l'école Centrale de Lille et M. Patrick Charpentier, professeur à l'Université de Lorraine qui ont accepté d'être examinateurs et membres du jury de thèse.

Je tiens à remercier tous mes collègues de l'équipe de développement de TDC Software, Christelle, Jorge, Aurélie, Emmanuel, Lydie, Sihem, Corinne, Mireille, Marie, pour leur sympathie et tous les moments mémorables qu'on a partagés ensemble, même en dehors des heures de travail (pauses midi, soirées à thèmes variés, jogging, randonnées, pique-niques, jeux, cinés, et j'en oublie). C'était vraiment un grand plaisir de travailler avec vous.

Je tiens à remercier également mes amis que j'ai rencontrés au sein du laboratoire ERPI : Nathalie, Francisco, Daniel, Giovanni, Barbara, Pierre, Julien, Elisa, Lamia, Alexis, Monica, Emmanuel, Simona, Papken. Merci pour avoir partagé les moments heureux et les incertitudes de doctorant.

Une pensée spéciale à ma famille et mes amis au Liban et en France. Je vous remercie d'être toujours à mes côtés pour me soutenir et d'avoir cru en moi.

Enfin, mes remerciements du plus profond de mon cœur à ma très chère maman, mon papa, mon mari, mon frère et mes deux sœurs, à qui je dédie ce manuscrit, pour votre amour inconditionnel et votre soutien extraordinaire. Merci d'être dans ma vie, tout simplement.

Table des matières

TABLE DES MATIERES	1
CHAPITRE 1. INTRODUCTION	4
1.1 MOTIVATION.....	4
1.2 CONTEXTE INDUSTRIEL	4
1.3 OBJECTIFS ET PREOCCUPATIONS DU TRAVAIL	5
1.3.1 <i>Formaliser les concepts liés aux traces d'activités et aux processus opérationnels.....</i>	<i>6</i>
1.3.2 <i>Définir un système multi-agents basée sur une approche organisationnelle pour assister les collaborateurs dans l'application de leurs démarches métier</i>	<i>7</i>
1.3.3 <i>Construire les processus opérationnels en fonction des rôles des acteurs métier</i>	<i>8</i>
1.4 PLAN DU DOCUMENT	9
CHAPITRE 2. ETAT DE L'ART	10
2.1 PROCESSUS DE CONCEPTION DE PRODUITS.....	10
2.1.1 <i>Définitions et typologie de la conception.....</i>	<i>10</i>
2.1.2 <i>Les principales caractéristiques de l'activité de conception.....</i>	<i>12</i>
2.1.3 <i>Quelques grands thèmes de recherche en conception.....</i>	<i>13</i>
2.2 DEFINITION ET CARACTERISTIQUES DE CONCEPTS CLES.....	14
2.2.1 <i>Donnée.....</i>	<i>14</i>
2.2.2 <i>Information</i>	<i>14</i>
2.2.3 <i>Connaissance</i>	<i>14</i>
2.2.4 <i>Compétence</i>	<i>15</i>
2.2.5 <i>Processus et plan d'actions</i>	<i>16</i>
2.3 GESTION DE CONNAISSANCES ET MEMOIRE DE PROJETS.....	16
2.3.1 <i>Des thèmes traités en gestion des connaissances</i>	<i>16</i>
2.3.2 <i>Modèles du produit, des processus et des décisions en conception.....</i>	<i>18</i>
2.3.3 <i>Mémoires organisationnelles.....</i>	<i>23</i>
2.4 ONTOLOGIES	25
2.4.1 <i>L'ontologie organisationnelle</i>	<i>26</i>
2.4.2 <i>L'ontologie Failure Analysis/ Failure Identification (FA/FI)</i>	<i>27</i>
2.4.3 <i>Le cadre d'ontologie organisationnelle pour le SBPM (Semantic Business Process Management).....</i>	<i>28</i>
2.5 TRACES D'ACTIVITE ET SYSTEMES A BASE DE TRACES.....	30
2.5.1 <i>Définition d'une trace</i>	<i>30</i>
2.5.2 <i>Systèmes à base des traces.....</i>	<i>31</i>
2.5.3 <i>Ontologies dans les systèmes à base des traces</i>	<i>32</i>
2.6 SYSTEMES MULTI-AGENTS	35
2.6.1 <i>Le paradigme Agent.....</i>	<i>35</i>
2.6.2 <i>Quelques systèmes multi agents existants</i>	<i>38</i>
2.7 MESURES DE SIMILARITE ENTRE PROCESSUS	43
2.8 SYNTHESE DE L'ETAT DE L'ART ET PROBLEMATIQUES	45

CHAPITRE 3. VERS UNE APPROCHE DE CAPITALISATION BASEE SUR LES DEMARCHES METIER.....	47
3.1 VERS UN DOUBLE CYCLE DE CAPITALISATION DES CONNAISSANCES	48
3.1.1 <i>Le cycle de gestion des connaissances</i>	48
3.1.2 <i>Proposition d'un double cycle de capitalisation des connaissances</i>	49
3.2 APPROCHE GLOBALE DE GESTION DES DEMARCHES METIERS	52
3.3 ONTOLOGIE DU DOMAINE DES DEMARCHES METIER ET DES PROCESSUS OPERATIONNELS : ONTOPROCESS.....	54
3.3.1 <i>Etape de construction de l'ontologie OntoProcess</i>	54
3.3.2 <i>Modèle conceptuel correspondant à OntoProcess</i>	55
3.3.3 <i>Extension d'OntoProcess relative aux activités et entités de conception spécifiques à un domaine</i> 61	
3.4 CONCLUSION	69
CHAPITRE 4. ARCHITECTURE DU SYSTEME MULTI-AGENTS MPPAS	70
4.1 INTRODUCTION	70
4.2 METHODOLOGIE DE CONCEPTION DU SYSTEME MULTI-AGENTS	71
4.2.1 <i>Méthodologies pour la conception de SMA existantes</i>	71
4.2.2 <i>Notre approche de modélisation organisationnelle et de construction des cartes d'identité des agents</i> 77	
4.3 ARCHITECTURE GLOBALE DU SYSTEME.....	79
4.3.1 <i>Identification des organisations d'agents</i>	79
4.3.2 <i>Niveau individuel</i>	80
4.3.3 <i>Synthèse et comparaison des propriétés des rôles du système multi-agents</i>	86
4.3.4 <i>Niveau organisationnel et communication</i>	89
4.4 CONCLUSION	94
CHAPITRE 5. SYSTEME A BASE DE TRACES POUR LA CONSTRUCTION DE PROCESSUS OPERATIONNEL	96
5.1 EXIGENCES SUR LA CONSTRUCTION DU PROCESSUS OPERATIONNEL	96
5.1.1 <i>Missions et finalités du SBT</i>	96
5.1.2 <i>Scénario d'utilisation lié à la demande de visualisation du processus opérationnel</i>	97
5.1.3 <i>Cas test 1</i>	98
5.1.4 <i>Cas test 2</i>	99
5.1.5 <i>Cas test 3</i>	100
5.1.6 <i>Cas test 4</i>	100
5.2 ARCHITECTURE DU SBT POUR LA CONSTRUCTION DU PROCESSUS OPERATIONNEL	101
5.3 LE MODULE DE COLLECTE DES TRACES COLLABORATIVES DE CONCEPTION	103
5.3.1 <i>Modèle des traces</i>	104
5.3.2 <i>Implémentation de l'extraction des traces XML</i>	109
5.3.3 <i>Implémentation de l'annotation des traces</i>	111
5.4 LE MODULE DE TRANSFORMATION	112
5.4.1 <i>Structure générale du module de transformation</i>	113
5.4.2 <i>Choix d'une représentation des processus en BPMN</i>	114
5.4.3 <i>Définition des règles de transformation des traces en éléments BPMN</i>	115
5.4.4 <i>Implémentation de la construction de processus opérationnel par le CPO</i>	117
5.5 MODULE DE VISUALISATION DU PROCESSUS OPERATIONNEL	118
5.6 CONCLUSION	120
CHAPITRE 6. SPECIFICATION D'INTERFACES ET VALIDATION SUR ETUDES DE CAS.....	121
6.1 INTERFACES UTILISATEURS ET MODULES LOGICIELS	121
6.1.1 <i>Interface de construction d'une démarche métier et de consultation du référentiel de démarches métier</i> 121	

6.1.2	<i>Interface enrichir les démarches métier</i>	122
6.1.3	<i>Interface consulter le processus opérationnel</i>	123
6.1.4	<i>Interface demander de l'assistance dans le processus opérationnel</i>	124
6.2	EXPERIMENTATIONS.....	125
6.2.1	<i>Résultats sur les quatre cas test</i>	125
6.2.2	<i>Etude de cas 1</i>	125
6.2.3	<i>Etude de cas 2</i>	128
6.2.4	<i>Etude de cas 3</i>	130
6.3	CONCLUSION	131
CHAPITRE 7.	CONCLUSIONS ET PERSPECTIVES	132
CHAPITRE 8.	REFERENCES BIBLIOGRAPHIQUES	136

Chapitre 1. Introduction

1.1 Motivation

Pour survivre dans les marchés ou améliorer leur rentabilité, les industriels cherchent sans cesse à développer des produits plus compétitifs en termes de qualité, coût, respect de l'environnement ou délai de mise sur le marché. Le niveau concurrentiel actuel impose à l'entreprise d'améliorer ses performances. Elle doit éviter les erreurs du passé et diminuer les temps de la réalisation de ses activités. Cette perspective requiert le développement de l'« entreprise apprenante » où les organisations réutilisent des connaissances acquises pour mieux réaliser leurs activités.

La gestion de connaissances est reconnue comme un facteur de compétitivité dans les entreprises. Nous nous intéressons à la capitalisation des expériences qui alimentent le savoir-faire méthodologiques des ingénieurs et plus particulièrement, lors des phases de conception. D'un projet à l'autre, les concepteurs s'appuient sur leurs expériences passées pour réaliser des activités relativement semblables, avec un enchaînement de ces activités relativement stable. Cependant, cette expérience est rarement formalisée, faute de temps et en raison de la difficulté pour le concepteur d'exprimer a posteriori sa façon de travailler.

La plupart des activités de conception sont instrumentées, c'est-à-dire supportées par des outils informatiques. Certains de ces outils autorisent la sauvegarde automatique des traces numériques des actions exécutées par les concepteurs. Nous proposons une approche pour la gestion de connaissances basée sur la formalisation, la capture et la réutilisation semi-automatisée des expériences liées à l'utilisation d'un environnement logiciel lors du travail des concepteurs.

1.2 Contexte industriel

Nos travaux de recherche sont réalisés au sein du laboratoire ERPI (Equipe de Recherche sur les Processus Innovatifs) de l'Université de Lorraine et en collaboration avec la société TDC Software dans le cadre d'une thèse CIFRE.

TDC Software est un éditeur français de logiciels dans le domaine des méthodologies de conception et de maîtrise des risques.

TDC Software fournit une suite méthodologique pour la conception, formée de cinq modules dédiés chacun à une méthode de conception :

- Le module "Need" permet de recueillir la voix du client, de réaliser l'analyse fonctionnelle du besoin et d'élaborer le cahier des charges. Il supporte aussi la méthode d'analyse de la valeur.
- Le module "Structure" est dédié à l'analyse fonctionnelle interne, il permet d'associer les fonctions avec les composants de la solution et ainsi d'élaborer la vue organique du système.
- Le module "FMEA produit" (*Failure Mode and Effects Analysis*) permet de prévenir les risques engendrés par l'utilisation du produit,
- Le module "FMEA process" entend prévenir les risques engendrés par la fabrication du produit,
- Le module "FTA" (*Fault Tree Analysis*) permet de modéliser les arbres de défaillances.

Ces modules sont interconnectés assurant ainsi la cohérence des données de conception. Les logiciels capturent les traces des évolutions de chaque entité d'un projet de conception grâce à l'enregistrement de toutes les modifications. Ces modifications peuvent concerner la création, l'évolution ou la suppression d'une entité, d'un attribut ou d'une valeur. Chaque modification est datée et attitrée à un acteur, en fonction de ses identifiants de connexion.

TDC souhaite améliorer la performance de ses utilisateurs par la mise en place d'une analyse et de l'exploitation des traces d'activité de conception pour mettre en évidence les bonnes pratiques de conception à travers ses outils.

Dans le cadre de nos travaux de recherche, nous proposons des modèles, méthodes et outils de formalisation et de réutilisation des connaissances méthodologiques enregistrées dans les traces. Cette problématique est relativement générique et peut concerner d'autres éditeurs de logiciels.

1.3 Objectifs et préoccupations du travail

La contribution majeure de cette thèse peut être résumée ainsi :

Proposer une approche de construction et réutilisation des processus de référence (appelés par la suite démarches métier) basée sur l'exploitation des traces d'activité en fonction des rôles des acteurs lors des projets de développement de produit.

Cette contribution répond à trois objectifs. Le premier objectif est la définition du vocabulaire et d'une sémantique pour identifier, capturer et exploiter les connaissances liées aux démarches métier utilisés lors des projets de conception de produit. Le second objectif est de concevoir un système intelligent, basée sur une approche organisationnelle, facilitant la construction et la réutilisation de démarche métiers. Le troisième objectif est de concevoir et implémenter la construction de processus opérationnels, réellement mis en œuvre, à partir des traces et en fonction des rôles des acteurs métier.

1.3.1 Formaliser les concepts liés aux traces d'activités et aux processus opérationnels

Au cours de leurs activités, les concepteurs utilisent des outils supportant les méthodes de conception où ils manipulent des entités différentes (fonctions, modèles liés aux produits, phases de vie, critères, etc.). Ces outils permettent la sauvegarde des évolutions de ces entités dans des traces d'activités.

Définition : Les traces d'activité sont des représentations explicites et persistantes des actions réalisées et des informations manipulées par les acteurs métier lorsqu'ils accomplissent leurs activités de conception.

Nous nous intéressons à l'analyse de ces traces afin de capturer les informations créées et utilisées par les concepteurs. Afin de transformer ces informations en connaissances, nous décrivons l'ensemble des concepts utilisés dans les traces (sous la forme d'un modèle de traces) et nous définissons leurs relations en construisant une ontologie du domaine.

Une ontologie est une abstraction d'un domaine en termes de concepts et relations, qui sont exprimés dans un langage de représentation des connaissances, permettant leur réutilisation et leur partage entre des systèmes ou des utilisateurs (Hadzic, Pompit, Dillon, & Chang, 2009). Les ontologies spécifient un vocabulaire associé à une sémantique utile à la compréhension d'un domaine d'informations.

Les ontologies permettent, par exemple, à un système intelligent de type système multi-agents, de déterminer les connaissances qu'il doit capitaliser et la manière de les annoter et archiver. De plus, l'ontologie fournit la possibilité d'inférer pour mieux réutiliser ces connaissances. D'autre part, l'utilisation d'une ontologie permet à des agents logiciels d'échanger des messages, de les interpréter et d'en comprendre le contenu. L'interprétation dépend de l'existence d'une compréhension partagée entre les agents. Dans ce cas, l'agent n'a pas besoin d'une vue complète des connaissances composant le monde où il est, mais simplement d'une vue restreinte, commune et partagée. Cette compréhension distribuée entre les agents doit également exister entre les agents et les acteurs métier. Ceci nécessite un certain nombre de concepts compréhensibles par tous. Une ontologie de domaine permet de définir ces concepts et leurs relations.

La construction des ontologies est un axe de recherche de l'ingénierie des connaissances. Cette discipline produit des méthodes et des techniques de recueil, d'analyse et de structuration des connaissances, des plates-formes de modélisation, ainsi que des représentations des connaissances (Charlet, 2003).

Dans le cadre de notre travail, une ontologie du domaine des processus utilisés lors des projets de conception spécifie la structuration des connaissances utilisées et créées par les acteurs métier. Elle fournit également un vocabulaire pour annoter, manipuler et inférer sur les informations issues des traces d'activité afin de construire des processus opérationnels.

Définition : *Un processus opérationnel est une vue représentant le déroulement construit des activités réalisées sur un projet particulier.*

1.3.2 Définir un système multi-agents basée sur une approche organisationnelle pour assister les collaborateurs dans l'application de leurs démarches métier

L'activité de conception est un environnement distribué où évoluent des acteurs métiers en travaillant de manière collaborative et coopérative.

Chaque acteur métier possède sa propre façon de réaliser les activités métier. Même si la plupart des activités réalisées sont similaires d'un acteur à un autre possédant le même rôle, il réside cependant une différence dans la manière d'enchaîner ces activités. Afin de comprendre la logique de conception utilisée par chaque acteur, nous proposons de comparer le processus opérationnel construit à partir des traces d'activité de l'acteur avec le processus de référence de l'entreprise que nous appelons démarche métier. Nous définissons une démarche métier ainsi :

Définition: *Une démarche métier est le processus de référence supportant les activités réalisées par les acteurs métier lors d'un projet de développement de produit.*

Dans la littérature, le terme « processus métier » est souvent utilisé sans faire la distinction entre ce qui est réellement mis en œuvre et ce qui est préconisé comme étant une bonne pratique. Dans notre mémoire, pour marquer cette distinction, nous utilisons les termes de « processus opérationnel » et de « démarche métier ».

Seul un expert (ou un groupe d'experts) est en mesure de créer une démarche métier. En effet, en créant une démarche métier, l'expert formalise et standardise, en se basant éventuellement sur les normes existant dans la littérature, les expériences communes issues de différents projets. La démarche métier doit être validée par consensus avec l'ensemble des acteurs métier.

Bien souvent, les sources d'information utilisées dans des activités métier sont hétérogènes et distribuées. Dans notre cas, le problème de l'hétérogénéité des sources ne se pose pas puisque nous faisons l'hypothèse que les acteurs métier utilisent le même outil de conception. Cependant, les activités sont distribuées. Pour supporter notre approche de gestion des démarches métier, nous proposons de définir un système multi-agents basée sur une approche organisationnelle.

Le paradigme Agent facilite la conception d'une architecture logicielle au-dessus d'un environnement informationnel distribué. Un agent logiciel possède la faculté de communiquer avec d'autres agents. De plus, il est en mesure de réaliser des tâches difficiles, d'accepter et de déduire des instructions ou des demandes associées à son activité ainsi que de prendre des décisions (Corkill, D. Lander, 1998). Un système multi agent peut ainsi résoudre des problèmes complexes en les décomposant et en les partageant entre l'ensemble

des agents. Il est également prédisposé à fournir une assistance proactive aux acteurs métier pour la réutilisation des connaissances.

Les agents sont utilisés dans le domaine de l'ingénierie des connaissances pour reproduire les capacités et échanges des humains et pour organiser les connaissances en facilitant le classement et la réutilisation des documents. Ils apportent l'approche cognitive et sociale en modélisant des comportements intelligents individuels et collectifs (Charlet, 2003). Dignum (Dignum, Dignum, & Meyer, 2004) et Van Elst (Van Elst, Dignum, & Abecker, 2004) ont introduit le concept d'Agent Mediated Knowledge Management (AMKM) dont l'objectif est d'associer les théories de l'ingénierie des connaissances avec les modèles basés sur les agents. Les auteurs expliquent que « les caractéristiques principales des agents (habilité sociale, autonomie et proactivité) peuvent pallier les inconvénients des approches centralisées de gestion des connaissances ». Ainsi, un agent dont la tâche est de suivre les connaissances utilisées par les humains dans leur travail quotidien devient : « un assistant personnel qui collabore avec l'utilisateur dans le même environnement » (Maes, 1994). Nous pouvons citer de nombreux exemples d'agents assistants pour la gestion des connaissances tels que Letizia de Lieberman (Lieberman, 1995), OntoBroker développé par Staab et Schnurr (Staab, Schnurr, Studer, & Sure, 2000) ou le projet AACC Enembreck et Barthès (Enembreck & Barthès, 2003). Des SMA ont également été développés pour supporter des processus d'entreprises. Ces SMA sont définis comme « des systèmes utilisant des méta-informations relatives au processus d'entreprise qui relient des modèles formels appropriés et des ontologies pour obtenir une compréhension partagée facilitant l'exécution de traitements de gestion des connaissances » (Abecker, A., & Van Elst, 2003). Nous pouvons citer l'exemple des travaux de Qinglin (Qinglin & Zhang, 2009) qui ont défini un SMA pour rendre les opérations d'industrialisation plus flexible, plus économique et efficace en énergie. Dans la même catégorie, le système MAS-PLs proposé par (Cirilo, Nunes, Kulesza, & Lucena, 2012) gère les processus de dérivations dans les lignes de production.

Le modèle Agent que nous proposons pour la gestion des démarches métier a deux objectifs :

- Découvrir, capitaliser et valider les connaissances enregistrées dans les traces d'activités en construisant des processus opérationnels;
- Fournir une assistance adaptée à chaque acteur en fonction de son rôle, en comparant le processus opérationnel en cours mis en œuvre par les acteurs métier avec la démarche métier utilisée.

1.3.3 Construire les processus opérationnels en fonction des rôles des acteurs métier

Parmi les fonctionnalités essentielles du système intelligent que nous proposons, la construction du processus opérationnel à partir de la collecte des traces brutes est essentielle.

Les traces peuvent être réutilisées à deux fins : l'analyse et l'assistance. (Settouti, Prié, Mille, & Marty, 2006a) proposent une architecture d'un système à base de traces (SBT) composée de trois systèmes interdépendants : système de collecte, de transformation et de visualisation.

Des SBT ont déjà été développés pour formaliser des expériences d'utilisateurs d'un système informatique (Champin, Prié, & Mille, 2003) des épisodes d'apprentissage (Reffay et al., 2008) ou des processus (M. H. Karray, 2012) (M.-H. Karray, Chebel-Morello, & Zerhouni, 2014).

Partant d'un modèle de traces, les traces brutes sont transformées en traces structurées et annotées, en fonction du rôle des acteurs métier et du module de l'outil de conception utilisé. Des règles de transformation permettent de faire le lien entre d'une part, les entités modifiées dans les traces par les actions de l'utilisateur du module et d'autre part, les tâches réalisées. L'exploitation des processus opérationnels ainsi construits va influencer le choix du format qui sera retenu pour formaliser les processus.

1.4 Plan du document

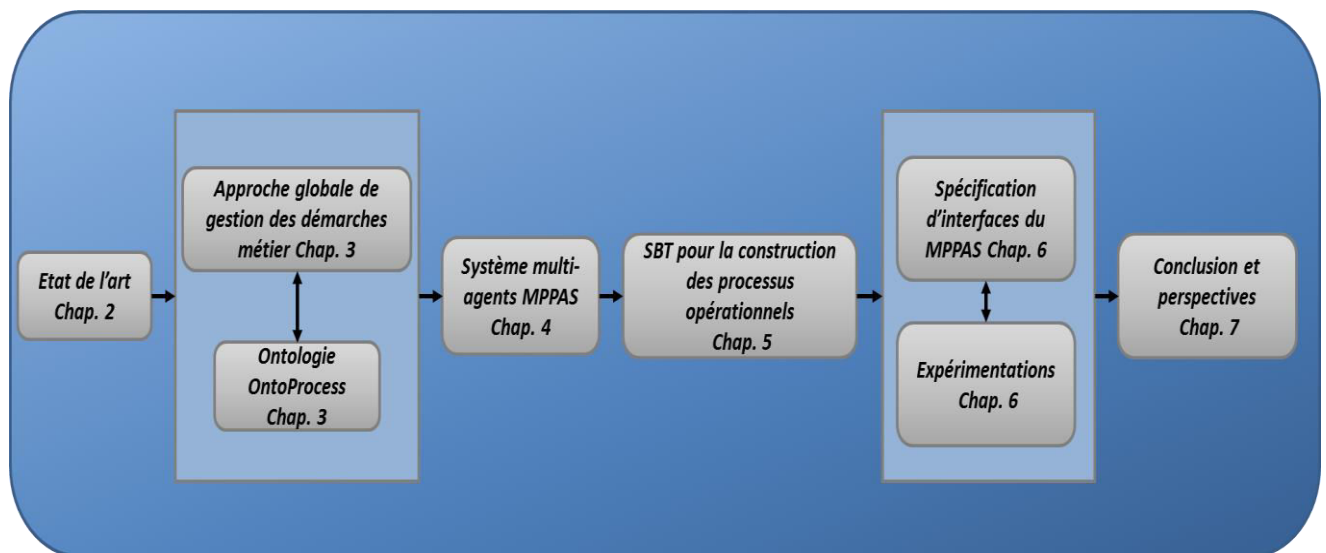


Figure 1. Plan du document

Le corps de ce mémoire est structuré en six chapitres (Figure 1) :

- Un état de l'art sur les techniques et méthodes de constitution de mémoires organisationnelles, de systèmes à base de traces et sur l'utilisation des ontologies et des systèmes multi-agents pour la capitalisation des connaissances,
- Une approche globale de gestion de démarches métier ainsi qu'une ontologie du domaine étudié,
- Une architecture d'un système intelligent de gestion des démarches métier, appelé MPPAS et basé sur un système multi-agent,
- Un Système à Base de Traces et des mécanismes de construction de processus opérationnel à partir de l'enregistrement de traces brutes,
- Des expérimentations pour validation partielle des propositions sur des cas test et sur deux études de cas,
- Une conclusion et des perspectives.

Chapitre 2. Etat de l'art

Le domaine de la conception a été exploré par de nombreux chercheurs pour développer de nouveaux outils et méthodes pouvant aider les concepteurs à améliorer leurs performances.

Ce premier chapitre vise à décrire le domaine de la gestion des connaissances en conception de produit et à positionner globalement l'objectif de notre travail. Nous présentons d'abord une définition et les caractéristiques du processus de conception. Notre état de l'art portera en particulier sur la constitution de mémoire de projets et sur la traçabilité des activités de conception. Nous terminons ce chapitre par un positionnement de notre problématique.

2.1 Processus de conception de produits

La conception de produits (de biens ou de services) est considérée comme une activité intellectuelle complexe. Elle peut être définie comme un processus de création et de définition d'un produit en réponse à un ensemble de besoins et d'exigences provenant de différentes parties prenantes.

Dans cette section, nous rappelons des définitions et une typologie de la conception. Ensuite nous synthétisons les caractéristiques essentielles de l'activité de conception.

2.1.1 Définitions et typologie de la conception

L'activité de conception est située en phase amont du cycle de vie du produit. Elle est constituée d'un ensemble de tâches interdépendantes : étude de faisabilité, modélisation fonctionnelle, recherche de solutions, modélisation géométrique, calcul et dimensionnement, optimisation tenant compte des contraintes de fabrication et d'assemblage...

La norme AFNOR X50-127 définit le processus de conception de la façon suivante : « partant des besoins exprimés, le processus de conception définit pas à pas le produit qui doit répondre aux besoins et aux attentes par des choix successifs portant sur des points de plus en plus détaillés ».

L'activité des concepteurs revient donc à transformer un besoin client, exprimé en termes de services et contraintes, en une description détaillée et suffisamment complète du produit. La description finale du produit est généralement représentée sous diverses formes : schématique, physique (maquette), graphique (plan), numérique ou simplement textuelle.

On distingue trois types de conception selon l'objectif, la disponibilité des connaissances au début du projet, la structuration du plan d'action et le degré d'autonomie et de créativité du concepteur (Perrin, 1999) :

- La conception routinière. Elle représente le cas où les connaissances à exploiter sont clairement identifiées et les objectifs sont globalement connus. Les plans d'action sont aussi globalement connus et l'autonomie créative du concepteur est limitée. Il s'agit surtout de choisir parmi les solutions existantes les meilleures pour répondre à de nouveaux problèmes dans le même contexte (produits semblables déjà réalisés).
- La conception innovante. Les connaissances et les objectifs de départ sont globalement connus. Ils correspondent généralement à un nouveau besoin exprimé par des clients. Par contre, les plans d'action ou les démarches du concepteur sont indéterminés. Le problème revient donc à définir les meilleurs plans d'action permettant d'arriver à ces objectifs tout en complétant la construction de ces objectifs.
- La conception créative. C'est le cas le plus complexe où ni les connaissances de départ ne sont disponibles, ni les plans d'actions ne sont déterminés au préalable. En d'autres termes, la conception créative naît d'une idée (mise en œuvre d'une invention ou d'une nouvelle technologie, par exemple) qui généralement fait l'objet d'une étude de faisabilité, permettant de décider du lancement d'un projet d'innovation. Les objectifs à atteindre, les fonctions du produit et le plan d'actions seront construits au fur et à mesure de l'avancement du projet. Les modélisations, simulations, prototypages et expérimentations permettront de développer de nouvelles connaissances.

Dans la réalité, la distinction entre ces trois types de conception n'est pas toujours évidente. Par exemple, un projet de conception automobile peut intégrer le résultat de sous-projets de conception de modules. A un sous-projet de conception innovante correspond alors une alternative moins risquée de conception routinière, qui revient à adapter une solution existante, pour limiter les risques d'échec du projet global.

Dans un contexte de réduction des temps de conception, les pratiques des industriels ont évolué d'une ingénierie séquentielle (Pahl & Beitz, 1996) où les activités de conception se déroulent de manière successive dans le temps, vers une ingénierie qualifiée de concurrente, de simultanée ou d'intégrée (Tichkiewitch, 1996). Ces visions de la démarche d'ingénierie sont enrichies par les méthodes et outils du domaine de travail collaboratif et se concrétisent par la mise en parallèle des activités de conception dans un cadre de travail collaboratif où les ressources et les acteurs de l'entreprise partagent leurs données. Dans leur ouvrage, (Pahl & Beitz, 1996) ont proposé un modèle théorique structurant le déroulement progressif du processus de conception.

2.1.2 Les principales caractéristiques de l'activité de conception

Les caractéristiques de l'activité de conception ont été mises en évidence par plusieurs chercheurs (Belkadi, 2006), (Lhote, Chazelet, & Dulmet, 1999), (Micaëlli & Forest, 2003), (Darses, Détienne, & Visser, 2001)... Nous pouvons retenir les points suivants :

- La conception est une activité créative et complexe. Partant d'un existant jugé insatisfaisant et considéré comme un problème, il s'agit, en un délai limité, d'imaginer, de réaliser et de valider une solution satisfaisant le mieux possible les différentes parties prenantes.
- La conception est une activité instrumentée, qui repose sur différentes représentations (modèles, schémas, maquettes, prototypes ...) d'objet n'existant pas encore et à venir.
- La conception repose sur un processus informationnel de caractère progressif, itératif et interactif. Le système à concevoir est développé, c'est-à-dire imaginé, réalisé, affiné, etc., pas à pas. Le concepteur peut répéter plusieurs fois certaines tâches pour affiner la spécification ou la définition d'une solution. La conception est un processus au cours duquel il y a en permanence interactions entre les acteurs et productions d'objets intermédiaires. Le rôle de ces objets intermédiaires comme le montre (Blanco, 1998) est principalement de garder une trace au cours du projet de conception et de servir comme moyen de compréhension entre les différents acteurs.
- La conception est un processus de décisions. Elle nécessite de multiples prises de décisions lors des différentes tâches, et des choix et arbitrages à toutes les phases. La plupart du temps ces décisions sont critiques, car à la fois incertaines, coûteuses ou faiblement réversibles.
- La conception est une activité cognitive, requérant, manipulant et générant des connaissances. Celles-ci portent autant sur l'objet conçu (connaissances techniques) que sur le processus de conception, son organisation ou son pilotage.
- Les activités de conception sont organisées. La conception de systèmes complexes requiert une structuration des processus permettant de satisfaire les attentes des clients et une mobilisation organisée de nombreux acteurs appartenant à des Métiers ou des départements différents.
- Les activités de conception sont gérées. La gestion de projet, au niveau opérationnel, permet de planifier l'avancement des activités, de faire le suivi des performances (mesure des écarts entre résultats réels et objectifs prévisionnels), de valider les résultats par des revues de projet et de prendre des mesures correctives en cas de dérives.
- La conception est une activité qui mobilise et développe des compétences. Les managers doivent assurer l'affectation adéquate des concepteurs aux tâches de conception et la gestion opérationnelle des projets. Ils doivent aussi concevoir et piloter un pro-

cessus de développement des compétences assurant la compétitivité future de l'entreprise.

Une conséquence de la complexité de l'activité de conception est que cette activité est rarement individuelle mais présente un caractère collectif avec des interdépendances souvent fortes entre des acteurs qui proviennent d'univers disciplinaires différents et la nécessité d'intégrer de multiples points de vue (cognitif, technique, social, économique, organisationnel, ...).

Répondre à cette complexité nécessite le développement de méthodes et d'outils afin d'aider les concepteurs à améliorer les performances de leurs activités.

2.1.3 Quelques grands thèmes de recherche en conception

Parmi les thèmes de recherche en conception, des travaux sont dédiés à la théorie de la conception. Ils concernent l'analyse de l'activité de conception afin d'interpréter sa complexité et de proposer des modèles théoriques facilitant sa compréhension. Nous pouvons mentionner dans ce thème des travaux tels que la théorie C-K de (Hatchuel, 2002) qui considère le raisonnement de conception comme une co-évolution par interactions d'un espace de concepts et d'un espace de connaissances, l'approche systématique de conception développée dans (Pahl & Beitz, 1996), l'approche artificialiste de la conception (Micaëlli & Forest, 2003)...

De nombreux travaux s'intéressent au développement d'outils supportant les activités de conception de produit (et de ses systèmes associés : système de production, système de maintenance ...). Différentes catégories d'outils de représentations du produit et de manipulation des données peuvent être identifiées :

- des supports techniques tels que les outils de CAO, de calcul et de simulation bien spécifiques à chaque domaine (Hadj-Hamou, Aldanondo, Lamothe, & Caillaud, 2002), des systèmes de gestion des données techniques (SGDT), etc.,
- des outils d'aide à la décision technique avec des méthodes d'optimisation et d'analyse multi-critères (Yannou & Bonjour (sous la direction de), 2006), et
- des supports aux flux d'informations en conception collaborative, de type collectifs, workflows ou CSCW (Boujut & Laureillard, 2002), (W. Shen, Yong, Yang, Barthes, & Luo (editors), 2008) (W. Shen et al., 2008),
- des supports à la mise en œuvre de méthodologies de conception et constituant une base de données partagées dans les projets.

La suite méthodologique TDC peut être placée dans les deux dernières catégories.

2.2 Définition et caractéristiques de concepts clés

Nous allons d'abord rapidement définir et caractériser des concepts clés de notre travail de thèse, à savoir : donnée, information, connaissance, compétence, processus et plan d'actions.

2.2.1 Donnée

Selon (Prax, 2000), une donnée est : « un fait discret et objectif ; elle résulte d'une acquisition, d'une mesure effectuée par un instrument naturel ou construite par l'homme. Elle peut être qualitative ou quantitative. Il n'y a normalement pas d'intention ni de projet dans la donnée, c'est ce qui lui confère son caractère d'objectivité ». Redman (Redman, 1992) propose une autre définition de données : "Un élément de donnée (datum or data item) est un triplet <entité, attribut, valeur>. Des données (data) sont alors une collection de tels triplets ».

2.2.2 Information

En ce qui nous concerne, nous considérons que donnée et information sont de même nature. Nous retenons que la principale distinction entre les deux concepts est que la donnée est une représentation sous forme brute d'une capture de renseignements (une valeur attachée à un attribut) portant sur quelque chose (une entité), tandis que l'information est une structuration d'un ensemble de données permettant de leur associer une sémantique. Nous parlerons de bases de données comme étant des systèmes informatiques qui manipulent, structurent, gèrent, traitent et partagent des données ou des informations d'une organisation.

2.2.3 Connaissance

Il est délicat de vouloir donner une définition au concept de connaissance. Cependant, en génie industriel, une caractéristique clé de la connaissance fait aujourd'hui consensus : une connaissance n'existe pas en dehors d'un individu, elle est de l'ordre d'une représentation cognitive, elle est dépendante d'une activité cognitive humaine. Selon (Prax, 2000), « la connaissance résulte d'une acquisition d'information et d'une action, elle est à la fois mémoire et processus de construction d'une représentation ». C'est une combinaison de sens du contexte, de mémoire personnelle et de processus cognitifs. Cette définition rejoint celle proposée dans (Ermine, Chaillot, Bigeon, Charreton, & Malavieille, 1996) où la connaissance est perçue comme de « l'information qui prend une certaine signification dans un contexte donné » et qui est traitée pour pouvoir être réutilisée.

Si une information est un ensemble de données auquel est associé une sémantique, la connaissance, quant à elle, est associée à une structure cognitive permettant d'interpréter un ensemble d'informations afin de mener un raisonnement dans une situation particulière (ou un contexte d'utilisation) et pour une finalité déclarée (activité finalisée, résolution de problème, prise de décision...).

En 1995, Nonaka et Takeuchi publient un livre sur la formation des connaissances et son utilisation dans les entreprises japonaises (Nonaka & Takeuchi, 1995). Pour eux, la connaissance est une croyance vraie et justifiée par le contexte. Les auteurs proposent un modèle de création et de transfert des connaissances comportant quatre modes :

- La combinaison qui est le processus de création de connaissances explicites à partir de la restructuration d'un ensemble de connaissances explicites déjà acquises.
- L'internalisation qui est le processus de conversion de connaissances explicites en connaissances tacites à l'aide de processus d'apprentissage.
- La socialisation qui est le processus de transmission de connaissances tacites, par des échanges verbaux, par de l'observation, par l'imitation et surtout par la pratique.
- L'externalisation : qui est le processus qui permet le passage de connaissances tacites en connaissances explicites, sous la forme de concepts, modèles ou hypothèses.

Ce modèle fait intervenir deux catégories de connaissances : les connaissances explicites (ou explicitables) sont les connaissances pouvant être facilement extraites et représentées sur des supports partageables. A l'inverse, les connaissances tacites sont difficilement explicitables voire impossibles à expliciter.

Nous ne rentrons pas dans le détail de ces typologies qui posent certaines interrogations sur les limites entre connaissance tacite et explicite, sur les limites entre connaissance et savoir (Bonjour, Dulmet, & Lhote, 2002). Nous retenons par contre que la connaissance est « stockée » dans la mémoire de l'individu et doit être reconstruite à chaque réutilisation. Tous les autres supports de stockage sont des systèmes "à base de connaissances". Ils ne peuvent en aucun cas comporter des connaissances (au sens propre du mot) mais ils manipulent une forme particulière de données descriptives d'une connaissance explicitée.

2.2.4 Compétence

La compétence est un concept plus global qui met en œuvre l'ensemble des précédents concepts. La définition de la compétence a souvent été associée aux notions de connaissance, de savoir et de savoir-faire (Prax, 2000). Le Boterf (Le Boterf, 2000) définit une personne compétente comme « une personne qui sait agir avec pertinence dans un contexte particulier, en choisissant et en mobilisant un double équipement : ressources personnelles (connaissances, qualités, culture, émotions ...) et ressources de réseau (bases de données, documents, réseaux d'expertise, ...) ». Pour cet auteur, la compétence est synonyme d'un savoir agir reconnu (Le Boterf, 2000). La définition souvent admise par les industriels est celle qui a été énoncée par le mouvement des entreprises de France en 2002 comme étant : « une combinaison de connaissances, savoir-faire, expériences et comportements s'exerçant dans un contexte précis. Elle se constate lors de sa mise en œuvre en situation professionnelle à partir de laquelle elle est validable ».

Une compétence n'existe pas au préalable mais elle est toujours construite, mise en œuvre et développée au cours d'une action finalisée qui peut être : la réalisation d'une tâche (ou d'une mission), la résolution d'un problème particulier, une initiative personnelle dont le résultat produit est reconnu par l'entreprise ...

En ce qui nous concerne, nous retenons la définition proposée dans (Bonjour et al., 2002) qui met l'accent sur la finalité, la contextualisation et sur la performance associée à la compétence en terme de valeur pour l'entreprise. Cette définition peut s'appliquer aussi bien à un acteur individuel ou collectif : « La compétence est la mobilisation d'un ensemble de savoirs hétérogènes aboutissant à la production d'une performance reconnue, par rapport à un environnement donné et dans le cadre d'une activité finalisée. »

2.2.5 Processus et plan d'actions

D'après la norme ISO 9000-2000, un processus est un système d'activités qui utilise des ressources pour transformer des éléments entrants en éléments de sortie. Cette définition considère l'activité comme un élément constitutif du processus.

La norme AFNOR a adopté la définition suivante de l'activité : « une transformation d'un flux de matières et/ou d'informations sous l'influence d'un ensemble de contrôles et sous réserve de disponibilité de ressources nécessaires et de temps. Cette transformation peut être décrite par une combinaison d'opérations ».

Un plan d'actions qui est utilisé à la fois en psychologie cognitive et en management est pertinent pour décrire une activité humaine. Il est défini dans (Bonjour, 2008) comme étant « l'organisation préalable retenue par un acteur, pour mener à bien un projet (ou une mission) qui lui a été confié ». Cette organisation concerne à la fois la décomposition des objectifs de la mission en un agencement de sous-objectifs et l'attribution de ressources pour réaliser ces sous-objectifs.

La section suivante présente un état de l'art sur la gestion des connaissances. Celui-ci est orienté vers les modèles et méthodes de constitution de mémoires de projets.

2.3 Gestion de connaissances et mémoire de projets

La traçabilité et la capitalisation des connaissances en conception offrent aux concepteurs et aux chefs de projets la possibilité de structurer puis de réutiliser les informations et les connaissances passées et présentes afin de résoudre ultérieurement de nouveaux problèmes d'ordre technique (sur le produit) ou managérial (sur l'organisation de conception).

2.3.1 Des thèmes traités en gestion des connaissances

La gestion des connaissances est un domaine de recherche à part entière (Schreiber, 1992; Studer, Benjamins, & Fensel, 1998). Il ne s'agit pas d'une simple opération de stockage et d'archivage de données et d'informations mais plutôt d'une acquisition intelligente de différentes connaissances relatives à différents projets et acteurs.

Le Knowledge Management (KM) est une discipline qui comprend des processus et des techniques pour la création, la collecte, l'indexage, l'organisation, la distribution, l'accès et l'évaluation à des connaissances institutionnelles pour améliorer les performances, et plus généralement pour l'exploitation du capital intellectuel, incluant les opportunités de réutilisation (Tsui, Garner, & Staab, 2000).

Les systèmes à base de connaissances (SBC – Knowledge-Based Systems KBS) ont pour objectif de fournir aux utilisateurs des accès aux connaissances et à des raisonnements spécifiques à leur domaine d'expertise.

Un des objectifs des SBC est de partager des connaissances expertes et de faciliter leur réutilisation et l'apprentissage de nouvelles connaissances. Cela requiert la mise en œuvre de processus relatifs à l'acquisition, la formalisation, la découverte, l'intégration et l'évaluation des connaissances (Lee & Chen, 2012a). D'un point de vue technique, ces systèmes sont constitués d'une base de connaissances, d'un moteur d'inférence et d'une interface utilisateur (Lai, Huang, & Wang, 2011).

Une étude récente (Lee & Chen, 2012a) sur les tendances en KM de 1995 à 2010 a souligné que (a) les recherches sur les mémoires organisationnelles et sur les modèles de KM sont passées d'un domaine sous-exploré en 1995 à un domaine de recherche clé en 2010 rencontrant ainsi un intérêt croissant ces vingt dernières années et (b) la réutilisation des connaissances est considérée en 2010 comme un domaine de recherche important mais sous-exploré et encore immature.

Une mémoire organisationnelle (aussi appelée mémoire d'entreprise) vise à améliorer la prise de décisions et s'intéresse à toute forme de connaissances, utilisées par des personnes dans leurs pratiques professionnelles, que ces connaissances soient formalisées dans un système d'information ou qu'elles soient tacites (Tsui et al., 2000). Ce type de recherche s'intéresse fortement à la formalisation et au partage des connaissances.

La section 2.3.2 reviendra sur des modèles du produit, du processus de conception et de traçabilité des décisions qui ont été développés entre 1995 et 2005, dans le but de constituer des mémoires organisationnelles ainsi que sur des travaux portant sur le développement de mémoires de projets.

Les représentations basées sur les ontologies ont été largement étudiées entre 1995 et 2000 (Lee & Chen, 2012a) et sont maintenant fréquemment utilisés comme formalismes des connaissances du domaine étudié. Elles seront présentées en section 2.4.

D'une façon générale, les techniques d'Intelligence Artificielle ont joué un rôle clé pour améliorer la formalisation et la réutilisation des connaissances (Tsui et al., 2000), (Weiming Shen, 2003), (Mikos, Ferreira, Botura, & Freitas, 2011).

Certaines techniques d'IA sont adaptées aux approches de résolution de problèmes (par exemple, le cycle du raisonnement à partir de cas, RàPC ou Case-Based Reasoning cycle (Bergmann, Kolodner, & Plaza, 2005)), à la prise de décision distribuée (par exemple, les systèmes multi-agents ou Multi-agents Systems (Enric Plaza, 2005)), ou à la découverte de connaissances dans de grandes bases de données (par exemple, les techniques de fouilles de données ou de découverte de règles (Tsai, 2013) (Liu & Harding, 2008)).

La tendance actuelle est d'associer plusieurs techniques d'IA pour mieux couvrir et traiter l'ensemble du cycle des connaissances et mieux répondre aux divers challenges sur la gestion des connaissances.

D'un point de vue pratique, la réutilisation efficace des expériences et l'apprentissage de nouvelles leçons (lessons-learned) sont des atouts de plus en plus importants pour les entreprises et représentent de réelles sources d'avantages compétitifs dans différents domaines comme la conception des produits, la maintenance, le diagnostic, la gestion des risques, la gestion de qualité, la gestion de projets ...

D'un point de vue scientifique, des modèles et des techniques ont été élaborés pour faciliter la contextualisation et la réutilisation des connaissances comme les techniques à base de cas ou à base de traces, et les systèmes de retours d'expériences ou de leçons apprises (Bergmann et al., 2005) (Weber, Aha, & Becerra-Fernandez, 2001) (M.-H. Karray et al., 2014; Davy Monticolo, Mihaita, Darwich, & Hilaire, 2014; Potes Ruiz, Kamsu Foguem, & Grabot, 2014). Un numéro spécial de la revue Knowledge-Based Systems, intitulé *Enhancing experience reuse and lesson learning* paru en 2014 dresse un état de l'art des travaux dans ce domaine.

Il existe plusieurs méthodes et outils de gestion de la connaissance qu'on peut classifier en deux catégories (Smain Bekhti & Matta, 2003) :

- Méthodes de capitalisation de connaissances : elles utilisent des techniques d'extraction (interview d'experts, collecte des documents) et de modélisation des connaissances. On trouve des méthodes bien connues comme REX, MKSM ou KADS ainsi que des approches d'élaboration de mémoire de projet.
- Méthodes d'extraction directe : elles visent à recueillir l'information directement à partir des traces de l'activité. On distingue des techniques de fouille de données (comme le data mining, text mining ou rule mining) et des techniques de traçabilité qui s'appuient sur des systèmes à base de traces.

Chaque type d'approche possède ses avantages et ses inconvénients. Le premier type d'approches permet d'obtenir de la connaissance riche en sémantique mais cela nécessite d'obtenir l'adhésion et la participation d'experts pour formaliser la connaissance et souvent, d'un spécialiste de la connaissance (cogniticien) pour aider à recueillir l'information ou à expliciter la connaissance. Dans le deuxième type d'approches, le processus de recueil et de capitalisation est assez transparent mais, souvent, on obtient des connaissances incomplètes ou pauvres au niveau sémantique.

Dans la section suivante, nous revenons sur les travaux réalisés dans les années 1995-2005 environ portant sur les modèles de produit, processus et décisions supportant des mémoires organisationnelles.

2.3.2 Modèles du produit, des processus et des décisions en conception

En conception, nous pouvons identifier trois grandes catégories de modèles pouvant supporter des mémoires organisationnelles. Ces catégories sont relatives à des flux informationnels relatifs :

- au produit qui permettent de le définir selon différentes vues : fonctionnel, structurel, comportemental, géométrique, dysfonctionnel...
- au processus qui concernent la description des tâches, de l'organisation des processus et des contraintes qui régissent la réalisation,
- aux décisions qui sont produites et discutées au cours du projet sur tous les niveaux : opérationnel, tactique et stratégique.

Toutefois, d'autres catégories de modèles sont manipulées par les concepteurs, en particulier, pour décrire les systèmes associés au produit au cours de son cycle de vie : modèles logistiques et d'assemblage, modèles réalisationnels, modèles de démantèlement ... Nous limiterons notre analyse aux trois catégories précédentes d'informations.

Notre état de l'art se limitera à présenter les concepts de base de certains modèles du produit et du processus dédiés à la structuration des connaissances en conception.

2.3.2.1 Modèles du produit

Au niveau des méthodes spécifiques dédiées à la traçabilité des activités de conception, les premiers travaux se sont intéressés à la modélisation du produit, objet principal du processus de conception.

Les modèles du produit ne se limitent pas à une description physique ou organique (géométrique, structurelle, ...) mais visent à prendre en compte aussi des aspects fonctionnels

et comportementaux. Ces différents aspects forment l'espace de conception dans lequel se construit progressivement la solution technique.

Dans le cadre du projet DEKLARE, Saucier (Saucier, 1997) a proposé un modèle reliant les aspects fonctionnels aux aspects physiques des solutions techniques en intégrant une représentation de la propagation des contraintes entre les deux vues. D'autres modèles ont été proposés en génie industriel et en génie mécanique comme le modèle entité-fonction (Eynard, 1999).

Harani (Harani, 1997) a introduit la notion de point de vue pour prendre en compte d'un côté, les différentes représentations qu'utilisent les concepteurs sur le produit (fonctionnel, comportemental et structurel) et d'un autre côté le domaine technologique tel que le point de vue mécanique, électrique, etc. Le modèle de Harani a été développé avec l'outil Merise dans une optique de conception d'un système d'information. Tichkiewitch (Tichkiewitch, 1996) propose un modèle produit pour la structuration des données de chacun des métiers intervenant dans la conception du produit et la classification des données produit selon leur utilité pour les métiers identifiés.

2.3.2.2 Modèles du processus

La modélisation du processus de conception vient compléter celle du produit. Elle consiste à spécifier et à guider l'évolution des états des modèles du produit le long du projet pour aboutir à un produit spécifique. Un modèle du processus de conception comporte une description des différentes tâches à réaliser pour mener à bien un projet de conception. La modélisation des processus est au cœur de toute modélisation d'entreprise. Elle n'est pas une fin en soi, mais présente un intérêt particulier pour bien maîtriser l'évolution des processus, afin de piloter leur aboutissement, ou aussi pour sauvegarder un historique pour des fins de réutilisation.

La plupart de modèles de conception présentent leur démarche comme un enchaînement des tâches. D'après l'analyse faite par (Scaravetti, 2004) sur différents processus de conception dans la plupart de processus il existe un découpage en phases. Ce découpage répond au besoin de segmenter le temps et de prévoir des étapes de validation.

Des travaux ont porté sur la dualité entre modèles de processus et modèles de produit. Vargas (Vargas, 1995) a proposé un modèle processus, complété par la suite par le modèle produit de Saucier dans le projet DEKLARE. Ce modèle est une représentation sous forme arborescente du processus de conception. Il est composé d'un ensemble de tâches (problèmes à résoudre), d'un ensemble de méthodes (manières de résoudre des tâches) et d'un ensemble de méthodes élémentaires (calculs et dialogues avec l'utilisateur). Les tâches et les méthodes sont organisées sous la forme d'un graphe ET/OU. Chaque tâche est réalisée par plusieurs méthodes et chaque méthode peut être dédiée à une ou plusieurs tâches. Le raisonnement du concepteur est modélisé sous forme de stratégies de résolution par l'intermédiaire de choix successifs (de tâches et de méthodes) faites par le concepteur durant le processus de conception.

Un modèle de processus de conception décrit l'enchaînement des tâches à suivre pour mener à bien un projet de conception dans un domaine d'application donné (Harani, 1997). L'objectif de cette modélisation est d'appuyer l'analyse de l'activité et de permettre la capitalisation des connaissances liées à son exécution. Dans la perspective de conception de systèmes d'information, Harani (Harani, 1997) a complété son modèle "multipoint de vues"

du produit par un modèle du processus (Figure 2). Les informations du modèle sont structurées selon quatre types de concepts :

- Le concept de processus représente l'enchaînement de tâches de conception mettant en évidence les étapes clés de la conception, il permet de faire le lien entre les tâches de conception et le produit concerné par ces tâches.
- Le concept de tâche modélise une étape du processus de conception, une tâche peut être élémentaire ou décomposable en un ensemble de sous-tâches.
- Le concept de ressource représente l'ensemble des intervenants dans la réalisation des tâches, cela peut être des méthodes de calcul, des logiciels, etc.
- Le concept d'état permet de visualiser l'évolution des différentes étapes du processus, de ses tâches et du produit, en conservant l'historique de tous les détails sur ces concepts.

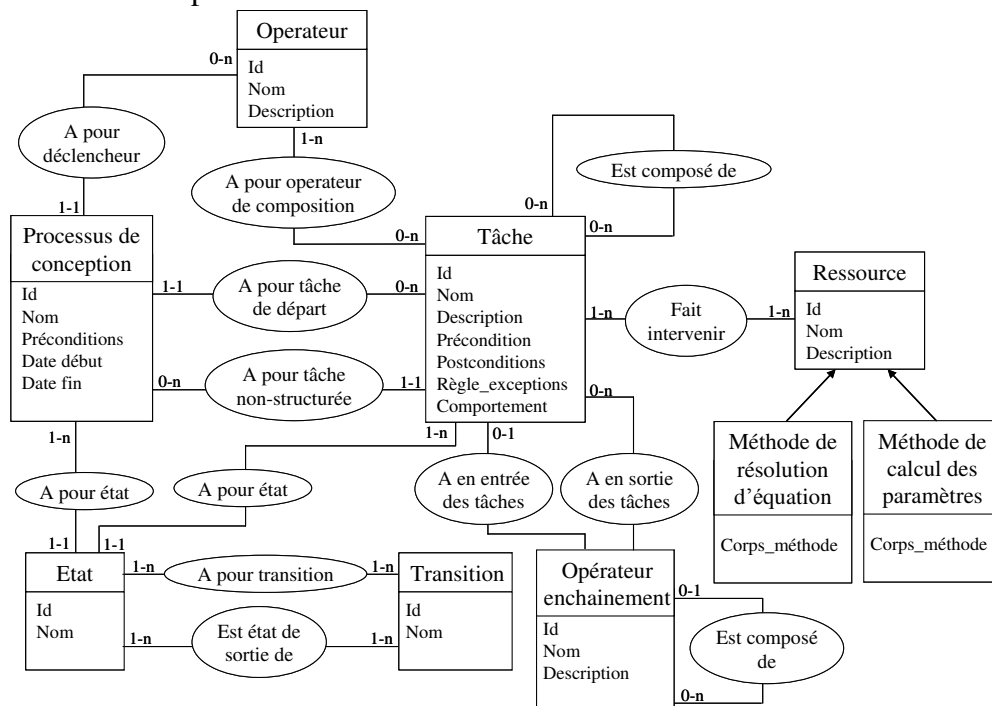


Figure 2. Exemple de modèle processus (Harani, 1997)

Dans la même perspective, Menand (Menand, 2002) a utilisé le formalisme UML pour définir le méta-modèle "MULTI" du processus de conception. Ce méta-modèle est composé de méta-classes concernant le domaine et le projet sur lequel porte le processus de conception. Il offre une sémantique qui permet de capturer les informations relatives au processus et les flux d'information entre acteurs à travers les deux sous-modèles "domaine" et "projet" ainsi que les éléments partagés avec le modèle produit. Dans ce modèle, le processus est composé d'un ensemble de tâches associées à des conditions d'ordonnement et d'exécution.

D'une façon plus générale, la modélisation des processus d'entreprise peut s'appuyer sur différents formalismes, notations ou langages (Weske, 2012), comme les réseaux de Petri, les diagrammes d'activités d'UML, le formalisme des graphes, SPEM (Software Process Engineering Meta-Model), BPMN (Business Process Model and Notation), XPDL (XML Process Definition Language), IDEF (ICAM DEFinition for Function Modeling) ou encore EPC (Event-driven Process Chain).

Différents auteurs ont présenté des options : les réseaux de petri (Heer & Würzberger, 2011), les flux de travaux ou *workflows* (van der Aalst, Ter hofstede, Kiepuszewski, & Barros, 2003). En effet, des implémentations pour représenter les processus de conception (Heer & Würzberger, 2011) ont été proposés. Cependant, le langage de modélisation BPMN (Business Process Model and Notation) est devenu le standard de-facto pour représenter de façon graphique les processus métier (Chinosi & Trombetta, 2012).

Un langage de modélisation (tel que BPMN) est un système de signes et de règles permettant d'exprimer une pensée et de communiquer. Dans BPMN, chaque signe comporte un symbole et une signification. Un résumé des éléments du langage BPMN est présenté dans la Figure 3.

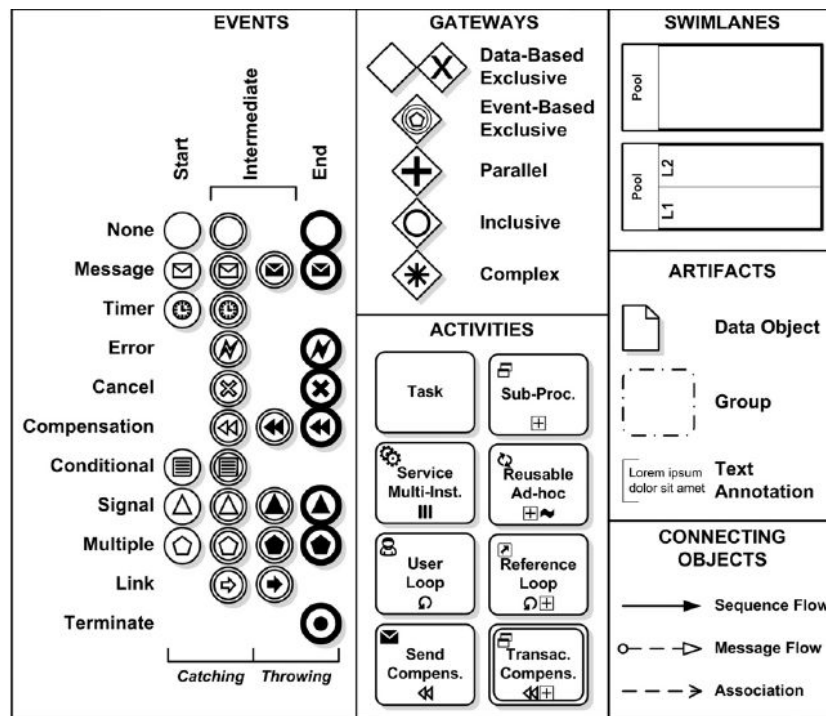


Figure 3. Eléments du langage BPMN (Chinosi & Trombetta, 2012)

Il existe quatre catégories d'éléments :

- La catégorie des « objets de flux » (« flow objects ») qui regroupe les « activités » (« activities »), les « évènements » (« events ») et les « gateways ».

Le standard BPMN 2.0 définit « activity » comme étant un travail réalisé dans le processus métier. Une activité peut être atomique (« task ») ou composée (« sub-process », « call activity », etc).

Les « events » sont des éventualités qui peuvent s'introduire au cours du processus et qui peuvent enclencher des changements sur le processus comme la réception d'un message, le déclenchement d'un minuteur ou le repérage d'une erreur, etc.

Les « gateways » sont utilisés pour contrôler les flux de séquence quand ceux-ci convergent ou divergent.

- La catégorie des « connecting objects ». Les « connecting objects » offrent deux façons d'interconnecter les activités entre elles : le « sequence flow » pour tracer le flux de séquence entre les activités et le « message flow » pour tracer le flux de messages échangés entre les activités. L'élément « association » permet de relier un « artifact » à un objet de flux.

- La catégorie des « *artifacts* » (*data object et annotation*). Les « *artifacts* » sont utilisés pour fournir des informations supplémentaires sur les éléments d'un processus.
- La catégorie des « *swimlanes* » (*pool et lane*). Les « *swimlanes* » donnent la possibilité de faire des regroupements des éléments BPMN.

2.3.2.3 Modèles de traçabilité des décisions

Cette dernière catégorie est une forme particulière des modèles de processus et s'intéresse particulièrement à la représentation des aspects décisionnels, individuels et collectifs, relatifs à la conduite du processus de conception et aux choix des solutions techniques pour répondre aux objectifs de conception. Il s'agit aussi de la traçabilité des justifications des choix et des décisions tout au long du déroulement du projet afin de faciliter le post-diagnostic des causes d'échec d'un projet.

Eynard (Eynard, 1999) a proposé dans le cadre de sa thèse d'introduire le concept d'"activité de conception" pour représenter simultanément les différentes actions opératoires (exécuter), décisionnels (décider) et de recherche de solution (conception) dans un projet de conception. Une activité type est alors associée à une transformation d'état sur son objet, un ensemble de ressources comme support et éventuellement un support complémentaire regroupant les objectifs à atteindre, les contraintes ainsi que les critères et variables de décision et d'évaluation.

Bekhti (S Bekhti, 2003) s'est intéressé dans ses travaux de thèse à la modélisation de la logique de conception à travers une traçabilité des différentes décisions prises au cours des réunions de revue de projet ou autres réunions coopératifs pour résoudre un problème technique ou un conflit. L'outil "DypKM" proposé par cet auteur décrit "un processus dynamique de définition et de réutilisation de mémoires de projets". Il utilise le concept de schémas relationnels pour structurer la logique de conception à travers six éléments complémentaires :

- Le problème discuté au cours d'une réunion collaborative. Ce problème peut être aussi bien un objectif à atteindre, un problème technique relatif au produit ou un problème dans le processus de conception ou dans l'organisation du projet.
- Les participants au processus décisionnel et aux réunions de travail ainsi que leur rôle, leurs compétences et leur appartenance organisationnelle.
- Les suggestions émises par les participants au cours d'une réunion collaborative ainsi que les différentes propositions de résolution de problèmes.
- Les arguments émis par les participants dans une discussion pour appuyer une suggestion ou au contraire s'opposer à certaines suggestions des autres participants.
- Les critères représentant une caractérisation des éléments de négociation. Ils permettent de mettre à jour les points principaux mis en avant dans une discussion et évaluer les différentes suggestions et décisions prises au cours du processus.
- Les décisions éventuelles prises à la fin de réunion collaborative. Un lien est établi entre la décision et la situation du problème spécifiant l'état de résolution de ce dernier et permettant de garder une trace de la projection de la décision sur le projet.

Un autre exemple de modèle de traçabilité des justificatifs de décision en conception a été proposé par (Bigand, Mekhilef, & Bourey J.P., 2006). Ce modèle, décrit avec UML, utilise une méta-classe générique nommée "classe entité" pour représenter au même niveau toutes les connaissances d'un projet de conception ainsi que les relations entre les différents éléments du domaine étudié. L'ensemble des décisions prises en cours de projet de conception est répertorié dans la documentation du projet. Un document est composé d'Item(s) lui-même décomposable en autres Item(s), ce dernier se compose de Contenu pouvant être une image, du texte ou une référence à une autre entité le tout relié à la justification de décision par une relation d'héritage. Dans un autre modèle complémentaire, les auteurs introduisent la notion de version pour conserver la trace des versions successives de ces entités.

2.3.3 Mémoires organisationnelles

En conception, un cycle de traçabilité se traduit par une double relation permanente entre l'activité des acteurs de conception et les mémoires de connaissances associées (Figure 4). Ces mémoires sont alimentées en permanence par des mécanismes de traçabilité de l'activité du concepteur et/ou du manager (Belkadi, 2006). Le concepteur à son tour exploite les connaissances capitalisées pour les réutiliser dans des problèmes qu'il juge similaires. Par son activité, le concepteur génère de nouvelles connaissances, qui pourront être traitées pour mettre à jour le contenu de la mémoire. Généralement, la mise à jour s'effectue après validation par un expert de la qualité des informations renseignées par le concepteur (exactitude, complétude, domaine de validité).

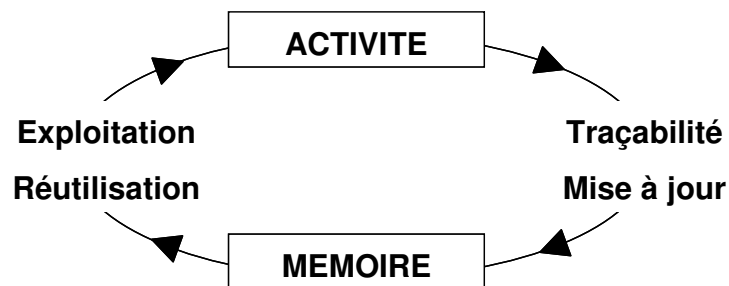


Figure 4. Cycle traçabilité – exploitation en conception

Derrière les concepts de traçabilité et de capitalisation de connaissances (Belkadi, Bonjour, & Dulmet, 2006), on retrouve le concept de mémoire d'entreprise (Barthès, Dieng, & Kassel, 1999) qui se décline en plusieurs types : mémoire métier, mémoire organisationnelle, mémoire projet etc.

La mémoire de projet est définie comme étant « une mémoire des connaissances et des informations acquises et produites au cours de la réalisation des projets ». Selon (Matta, Ribiere, & Corby, 1999), la structure d'une mémoire de projet comprend deux grandes parties :

- Une description du contexte du projet et de tous les éléments présents au cours de son déroulement. Elle concerne d'une part, la description des ressources et des contraintes présentes dans l'environnement de travail et d'autre part, l'organisation du projet en termes d'objectifs, de tâches et de participants.

- Une description de la logique de conception à travers des informations concernant les problèmes rencontrés, les méthodes utilisées pour résoudre ces problèmes ainsi que les solutions et les décisions retenues au cours d'un projet de conception.

Plusieurs méthodes, génériques et spécifiques, de capitalisation de connaissances et de construction de mémoire d'entreprise, ont été développées dans la littérature (Liao, 2003), (Matta, Corby, & Ribiere, 1999). Certains proposent des outils de conception collaborative pour aider à la capitalisation mais surtout au partage des connaissances du projet.

Au niveau des méthodes génériques, nous pouvons mentionner la méthode KADS (Schreiber, 1992), la méthode MKSM (Ermine et al., 1996), la méthode REX (Malvache & Prieur, 1993). Ces méthodes ont déjà largement été décrites dans des travaux sur la gestion des connaissances.

(Kwan & Balasubramanian, 2003) proposent un modèle OMT de ce concept dans une optique de réalisation d'une mémoire organisationnelle intégrant les informations contextuelles. Le système de connaissances proposé a pour objectif d'intégrer la capture des connaissances dans les processus organisationnels en tenant compte du contexte de création et d'utilisation de ces connaissances.

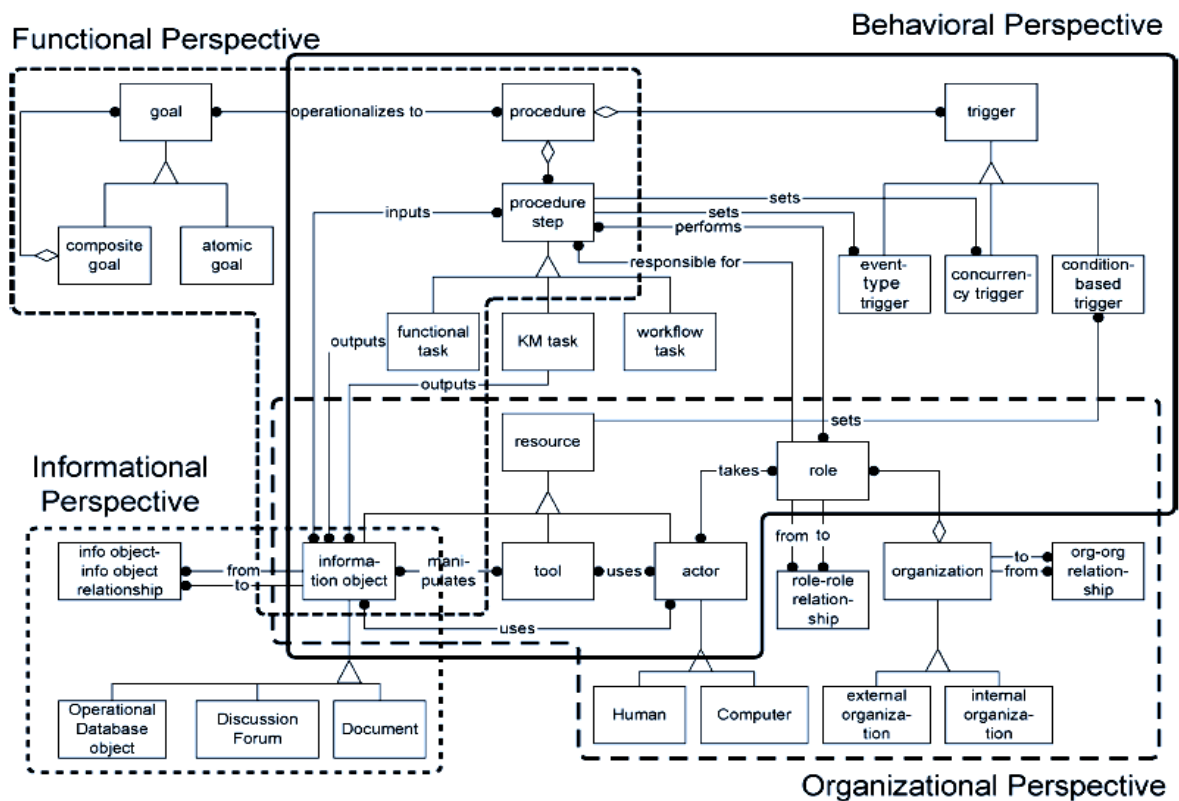


Figure 5. Modèle KnowledgeScope d'une mémoire organisationnelle (Kwan & Balasubramanian, 2003)

Les éléments du contexte organisationnel sont structurés dans ce modèle à travers quatre vues complémentaires (Figure 5) :

- La vue fonctionnelle traduit les différents objectifs organisationnels ainsi que les étapes de processus de transformation portant sur les informations et les objets.
- La vue comportementale décrit la réalisation des activités, le fonctionnement et le contrôle des différentes procédures à travers les différents déclencheurs.

- La vue informationnelle englobe différentes informations manipulées par un outil et / ou produites au cours des activités opérationnelles ou de KM.
- La vue organisationnelle comporte deux types d'informations : un inventaire des ressources intervenant dans la réalisation des tâches et une description de la structure organisationnelle à travers les différents rôles que peuvent tenir les acteurs.

La description des concepts clés de la situation est regroupée dans les vues organisationnelle et fonctionnelle du modèle : acteur, outil, objectifs, tâche, informations (en tant qu'objets) et le processus ("procédure" dans le modèle). Le concept de "rôle" joué par un acteur dans une activité ("procédure step" dans le modèle) est important. Il correspond ici à une position dans l'organisation. Le concept de "procédure" permet de relier l'activité de l'acteur aux différents objectifs de l'organisation. Trois types de tâche sont distingués dans le modèle : des tâches fonctionnelles portant sur l'objet (de nature informationnelle ici), des tâches de gestion des connaissances ("KM task") et des tâches de gestion des flux d'information ("workflow task").

2.4 Ontologies

L'ontologie représente un vocabulaire consensuel qui définit le sens des concepts et des relations entre ces concepts. C'est une description formelle et explicite des concepts dans un domaine donné, de leurs propriétés et de leurs relations (Gruber, 1993)(Noy, McGuinness, & others, 2001). On utilise les classes pour spécifier des concepts et des propriétés pour représenter ses attributs et ses relations.

L'ontologie est un outil de l'intelligence artificielle (Lee & Chen, 2012b) et constitue un outil conceptuel puissant pour la modélisation des connaissances. La construction d'une ontologie permet de définir une terminologie et une sémantique facilitant la représentation et la compréhension des connaissances.

L'ontologie est utilisée dans le domaine de gestion de connaissances pour les fins suivantes :

- Formaliser les concepts de base constituant les mémoires organisationnelles,
- Capturer les connaissances provenant de ressources hétérogènes et distribuées,
- Récupérer et partager les connaissances stockées dans la mémoire,
- Assurer une bonne communication entre agents dans les systèmes multi-agents qui doivent formuler et comprendre les messages qu'ils échangent.

D'après (Kamsu Foguem, Coudert, Béler, & Geneste, 2008), les ontologies ont les rôles principaux suivants: communication, interopérabilité, raisonnement et inférence.

De nombreux travaux ont été réalisés dans différents domaines pour développer des ontologies et élaborer des systèmes à base de connaissances : par exemple, l'enseignement de l'architecture (Eliseo, Oliveira, & Pellegrino, 2011), la gestion des risques (Ebrahimipour, Rezaie, & Shokravi, 2010) ou la modélisation des produits (Bock, Zha, Suh, & Lee, 2010)(Sun, Fan, Shen, & Xiao, 2012).

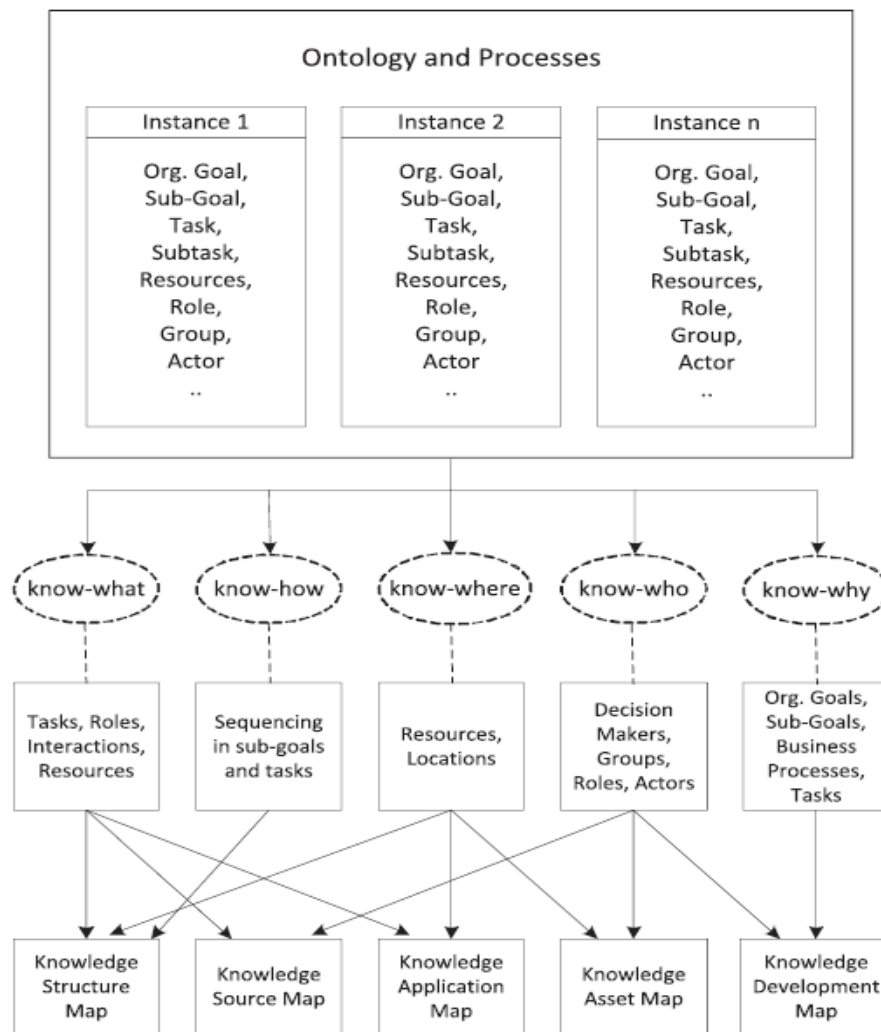


Figure 7. Cartes de connaissances pour les processus de conception (L. Rao et al., 2012)

Les cartes de connaissances sont construites selon la perspective d'analyse qui va servir aux décideurs pour prendre aux mieux les décisions dans le but de rendre le processus plus efficace.

Cette ontologie organisationnelle est construite à l'aide de différentes techniques de modélisation organisationnelle comme AALADIN, AGR (Agent/Group/Role), MOISE+ et AIML (Agile Integration Modeling Language). La partie formelle est développée à l'aide de l'outil Protégé-OWL.

2.4.2 L'ontologie Failure Analysis/ Failure Identification (FA/FI)

Raghu et al. (Raghu & Vinze, 2007) définissent le processus métier comme étant le contexte de la connaissance à gérer et comme une collection d'activités et de tâches interdépendantes et organisées dans le but d'atteindre des objectifs métiers spécifiques. Selon les auteurs, le processus métier traverse souvent de multiples départements fonctionnels et de multiples hiérarchies à l'intérieur ou à l'extérieur de l'organisation concernée, ce qui exige que les activités soient bien coordonnées pour accomplir efficacement les objectifs.

Les concepts de coordination sont expliqués dans les travaux de (Malone & Crowston, 1994; Malone & Smith, 1988; H. R. Rao & Chaudhury, 1995). En se basant sur ces travaux,

Raghu et al. (Raghu & Vinze, 2007) définissent quatre aspects clés d'un contexte de connaissances : l'exécution du flux d'activités (workflow), le traitement des informations, la prise de décision et la structure de motivation. Ils lient ensuite ces quatre aspects à trois facettes de gestion des connaissances dans les organisations : 1) stockage et récupération 2) partage et transfert, 3) création et synthèse de la connaissance (Figure 8).

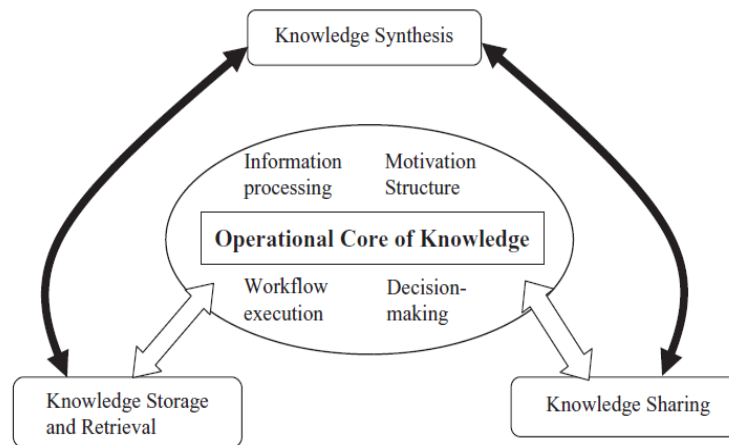


Figure 8. Le contexte processus métier dans la gestion de connaissances (Raghu & Vinze, 2007)

Les activités et les relations de précédence entre les activités forment la structure « workflow » du processus métier.

L'utilité du « workflow » dans le contexte KM se vérifie : dans les organisations orientées processus et « workflow » telles que les départements supports aux clients, ou de traitement de réclamations, il est possible de bénéficier de la récupération des instances de workflow passées dans des contextes similaires. Les efforts du KM pour améliorer l'accès et la récupération de ces informations visent à améliorer l'efficacité et la consistance du flux. La définition des règles métier de l'exécution du workflow exige la capture des bonnes pratiques dans les organisations.

Les auteurs illustrent leur approche avec un exemple sur le processus d'identification et d'analyse des défaillances (FA/FI). La solution envisagée définit une structure de l'exécution du flux des activités pour autoriser un stockage adapté des données liées aux défaillances dans des formats de base de données traditionnels et dans les rapports d'analyse. Il définit cette structure à l'aide d'une ontologie. Par contre, les auteurs n'expliquent pas comment ils ont construit leur ontologie de la structure du flux des activités du processus FA/FI, ce qui rend leur approche difficile à réutiliser.

2.4.3 Le cadre d'ontologie organisationnelle pour le SBPM (Semantic Business Process Management)

Le SBPM représente une initiative très intéressante dans le cadre des travaux visant à atteindre un haut niveau d'automatisation du cycle de vie de BPM grâce aux technologies du web sémantique.

Dans le cadre du SBPM, il s'avère intéressant d'utiliser des ontologies pour représenter les aspects statiques et dynamiques de l'entreprise et de sa chaîne de valeurs (Filipowska, Hepp, Kaczmarek, & Markovic, 2009).

La représentation sémantique des processus définie par le SBPM est divisée en trois groupes :

- L'ontologie Processus pour décrire la structure du processus (D.1.1, 2007).
- L'ontologie organisationnelle qui fournit une description des artefacts, des acteurs, etc. évoluant lors du processus.
- L'ontologie « domaine spécifique » qui fournit une description plus spécifique d'une organisation d'un domaine donné.

(Filipowska et al., 2009) développent l'ontologie organisationnelle qui comprend :

- L'ontologie « structure organisationnelle » qui se focalise sur la structure organisationnelle (hiérarchique) de l'entreprise.
- L'ontologie « unités organisationnelle » qui fournit une spécification des unités typiques de l'entreprise.
- L'ontologie « rôles métier » pour représenter les rôles présents dans les organisations.
- L'ontologie « fonctions métier » qui fournit une hiérarchie des différentes fonctions pouvant être réalisées dans l'entreprise.
- L'ontologie « ressources métier » pour décrire les ressources consommées ou résultant d'une tâche du processus.
- L'ontologie « objectifs métier » pour décrire la hiérarchie des objectifs métiers.

Structure organisationnelle	Organisation, entité légale/illégale, unité organisationnelle, fonction métier, personne, compétence, rôle, position organisationnelle, ressource
Unité organisationnelle	Corporation, division, département, groupe, comité
Rôles métier	Rôle métier, rôle interne, rôle externe, etc. (37 concepts).
Fonctions métier	Fonction : gestion des clients, gestion marketing, gestion des ventes (40 concepts). Activity Or Step: UpdateSalesOrder, etc. (920 concepts).
Objectifs métier	Objectif quantitatif, objectif qualitatif, description, mesure, priorité, valeur désirée, valeur actuelle, etc.
Ressources métier	Poste de soudage, récipient de galvanisation, etc.

Tableau 1. Concepts clés de l'ontologie SBPM

2.5 Traces d'activité et systèmes à base de traces

Dans cette section, nous présentons les principaux travaux concernant les systèmes à base de traces.

2.5.1 Définition d'une trace

Une trace est définie par (Chabin, 2004)(Settouti, Prié, Mille, & Marty, 2006b) comme étant une « *empreinte, marque laissée par une action* ». Dans un environnement logiciel, les traces numériques sont des enregistrements automatiques temporalisés et exploitables, et dont les constituants sont liés à la manipulation de l'environnement.

Certains documents numériques possèdent des méta-données, par exemple les dates de création ou de modification. Ces enregistrements automatiques temporalisés et exploitables, et dont les constituants sont liés à la manipulation de l'environnement numérique impliqué, sont considérés comme des traces numériques d'utilisation.

En fonction de l'environnement numérique, les traces peuvent être des clicks, des logs, des créations d'objets, des modifications, des vidéos, etc. Le choix des données à acquérir dans les traces est lié aux objectifs de l'observation. Parmi les exemples, on trouve l'identification des styles des apprenants dans des environnements d'apprentissage en ligne (e-Learning) (Nabalia Bousbia, Rebaï, Labat, & Balla, 2010) (Clauzel, Sehaba, & Prié, 2011), l'analyse de l'activité des utilisateurs (Sanderson & Fisher, 1994), la modélisation de l'activité de conduite d'une voiture (O. Georgeon, Henning, & Bellet, 2007), la reconnaissance d'activité (Gu, Chen, Tao, & Lu, 2010) ou l'analyse de l'utilisabilité des interfaces dans le domaine d'interaction avec l'ordinateur (Hilbert & Redmiles, 2000).

Selon (Cram, Jouvin, & Mille, 2007), les logs désignent des fichiers dans lesquels sont stockées des traces de requête ou opération d'un utilisateur, enregistrées au format textuel. L'enregistrement des logs constitue l'historique des interactions.

Selon (Nabila Bousbia, 2011), la collecte des données fournit des données qualifiées de traces brutes, difficilement exploitables. La création des traces modélisées à partir des logs est un processus complexe qui nécessite plusieurs opérations (filtrage, recomposition en sessions, etc.). Les logs peuvent contenir de grands volumes d'informations, parfois sans le contexte dans lequel les informations ont été générées. Les données stockées ne sont pas structurées et pour la plupart inadaptées à l'objectif souhaité (Cram et al., 2007).

Pour pouvoir réutiliser les expériences d'interaction, il faut donc modéliser les traces (Cram et al., 2007). Les traces brutes sont transformées en traces modélisées par le système de transformation du SBT. Dans ce cadre, Settouti et al (Settouti et al., 2006a) définissent une trace comme une collection d'observés qui peuvent être temporellement situés. Un observé peut être tout élément de l'environnement de l'utilisateur (une entité, une action) qui fait sens dans l'observation de son activité.

Selon (Cram et al., 2007), chaque observé de la trace peut être considéré comme une instance de classe d'observé appartenant à une structure hiérarchique de classes d'observés appelée le modèle de trace. Ce modèle de trace peut être formalisé par une ontologie définissant les interactions utilisateur/système.

2.5.2 Systèmes à base des traces

Les traces peuvent être réutilisées à deux fins : l'analyse et l'assistance. La réutilisation des traces d'interactions est une approche de la réutilisation des expériences passées.

(Settouti et al., 2006a) proposent une architecture du SBT composée de trois systèmes interdépendants : système de collecte, de transformation et de visualisation.

Le système de collecte enregistre les interactions en créant une trace brute. Les traces sont enregistrées suivant une hiérarchie de classes observées appelée le modèle de traces.

Le système de transformation est au cœur du SBT. Il génère de nouvelles connaissances à partir des traces collectées. La définition des règles de transformation dépend de l'objectif d'utilisation des traces. L'ensemble des traces collectées et transformées est proposé à l'utilisateur par l'intermédiaire d'un système de visualisation permettant de les exploiter, les analyser et les interpréter.

Souvent, les traces d'interaction collectées par le SBT sont des traces non structurées. Le système de transformation est basé sur une ontologie pour les transformer en traces modélisées et construire des connaissances utiles qui seront stockées en base de connaissances (M. H. Karray, 2012) (M.-H. Karray et al., 2014).

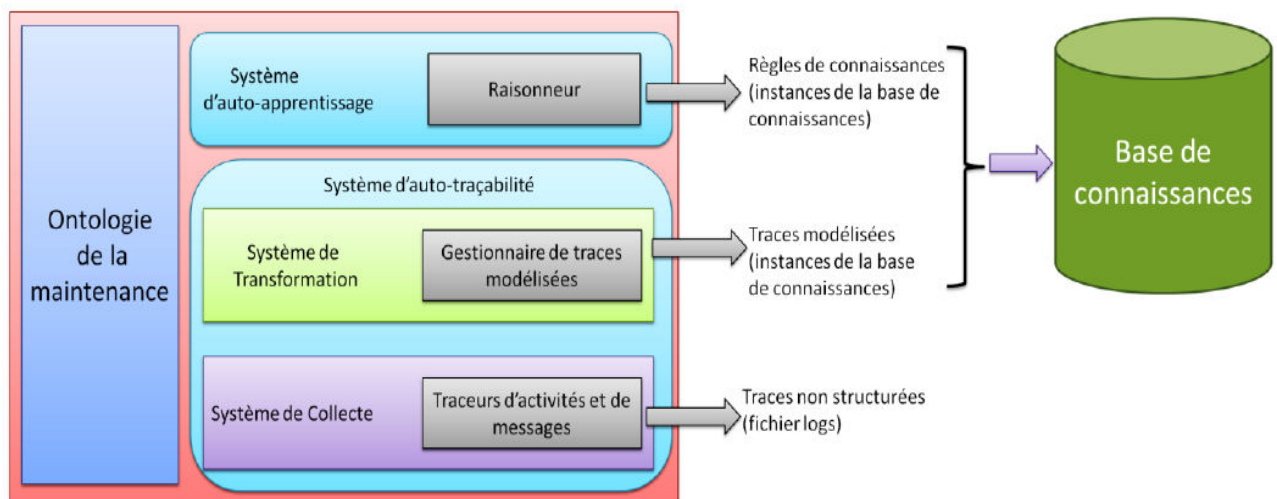


Figure 9. Système à base de traces pour une plateforme de maintenance (M. H. Karray, 2012)

Dans leur article (M.-H. Karray et al., 2014), les auteurs présentent un système à base de traces, appelé PETRA (pour : Process Evolution using a TRAce-based system), qui permet de tracer et analyser les activités supportées par une plateforme de maintenance. Ce système transforme les logs en des traces modélisées en utilisant une ontologie du domaine de maintenance. Ensuite un processus d'apprentissage qui est appliqué sur la vue conceptuelle permet de découvrir de nouvelles règles en utilisant une approche de type arbre de décisions. Finalement, ces nouvelles règles sont analysées et évaluées au regard des règles existantes dans le système à base de connaissances afin d'éviter les incohérences, les contradictions et les conflits.

2.5.3 Ontologies dans les systèmes à base des traces

Les traces sont des sources de connaissance très pertinentes. La difficulté demeure d'automatiser l'extraction des connaissances à partir des traces, ou de faciliter cette extraction.

Dans l'approche MUNETTE (Champin et al., 2003), les auteurs proposent de spécifier deux modèles : le modèle d'utilisation et le modèle d'observation, qui serviront de référence pour la collecte des traces dites primitives. Ces modèles vont définir ce qui sera extrait et collecté dans la trace primitive. Dans cette approche, les sources (traces brutes) ne sont pas explicitement déterminées, mais il s'agit des historiques de navigation. La connaissance réside dans les épisodes retrouvés dans les traces primitives. Ici, il ne s'agit pas de transformation, mais plutôt d'une analyse des traces primitives : les épisodes sont de même nature que les traces. Les traces sont constituées des objets d'intérêt (objets observables) que sont les entités, les événements et les relations.

Le modèle d'utilisation décrit quels types d'entités, d'événements et de relations il faut observer pour produire les traces primitives. Le modèle d'observation décrit le comment, c'est-à-dire l'ensemble de règles qui contrôlent le processus de production des traces.

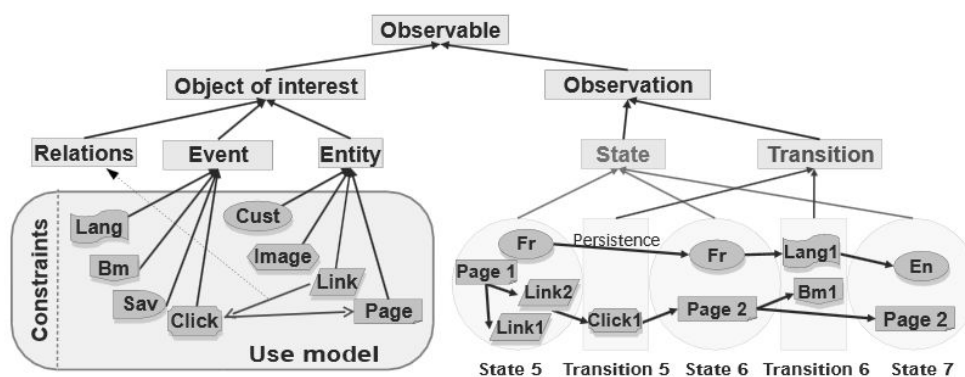


Figure 10. Modèle de traces selon (Champin et al., 2003)

En effet, les traces primitives permettent de décrire explicitement les interactions hommes machines. Dans l'approche MUNETTE, les auteurs proposent d'extraire à partir des traces primitives des épisodes qui sont spécifiques à des expériences correspondant à des tâches spécifiques et qui peuvent être réutilisés dans des cas similaires. Mais ils n'expliquent pas comment procéder pour retrouver ces épisodes à partir des traces.

Dans le travail d'O. Georgeon (O. Georgeon, 2008), l'auteur présente l'atelier ABSTRACT (Analysis of Behaviors and Situation for menTal Representation Assessment and Cognitive acTivity modeling) basé sur un système à base de traces. L'atelier comporte un éditeur d'ontologie permettant de définir les modèles des traces. L'auteur présente également sa méthodologie de construction de connaissances à partir des traces. Cette méthodologie apparaît dans le système de manipulation des traces permettant de les transformer et les visualiser.

En effet, l'outil d'import des traces collectées permet de convertir les traces collectées produites par le système de collecte en des traces collectées manipulables par le système.

L'utilisateur doit renseigner l'ontologie avant d'importer les traces collectées. Chaque objet d'intérêt susceptible de contenir de la connaissance et présent dans la trace collectée

doit avoir un type défini dans l'ontologie avant l'import (par exemple : accélération, volant gauche).

Le point d'intérêt importé est désigné par le terme « observé » et il est représenté par un symbole dans la trace.

L'exploitation des traces est assurée ici par le système de visualisation. L'objectif du système est de construire un outil qui permet d'analyser les activités et les comportements cognitifs lors de la conduite d'un véhicule. L'auteur n'est pas allé plus loin pour proposer une approche pour réutiliser les résultats de l'analyse des traces. En effet, les traces sont enrichies si besoin par le processus de transformation. Le moteur de transformation est capable de mettre en œuvre des règles sémantiques associées aux observés pour en déduire de nouveaux observés. Les règles permettent de définir une nouvelle sémantique particulière aux traces, elles sont mises en place sous forme des requêtes SPARQL.

Dans le même but, c'est-à-dire extraire et formaliser la connaissance pour sa réutilisation, (Roldán, Gonnet, & Leone, 2010) ont proposé un outil (TracED) qui permet la capture du processus de conception. Ils s'intéressent plus particulièrement aux choix de conception. Dans cette approche, l'utilisation des outils informatiques pour la collaboration est exploitée pour recueillir des traces de l'activité sans perturber l'activité de conception. Cet exemple est très intéressant puisqu'il nous montre qu'à partir des fragments d'informations numériques, il est possible de récupérer des détails du processus de conception (dans ce cas les choix de conception). Ces fragments d'information gardent une mémoire sur l'activité. Cela implique qu'on peut puiser dans cette mémoire pour acquérir de la connaissance.

D'après Lancieri (Lancieri, 2009), dans le domaine des activités humaines médiatisées, cette idée a été exploitée pour réaliser des études de phénomènes collectifs dans une approche psychosociale. C'est le cas de l'analyse qui se fait dans des sites web d'achats pour vérifier les comportements des utilisateurs et à partir de ces comportements, proposer des recommandations. Par exemple, quand un utilisateur entre dans un site web d'achat de musique, selon son profil (âge, sexe, éléments récemment visités, etc.), le site peut recommander les disques dont il pourra être intéressé parce que d'autres utilisateurs avec le même profil les ont achetés.

La diversité des applications existantes montre tout l'intérêt porté à l'extraction, la formalisation et l'exploitation des traces d'activités. Dans chaque application, un modèle de traces est requis. Un modèle de trace est le vocabulaire qui décrit les classes d'objets observés et les classes des relations. Le modèle de trace permet la compréhension et la manipulation des traces en décrivant abstraitement les éléments, c'est-à-dire en catégorisant les éléments observés et les relations entre ceux-ci.

La plupart des modèles de traces sont orientés vers l'analyse d'activités d'apprentissage collaboratif. Cependant, le modèle ABSTRACT (Analysis of Behavior and Situation for mental Representation Assessment and Cognitive acTivity modeling), qui est en même temps un outil, est applicable dans les différents types d'analyses d'activités et comportements.

La Figure 11 résume le fonctionnement de ce système. Le premier pas consiste à collecter les données brutes. Ensuite, les données brutes sont filtrées pour produire les premières traces dites symboliques. La troisième étape consiste en une modélisation de ces traces symboliques dans des traces d'un plus haut niveau d'abstraction. La dernière étape consiste en la production d'un modèle explicite de l'activité à l'aide des traces modélisées. Pendant le processus de modélisation, l'analyste formalise les transformations des données brutes en modèles de traces à l'aide des règles de transformation et d'une ontologie.

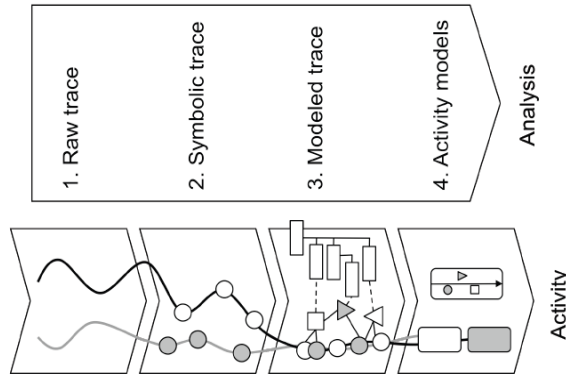


Figure 11. Modèle des traces d'activité ABSTRACT (OL Georgeon, Mille, & Bellet, 2012)

Dans le Tableau 2, quelques modèles de traces proposés dans la littérature sont présentés, ainsi que les connaissances qu'ils manipulent et leur utilité.

Système à base de traces	Auteur(s)	Domaine	Objectifs et proposition	Description du modèle
MUSETTE (Modeling USEs and Tasks for Tracing Experiences)	(Champin et al., 2003)	Activités de navigations WEB et autres activités	Modéliser des expériences des utilisateurs d'un système informatique	Traces composés de séquences des états et transitions, identification des patrons communs appelés signatures
Trace Based System and M-Traces	(Mille, 2006)	Situations d'apprentissage et autres activités humaines	Visualiser et construire les événements temporels par un raisonnement à partir de traces	Evènements temporels: Traces numériques d'utilisation, texte, vidéo
UTL (Usage Tracking Langage)	(Choquet & Iksal, 2007)	Environnement d'apprentissage	Créer, proposer, définir, obtenir et utiliser les traces	Description XML des informations
ABSTRACT	(Georgeon et al., 2007)	Analyse ergonomique de la conduite automobile	Analyser et représenter des comportements cognitifs	Traces des comportements, patrons, ontologies pour les nouvelles connaissances
MULCE	(Reffay, Chanier, Noras, & Betbeder, 2008)	Environnement enseignement/ apprentissage	Définir le matériel nécessaire de la plateforme de partage	Données d'interactions (vidéo, traces digitales, texte), contexte d'apprentissage (objectifs, protocole)
—	(El kadiri, Pernelle, Delattre, & Bouras, 2008)	Computer Supported Cooperative Work (CSCW), PLM	Piloter le processus collaboratif dans les systèmes PLM via le contrôle des indicateurs	Fichiers log et base de données
PETRA	(M.-H. Karray et al., 2014)	Plateforme de maintenance	Transformer les logs en des traces modélisées en utilisant une ontologie du domaine de maintenance	Données d'interaction (logs)

Tableau 2. Description et comparaison de systèmes à base de traces

2.6 Systèmes multi-agents

Dans cette section, nous présentons le paradigme Agent puis des travaux relatifs à l'utilisation des Systèmes Multi-agents pour l'ingénierie des connaissances.

2.6.1 Le paradigme Agent

2.6.1.1 La notion d'agent

De nombreuses définitions ont été données pour décrire la notion agent. Dans les dictionnaires français nous avons noté les trois définitions du mot « Agent » :

- Une personne qui réalise des actions ou accomplit des activités spécifiques ;
- Quelqu'un qui agit à la place d'un autre ou en utilisant son autorité ;
- Tout phénomène qui a une action déterminante.

La troisième définition détermine qu'un agent est un catalyseur qui effectue des actions qui auront des impacts dans un environnement donné. La seconde définition est celle qui est utilisée le plus souvent dans le langage naturel. Si on l'associe à la première définition on pense directement à des noms tels que l'agent d'affaire, l'agent d'assurance, l'agent de change, l'agent de police... Dans ce cadre la notion d'agent représente une personne qui agit en son nom et réalise des actions destinées à aider d'autres personnes. Cette notion rejoint donc la notion de service. De plus on comprend très vite que l'agent possède des qualités utiles mais qui sont néanmoins limitées à son domaine précis (affaire, assurance, police...).

Puisque nous avons expliqué la notion d'agent, il est nécessaire de comprendre comment travailler avec un agent. Habituellement nous demandons à l'agent de réaliser une certaine tâche en le sollicitant sur une de ses compétences. Par exemple nous demandons à notre agent d'assurance de nous établir un devis d'assurance pour un nouveau véhicule. Il est à noter dans ce cas que nous apportons plus d'importance à la qualité du résultat que l'agent fournit qu'au processus qu'il a utilisé pour accomplir la tâche demandée. L'agent lui-même doit adapter ses processus en fonction des objectifs qu'il s'est fixés (satisfaction client, rentabilité du cabinet d'assurance...). Parfois il peut arriver que les objectifs soient en conflit. Dans ce cas, l'agent devra choisir le bon objectif ou donner des priorités. Cette prise de décision est en rapport avec la complexité des actions qu'il a menées et ne peut être efficace que dans le cas où l'agent possède un nombre d'informations suffisant. L'agent peut ainsi décliner la demande ou poser des conditions sur la réalisation de la tâche.

Nous avons décrit dans ce paragraphe la notion d'agent au point de vue social dans le monde réel. Dans la suite de ce mémoire, pour éviter les confusions, nous nommerons ce type d'agent, « acteur ». La suite de ce chapitre est dédiée aux agents logiciels. Ces derniers sont des entités informatiques qui sont conçues dans l'esprit de reproduire les caractéristiques des agents du monde réel.

2.6.1.2 Les agents logiciels

Un programme informatique est appelé 'agent logiciel' si il exhibe un comportement possédant de nombreuses caractéristiques d'un agent humain. Les agents logiciels ne sont pas un concept nouveau et héritent des recherches du domaine de l'intelligence artificielle

distribuée (Russell & Norvug, 2009). Le paradigme agent est approfondi avec les travaux de Jennings et Wooldridge (Jennings & Wooldridge, 1998) où ce dernier explique que les agents sont performants à partir de l'instant où ils présentent des facultés à traiter une série d'applications. Dans ce cas le paradigme agent est plus approprié que celui des objets (Bersini & Wellesz, 2002) pour représenter des entités actives d'un domaine ou d'un système. En fait, les entités actives d'un domaine utilisent des ressources qui sont passives. Un document est considéré comme un objet passif. La différence entre un objet et un agent est la capacité de ce dernier à agir de façon autonome (Jennings & Wooldridge, 1998). Cette capacité procure aux agents le contrôle sur leurs comportements et la possibilité d'agir sans l'intervention des humains contrairement aux objets qui n'ont que le contrôle de leurs états. Un objet peut invoquer une méthode d'un autre objet qui sera exécutée alors qu'un agent peut solliciter une action de la part d'un autre agent, mais ce dernier décidera ou non de satisfaire la demande.

2.6.1.3 Les propriétés principales des agents

Comme nous l'avons dit précédemment, un agent est conçu pour offrir des services selon son domaine de compétences. Ce domaine doit être défini comme un environnement dans lequel l'agent perçoit des informations et peut déclencher des actions qui ont des effets sur cet environnement (Wooldridge, 2002).

Le modèle d'interaction de (Wooldridge, 2002) explique que l'agent ne peut pas clôturer une tâche tant que les interactions avec son environnement ne sont pas terminées. Ceci engendre plusieurs propriétés des agents. Nous citons celles qui sont présentées dans la littérature et qui nous intéressent tout particulièrement :

Propriété 1. Un agent logiciel est un programme informatique situé dans un environnement.

Propriété 2. Un agent logiciel offre des services. Son comportement peut être observé par les actions qu'il réalise ainsi que ses résultats. Les processus internes qui permettent de déclencher ses actions sont intégrés à l'agent et transparents aux utilisateurs.

Propriété 3. Un agent logiciel est en mesure d'agir de manière autonome sur son environnement dans l'objectif de réaliser ses objectifs c'est-à-dire d'accomplir des services.

D'après Wooldridge, un agent est dit 'intelligent' lorsqu'il est en mesure d'effectuer ses actions dans un environnement non-déterministe, c'est-à-dire un environnement dans lequel les événements ne sont pas prévisibles. L'ensemble des informations d'un tel environnement n'est pas accessible complètement. Le contrôle des actions par l'agent peut évoluer en fonction des informations successives qu'il perçoit. Cette capacité engendre d'autres propriétés telles que :

Propriété 4. Un agent logiciel est réactif c'est-à-dire qu'il adapte ses actions en fonction de l'environnement qu'il perçoit. L'agent doit donc partager ses décisions en fonction des informations provenant de son environnement et de ses objectifs.

Propriété 5. Un agent logiciel est proactif c'est-à-dire qu'il prend lui-même des initiatives en fonction de ses objectifs et des informations qu'il perçoit. Dans le cas de problèmes complexes, il peut déléguer une partie du problème à d'autres agents. Wooldridge appelle cette fonctionnalité l' « habilité sociale » des agents.

Propriété 6. Un agent logiciel possédant une habilité sociale est en mesure d'interagir avec d'autres agents pour satisfaire les tâches qui lui sont confiées.

Propriété 7. Un agent est en mesure d'apprendre des expériences passées. Le processus de décision des agents doit prendre en compte les expériences passées de l'agent.

2.6.1.4 Des agents aux Systèmes Multi-agents

Jennings et Wooldridge (Jennings & Wooldridge, 1998) définissent un système multi-agents (SMA) comme un ensemble d'agents hétérogènes possédant des motivations propres et agissant pour satisfaire leurs objectifs. De plus ces agents peuvent partager leurs tâches avec d'autres agents. Il n'y a pas de contrôle centralisé des actions. Les agents doivent raisonner et coordonner leurs actions. A l'intérieur de ces systèmes, ils peuvent faire face à des situations complexes en partageant leurs connaissances sur l'environnement. Ce dernier est défini par le monde dans lequel les agents évoluent et les caractéristiques des autres agents avec lesquels ils interagissent. Les propriétés d'un SMA sont nombreuses, elles dépendent du nombre d'agents, de leurs capacités, de leur manière d'interagir... Axel (Axelrod, 1997) souligne la difficulté de définir les propriétés d'un SMA :

« Emergent properties are often surprising because it can be hard to anticipate the full consequences of even simple forms of interaction ».

Ferber dans (Ferber, 1999) explique que les propriétés du SMA sont définies par celles des organisations qui le composent :

« Even societies considered as being complex such as colonies of bees or ants, should not necessarily be considered as individuals in their own right if we wish to understand their organization and the regulation and evolution phenomena prevailing there. In terms of multiagent systems, this means that an organization can emerge from the juxtaposition of individual actions, without its being necessary to define a specific objective (an element from the assembly O) which represents such an outcome ».

Nous expliquerons plus en détail le concept d'organisation et d'interactions entre agents dans la suite de ce mémoire. Il est à noter que Wooldridge (Wooldridge, 1999) précise qu'un comportement intelligent émerge de l'interaction entre plusieurs comportements simples.

La conception d'un système multi-agents comprend la description des rôles des agents, de leurs buts et l'organisation des groupes qui vont composer le SMA. Ce travail est difficile puisqu'il faut prendre en compte également les interactions entre les agents puisque c'est en partie grâce à elles que le système va démontrer son efficacité. Depuis plusieurs années, de nombreuses méthodes ont été proposées pour faciliter la conception des SMA. Nous présentons dans ce chapitre celles qui sont les plus adaptées à nos travaux de recherche.

2.6.2 Quelques systèmes multi agents existants

Cette section présente des travaux concernant des systèmes multi agents conçus au service des domaines se trouvant dans le cadre de nos travaux de recherche, c'est-à-dire l'ingénierie des connaissances et la conception de systèmes mécaniques.

2.6.2.1 *Des systèmes multi agents au service des projets d'ingénierie*

Comme nous l'avons expliqué auparavant, le paradigme agent offre la possibilité de modéliser des activités distribuées ainsi que les échanges entre les acteurs métier. Depuis plusieurs années des travaux de recherche se sont focalisés sur l'utilisation de SMA au service de la conception mécanique. En général les agents sont utilisés pour favoriser la coopération entre les concepteurs formant ainsi un pont entre les outils d'aide à la conception et les acteurs métier.

Un des premiers projets issu de ces recherches fut PACT (Palo Alto Collaborative Testbed) (Cutkosky & Engelmores, 1993). PACT est composé de plusieurs agents recueillant les informations issues des outils de conception. Chaque agent gère un outil de conception. Les interactions entre les agents permettent de partager et diffuser des concepts et terminologies. Un type d'agent supplémentaire, les « Agent's facilitator » est dédié à la communication entre les agents gérant les outils de conception. Le langage utilisé est le KQML (Knowledge Query Manipulation Language) qui facilite la communication entre les agents et la base de connaissances regroupant les données issues des outils de conception. L'objectif de PACT est de fournir aux concepteurs les informations sur les états du produit tout au long du cycle de développement.

Greco et Brown (Greco & Brown, 1996) ont décrit un système pour le paramétrage en conception de ressorts basé sur un ensemble de systèmes à bases de connaissances gérés par des agents appelés SiFas (Single Function Agents). Chacun des agents possède une fonctionnalité particulière (sélectionner un produit, estimer ses paramètres, les évaluer ...). Leurs interactions permettent de réaliser les tâches de conception. De plus la faculté d'apprentissage des SiFas leur permet de minimiser les conflits de paramétrage durant le processus de conception.

McAlinden et al. (McAlinden et al., 1998) décrivent comment des agents peuvent être implémentés pour faciliter le partage d'informations en conception. Les auteurs utilisent une ontologie basée sur le langage de communication entre les agents (ACL) défini par la FIPA. Dans ces travaux la collaboration entre les agents est réduite à l'échange de messages spécifiant les états des entités. Les agents ont pour objectifs de fournir les vues géométriques des pièces aux concepteurs en les affichant au format standard STEP.

Lees (Lees, Branki, & Aird, 2001) propose un SMA en mesure d'afficher les informations relatives à une pièce pour compléter son schéma numérique. Les agents s'échangent des messages XML utilisant une syntaxe dédiée appelée PIML (Product Information Markup Language) et permettant de décrire une pièce à concevoir. Les agents utilisent des mécanismes de raisonnement basés sur un RàPC (Raisonnement à Partir de Cas). Leurs objectifs sont de collecter un maximum d'information sur chaque pièce et de transmettre ces informations aux concepteurs en notifiant les expériences passées.

Wang et al. (Wang & Tang, 2005) proposent un système pour faciliter la coordination et la gestion du processus de conception. Le SMA est composé de sept agents (Project Management Agent, Design Agents, Product Data Management Agents, Design Coordination

Agent, Design Communication Agent, Computational Support Agent, Agent Manager) permettant à la fois de définir les tâches de conception pour un produit, de distribuer ces tâches entre les différents concepteurs, de coordonner les réalisations de chaque tâche et d'assembler l'ensemble des données collectées issues des tâches pour construire une solution globale.

Shumin et al. dans (Shumin, Hamada, Yue, & Weiming, 2006) utilisent le paradigme agent pour développer des assistants au concepteur. Un agent assistant construit un modèle de l'utilisateur comprenant ses intérêts et les tâches qu'il doit accomplir. A partir de ce modèle l'agent devient proactif et propose au concepteur des concepts par lesquels il pourrait être intéressé, les tâches suivantes qu'il devrait accomplir ou encore la consultation de rapports en relation avec ses intérêts. Ces agents assistants ne communiquent pas entre eux, ils sont dédiés à construire et enrichir le modèle utilisateur de leur acteur métier. Dans le même axe de recherche nous pouvons citer les travaux de Brazier et al. (Brazier, Moshkina, & Wijngaards, 2001) et Paraiso (Paraiso & Barthès, 2006) où le paradigme agent est utilisé pour l'assistance aux concepteurs.

Des récents travaux (Rosenman, Smith, Maher, Ding, & Marchant, 2007) ont montré que les SMA peuvent être associés à une plateforme de réalité virtuelle pour faciliter la collaboration entre les concepteurs. L'environnement 3D fournit un support à la collaboration en temps réel aux ingénieurs se trouvant sur des lieux distants et le permet ainsi de proposer des vues différentes selon leur domaine métier et construire une représentation partagée du produit. Les agents appelés 'médiateurs' ont pour objectif de gérer ces différentes vues en créant, modifiant les objets 3D. Ils associent un objet 3D aux concepteurs qui l'ont créé, ils gèrent également les moyens de communications entre les concepteurs par l'intermédiaire d'un "chat". Il existe un agent médiateur pour chaque concepteur. Un autre type d'agents, les agents 'collecteurs' ont pour objectif de gérer les logs de communication ainsi que les données relatives aux objets 3D.

2.6.2.2 Des systèmes multi agents au service de l'ingénierie des connaissances

2.6.2.2.1 Les courants de recherche

L'objectif de l'utilisation des SMA est de reproduire des phénomènes sociaux à partir des analyses et études d'environnements réels. La gestion des connaissances est un domaine qui nécessite la collaboration entre les acteurs métier pour la création, le partage, la mise à jour des connaissances. Ainsi plusieurs travaux de recherche se sont orientés sur la conception de systèmes à base d'agent pour aider les utilisateurs à gérer leurs connaissances. Ces travaux ont donné naissance au concept de « Multi-agents Information System » (MAIS)(Gandon, 2002). Un MAIS est un système multi-agents dont l'objectif est de fournir des fonctionnalités pour gérer et exploiter des informations distribuées (fournir des accès, mettre à jour des ressources, associer des ressources hétérogènes...).

Plus récemment, Van Elst et al. dans (Van Elst et al., 2004) ont proposé le concept d'« Agent-mediated Knowledge Management » (AMKM). Cette approche prend en compte les aspects collectifs d'un domaine tout en respectant les désirs et objectifs des acteurs du domaine. Les AMKM sont ainsi structurés en organisation d'agents avec une description de leurs rôles ainsi que des configurations rendant possible leurs interactions. Ces systèmes facilitent la gestion des connaissances dans des environnements informationnels dynamiques. Certains de ces SMA ont été conçus en complément d'outil de gestion de l'information

(workflow, ontologies, système de recherche d'information...) pour donner naissance à des plateformes telles que FRODO(Abecker et al., 2003), CoMMA (Gandon, Berthelot, & Dieng-Kuntz, 2002) ou Edamok (Bonifacio, Bouquet, & Traverso, 2002).

2.6.2.2.2 Gestion des connaissances sur le Web

Parmi les SMA au service de l'ingénierie des connaissances, nous pouvons citer ceux qui sont orientés recherche de connaissances sur le World Wide Web avec les travaux de Rhodes et Maes (Rhodes & Maes, 2000) qui présentent trois agents pour la recherche en temps réel (just-in-time) de connaissances : l'agent 'Remembrance' qui maintient une liste de documents qui sont en cours de réalisation ou de lecture, l'agent 'Margin Notes' qui utilise les documents consultés dans le navigateur web pour créer le contexte de recherche et l'agent 'Jiminy' qui se sert de l'environnement de l'utilisateur pour déterminer quelles sont les informations à rechercher. Nous pouvons citer également les systèmes 'Watson' (Budzik, Hammond, & Birnbaum, n.d.), Letizia (Lieberman, 1995) et son successeur 'Let's Browse' (Lieberman, Van Dyke, & Vivacqua, 1999) ou encore 'Kodama' (Helmy, Amamiya, Mine, & Amamiya, 2002) qui permettent de guider les recherches des utilisateurs en observant les pages web consultées ou les courriers électroniques.

2.6.2.2.3 Les SMA conçus avec un seul type d'agents

Ces SMA sont qualifiés d'homogènes, ils sont composés de plusieurs agents mais ces agents sont hérités de la même classe c'est-à-dire qu'ils ont tous du même type. Les plus connus sont les agents gérant des réseaux d'échanges Peer-to-Peer (Guizzardi, Aroyo, & Wagner, 2004), (Nedjl et al., 2002). Les interactions entre les agents sont destinées à l'échange de ressources.

Yu et al. (B. Yu, Venkatraman, & Singh, 2003) proposent un SMA homogène dont l'objectif est de s'inspirer des réseaux sociaux humains pour rechercher et délivrer la bonne information aux utilisateurs. Les agents développés dans le prototype MARS ont tous les mêmes objectifs : (i) répondre aux requêtes utilisateurs en délivrant la bonne information, (ii) consulter les autres agents pour compléter leurs informations. Les agents acquièrent ainsi deux qualités : l'expertise, c'est-à-dire la faculté de fournir la bonne réponse, et la sociabilité, c'est-à-dire la faculté de fournir des références précises issues des autres agents. Les agents MARS essaient de percevoir le contexte de l'utilisateur pour satisfaire ses demandes. Cette approche est également adoptée par Plaza dans (E. Plaza & Arcos, 2001) avec la spécification d'agents appelés 'CAPIA' (Context-Aware Personal Information Agents).

2.6.2.2.4 Les SMA conçus avec plusieurs types d'agents

Les SMA composés de plusieurs types d'agents sont qualifiés d'«hétérogènes». Ces agents sont caractérisés par la diversité de leurs objectifs et de leurs compétences. Ces SMA sont adaptés à une modélisation du monde réel car ils peuvent intégrer la complexité des situations d'un environnement dynamique.

Barthès et al. (Barthès & Tacla, 2002), (Tacla, 2003) ont créé un environnement de gestion des connaissances (EGC) dont l'objectif est l'acquisition des connaissances dans un laboratoire de recherche et de développement. L'environnement EGC est composé d'*ontologies* décrivant les objets et les tâches du domaine ; de *mémoires personnelles* qui à partir des éléments employés et de leurs historiques permettent de classer et d'indexer les événements pour chaque activité; d'*un carnet de contacts* élaboré à partir des centres

d'intérêts des utilisateurs ; *de mémoires de projet* composées des éléments (document, mail...), des historiques des tâches générales, des historiques des tâches projet, des caractéristiques et centre d'intérêts des utilisateurs et d'*un processus de recherche* permettant d'interroger les mémoires de projet. L'environnement est géré par un SMA conçu avec quatre types d'agents :

- L'agent *assistant personnel* gère l'interface utilisateur. Il sert de lien de communication entre les utilisateurs et les autres agents.
- L'agent *organisateur* classe les éléments des utilisateurs à l'aide des ontologies. L'agent organisateur communique avec l'agent assistant personnel afin d'échanger les éléments de connaissances.
- L'agent *projet* enrichit la mémoire projet en gérant le processus de préservation. Il emmagasine les caractéristiques des projets (thèmes d'intérêts, compétences, vocabulaire...)
- L'agent *archiviste* est un agent de service permettant l'enregistrement de documents et assurant la recherche documentaire à partir de la plateforme logicielle utilisée par le groupe projet.

Lors du projet CoMMA (Corporate Memory Management through Agents) (Gandon, 2002)(Gandon et al., 2002), un environnement de gestion de connaissances composé d'une ontologie de domaine (Gandon & Dieng-Kuntz, 2005)(O'CoMMA) et d'un SMA dédié permet de développer, structurer et utiliser des mémoires organisationnelles d'entreprise. L'objectif du SMA est de gérer des connaissances distribuées dans une organisation. Il est basé sur une analyse fonctionnelle descendante à partir des groupes, rôles et protocoles. Ces composants sont les plus hauts niveaux d'abstraction du système, ils ont ensuite été complétés à partir des études qui ont permis d'identifier certains types d'agents et de relations. Le système est ainsi constitué de quatre sociétés d'agents : trois sociétés sont affectées aux ressources (ontologie et modèle organisationnel, annotation, interconnexions pages jaunes) et une société est destinée aux utilisateurs. Afin de gérer les sociétés, le système est constitué de neuf agents :

- L'*agent ontologiste* dont le but est de sauvegarder et d'enrichir l'ontologie. Cet agent interagit avec tous les autres agents puisque chaque agent a besoin de l'ontologie pour accomplir ses tâches. Cet agent contrôle l'interface de manipulation des schémas RDF ;
- L'*archiviste modèle organisationnel* sauvegarde et fournit le modèle structurel de l'organisation. Son rôle est de gérer les requêtes et les recherches au niveau du modèle organisationnel. Il contrôle l'interface de manipulation des annotations RDF. Il appartient à la société 'ontologie et modèle organisationnel' ;
- L'*archiviste d'annotation* enregistre et recherche les annotations RDF propre aux documents dans le répertoire local auquel il est associé. Il cherche et extrait les références pour une requête ;
- Le *médiateur d'annotation* distribue les tâches de résolution de requêtes, il répond aux requêtes et propose un service d'archivage ;
- Le *contrôleur d'interface* assure la communication entre les hommes et les autres agents. Sa durée de vie est d'une session, il est créé lorsque l'utilisateur se logue ;
- Le *gestionnaire de profil* réalise l'apprentissage des préférences des thèmes des utilisateurs ;

- L'*archiviste de profils* sauvegarde les profils des utilisateurs lorsqu'ils sont demandés par d'autres agents. Il compare les nouvelles annotations et les nouveaux profils pour détecter les nouveaux documents qui pourraient intéresser un utilisateur ou fournir une nouvelle source d'information ;
- Le *gestionnaire du système agent* maintient les pages blanches où les agents sont enregistrés sous leur nom et adresse ;
- Le *directory facilitator* maintient les pages jaunes où les agents sont enregistrés à partir des services qu'ils fournissent.

Preece et al. (Preece et al., 2000) proposent une architecture multi-agents appelé KRAFT pour la gestion de connaissances provenant de sources multiples et hétérogènes. A la différence des environnements présentés précédemment, dans KRAFT, tout composant est vu comme un agent. Toute connaissance est transcrite dans un langage commun à tous les agents spécifiés dans une ontologie partagée. Ainsi chaque connaissance peut être comparée aux autres même si elle ne provient pas de la même source d'information. L'implémentation des agents est réalisée avec les standards de la FIPA et utilise le langage RDF du web sémantique pour la description des connaissances. Le SMA est composé de quatre types d'agents. Les agents 'User' gèrent les interfaces avec les utilisateurs pour la recherche de connaissances. Les agents 'Wrappers' sont en charge de la capitalisation des connaissances et de la communication du système avec les sources d'information (bases de données, bases de connaissances). Les agents 'Médiators' manipulent (organisent, filtrent, fusionnent...) les connaissances trouvées par les Wrappers. Les agents Facilitators sont en charge de la communication entre les agents particulièrement entre les 'mediators' et les 'user' pour faciliter la recherche de connaissances.

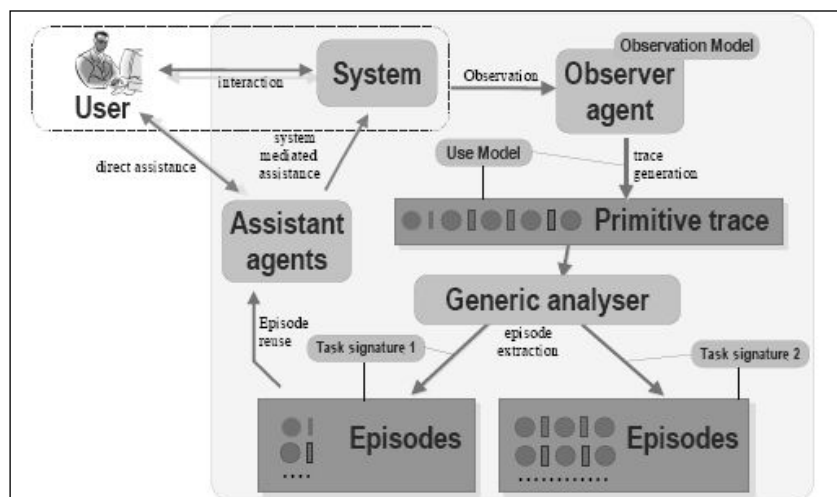


Figure 12. L'environnement MUSETTE

Une approche différente de la gestion des connaissances par les SMA est proposée par Champin et al. (Champin et al., 2003) où le système a pour objectif de capitaliser et partager les expériences et traces d'activités collectives des utilisateurs en fonction de leurs contextes d'utilisation. Le système appelé MUZETTE puis son évolution MAZETTE (Arana, Hassas, & Prié, 2004) utilise un modèle utilisateur décrivant les centres d'intérêts (entités, relations...). Les utilisateurs peuvent ensuite suivre leurs traces d'activités qui sont sauvegardées dans une base de données. Un agent 'Observer' suit les actions des utilisateurs

à partir d'un modèle d'observation. Il crée ensuite une trace d'activité de l'utilisateur. Ensuite un analyseur de traces extrait les traces pertinentes selon les intérêts de l'utilisateur (Figure 12). L'étape suivante est réalisée par l'agent 'Assistant' dont l'objectif est de mettre à disposition les expériences capitalisées.

Plus récemment des travaux ont été réalisés sur la modélisation de communautés de pratique à l'aide du paradigme agent dans l'objectif de faciliter la gestion des connaissances. Le système multi agents KARE (Knowledgeable Agent for Recommendations) (Guizzardi & Perini, 2005), (Guizzardi et al., 2004; Guizzardi, 2006) est réalisé pour simuler les comportements sociaux dans les communautés de pratique. Il a été conçu à partir de la méthodologie TROPOS. Le système est réalisé pour fonctionner sur un réseau Peer-to-Peer. KARE est composé de trois types d'agents (Figure 13).

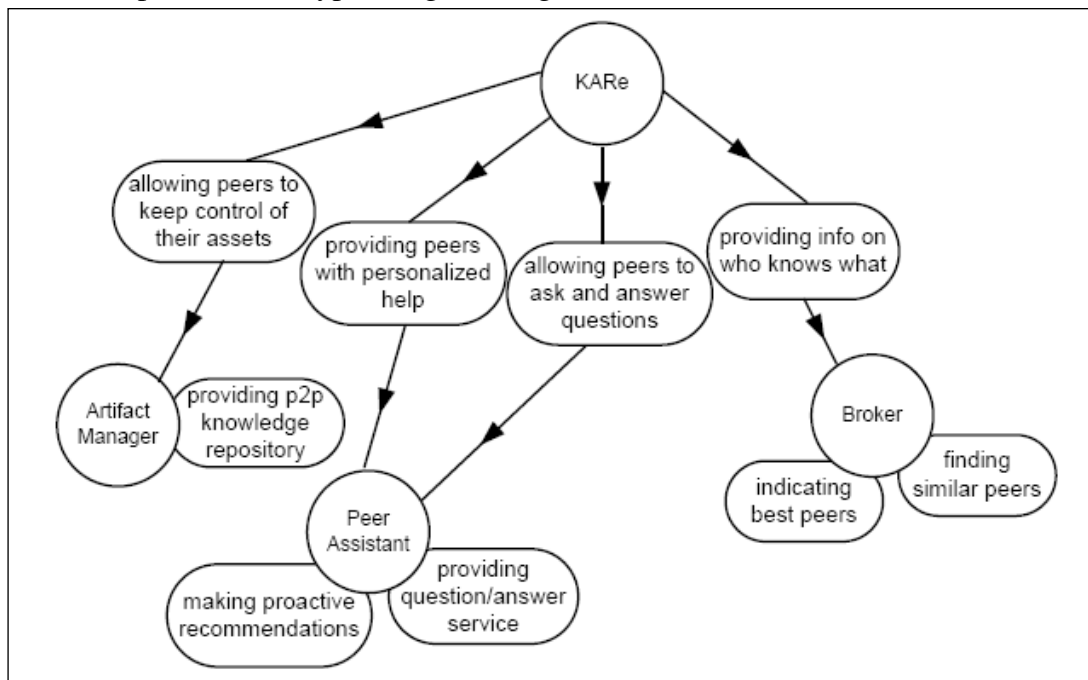


Figure 13. Le système KARE vu à partir de ses objectifs en fonction des agents

L' 'Artifact Manager' est en charge de gérer les connaissances des utilisateurs. Le 'Peer Assistant' est décuplé par le nombre de paires du réseau et a pour fonction la recherche des connaissances auprès des autres Peer Assistants. L'agent 'Broker' scrute les demandes et rend la recherche proactive en associant des paires similaires du réseau. Le 'Broker' est l'entité centrale du réseau, il facilite les communications entre agents. KARE permet de construire l'organisation d'agents à partir d'un réseau Peer-to-Peer où l'architecture de haut niveau peut être modélisée en termes de délégation d'objectifs comme le montre la Figure 13.

2.7 Mesures de similarité entre processus

Une mesure de similarité est utilisée pour comparer différents individus, éléments ou objets de même type (c'est-à-dire réputés comparables). Il s'agit de déterminer une similarité entre deux individus de même type "i" and "j" avec une mesure (ou degré) de similarité notée $d(i, j)$. Cette mesure peut être en relation avec des caractéristiques quantitatives et/ou qualitatives. La valeur de $d(i, j)$ est toujours positive, comprise entre 0 (aucune similarité) et 1 (égalité). Plus la valeur est forte, plus la similarité entre les individus i et j est grande.

De nombreuses recherches ont porté sur la détermination de mesures de similarité pour comparer deux processus distincts.

La plupart sont basées sur l'utilisation d'ontologies du domaine permettant de déterminer une distance entre les activités de ces deux processus ou de comparer la structure de ces deux processus.

Ainsi les activités sont identifiées par un label spécifique qui permettra de les comparer avec une mesure de similarité sémantique, déterminant une proximité entre concepts (Ehrig, Koschmider, & Oberweis, 2007). Une mesure de similarité structurelle peut aussi être défini pour comparer les ordres dans lequel ces activités sont réalisées (Tka & Ghannouchi, 2012).

Un travail de master recherche a été réalisé en 2013 au sein du laboratoire ERPI pour faire un état de l'art sur ces mesures de similarité entre processus (López Vásquez, 2013). Ce travail a proposé un protocole combinant une mesure de similarité sémantique avec une mesure de similarité structurelle.

Une mesure de similarité, dite syntaxique, que l'on pourrait dire basique, vise à comparer les lettres qui composent les mots (concepts) désignant les activités. Dans ce cas, lors de la mesure de similarité de deux concepts, une métrique souvent utilisée concerne le nombre de changement de lettres nécessaires (ajout, suppression, substitution) pour passer d'un mot à un autre (Dongen, van, Dijkman, & Mendling, 2008). Une méthode de calcul d'une mesure de similarité syntaxique a été proposée par (Dongen et al., 2008). Elle calcule le rapport entre le nombre de changement nécessaire pour passer d'un mot A1 à un mot A2 et la longueur du mot le plus long. Si toutes les lettres doivent être modifiées, la similarité est nulle. Si aucune lettre n'est à modifier, il y a égalité et la similarité vaut 1.

Notons Sim_{syn} une mesure de similarité syntaxique. Soit A_n l'un des mots et $L(A_n)$ la longueur de ce mot (nombre de lettres). "ed" désigne le nombre d'opérations d'édition nécessaires pour transformer A1 en A2. Le calcul est le suivant :

$$Sim_{syn} = 1 - \frac{ed(A_1, A_2)}{Max(L(A_1), L(A_2))} \quad (1)$$

Par exemple, pour les mots "inquiry" et "inquiries", il faut 3 opérations d'édition pour transformer ces mots (substituer "y" par "i", et ajouter deux lettres, "e" et "s"). Ainsi la similarité sera de $1 - 3/9 = 0,67$.

Cependant une mesure de similarité syntaxique n'est pas suffisante pour déterminer une proximité entre deux concepts (Sánchez, Solé-Ribalta, Batet, & Serratos, 2012) (Ehrig et al., 2007) car très souvent, les concepts sont désignés par des synonymes et par des mots écrits syntaxiquement très différemment.

Une mesure de similarité dite linguistique permet de tenir compte de la sémantique des mots en se basant sur un lexique (dictionnaire) du domaine. Notons que ce lexique n'est pas toujours disponible ou peut être incomplet.

Dans la littérature, le lexique le plus utilisé est certainement le logiciel libre WordNet qui recense les différentes définitions qu'un mot peut avoir et les regroupant dans des ensembles de synonymes (appelé synset). Ce groupe de mots peut présenter une relation hiérarchique. En fournissant deux concepts comme données d'entrée à WordNet, le logiciel retourne une

valeur numérique qui correspond à une mesure de similarité ou de correspondance entre les deux concepts (Pedersen, Patwardhan, & Michelizzi, 2004).

Cependant quand les désignations des activités sont composées de plusieurs mots qui peuvent être des verbes, des noms et des adverbes, il est nécessaire de tenir compte de la structure du groupe de mots. Une méthode de calcul d'une mesure de similarité linguistique a été proposée par (Ehrig et al., 2007). Elle utilise la fonction $p(A)$ qui recouvre chaque sens de la désignation retenue, à partir de requêtes dans WordNet. Alors on note $S = p(A_1) \cap p(A_2)$ l'ensemble des sens identiques dans la désignation par la comparaison des mots A_1 et A_2 . La fonction binaire $F(S)$ est définie ainsi :

$$F(S) = \begin{cases} 1 & \text{iff } p(A_1) \cap p(A_2) \neq \emptyset \\ 0 & \text{iff } p(A_1) \cap p(A_2) = \emptyset \end{cases}$$

Et la mesure de similarité linguistique est alors calculée par :

$$Sim_{Lin}(A_1, A_2) = \frac{F(S)}{\max(|p(A_1)|, |p(A_2)|)}$$

L'obtention d'une mesure de similarité se basant sur l'ordre des activités est encore plus délicate. Une méthode de calcul d'une mesure de similarité structurelle a été proposée par (Tka & Ghannouchi, 2012). Elle consiste à comparer deux à deux l'ordre dans lequel apparait un couple d'activités dans le premier processus, puis faire de même dans le deuxième processus, ensuite de comparer sémantiquement chaque couple d'activité pour enfin calculer une mesure globale de similarité. Les étapes du calcul peuvent se résumer ainsi :

- Construire la liste des couples d'activités (A_1, A_2) pour le processus P1
- Déterminer la séquence d'apparition de A_1 et A_2 (antériorité, parallélisme,...)
- Construire la liste des couples d'activités (B_1, B_2) pour le processus P2
- Déterminer la séquence d'apparition de B_1 et B_2 (antériorité, parallélisme,...)
- Comparer sémantiquement les couples (A_1, A_2) avec (B_1, B_2)
- Calculer la similarité structurelle comme le rapport entre le nombre de transitions similaires par le nombre total de transitions.

Une analyse des mesures de similarité proposées dans la littérature a été réalisée dans (López Vásquez, 2013).

2.8 Synthèse de l'état de l'art et problématiques

Parmi les types de connaissances manipulées en conception, nous avons distingué des connaissances qui portent sur le produit à concevoir et des connaissances qui portent sur les bonnes pratiques de conception qui peuvent se formaliser sous forme d'une démarche métier.

La capitalisation des connaissances dans les activités de conception est particulièrement difficile car les concepteurs, à la fin d'un projet, ne se souviennent pas complètement des différentes actions qu'ils ont réalisées ainsi que leur logique d'enchaînement (*design rationale*). Les approches habituelles d'explicitation de connaissances (observation, entretien, verbalisation ...) consomment beaucoup de temps et sont critiquées pour leur manque d'exhaustivité. Il manque des travaux à ce jour permettant de conserver l'historique des

actions des concepteurs pour les aider à prendre conscience de l'organisation de leurs activités.

Les travaux réalisés jusqu'à présent sur les traces portent peu sur le processus de conception. A notre connaissance, les aspects collaboratifs et multi-niveaux dans la construction de scénarios ou processus à partir de traces ne sont pas explicitement traités dans ces travaux.

Les SMA ont déjà démontré leur efficacité dans le domaine de la gestion des connaissances pour construire une mémoire organisationnelle à partir de l'annotation des documents générés par les concepteurs et validés par des experts.

L'originalité de notre contribution (Figure 14) réside dans l'intégration des points suivants :

1. développer une méthode permettant d'enrichir progressivement une **mémoire organisationnelle** à partir des expériences capitalisées lors de nouveaux projets.
2. définir une **ontologie** métier permettant d'enrichir la sémantique des traces et de guider la correspondance entre actions réalisées et tâches déduites lors de la construction du processus opérationnel,
3. spécifier un **système multi-agents** permettant de construire une mémoire organisationnelle ou base de connaissances (processus opérationnels, démarches métier) et d'assister les utilisateurs dans des activités futures,
4. utiliser les traces informatiques générées lors des activités collaboratives de conception pour construire, à l'aide d'un **Système à Base de Traces**, le processus opérationnel multi-niveaux.

Les deux premiers points feront l'objet du chapitre 3. Le troisième point sera traité dans le chapitre 4. Le dernier point sera présenté dans le chapitre 5.

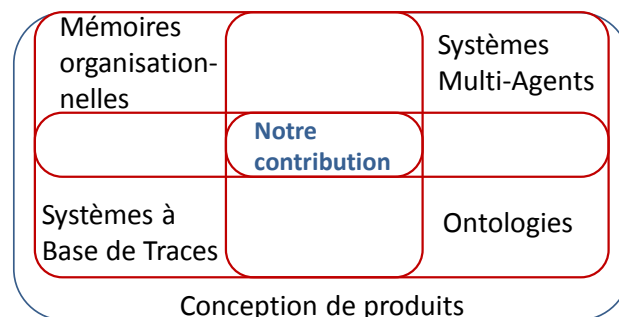


Figure 14. Notre contribution, à l'intersection de différentes thématiques ou techniques d'intelligence artificielle

Les difficultés scientifiques résident à la fois dans l'aspect collaboratif d'un projet de conception et de l'utilisation des outils de la suite TDC, et dans l'aspect multi-niveaux dans la modélisation d'un processus opérationnel.

Les difficultés technologiques concernent le passage d'un démonstrateur à un module intégré dans la suite TDC. L'implémentation logicielle nécessite l'intégration de programmes réalisés avec divers logiciels (programmation en Java avec MadKit pour le développement d'un SMA, programmation en C#, interopérabilité avec un logiciel de visualisation, Yaoqiang) dans les outils de la suite TDC.

Chapitre 3. Vers une approche de capitalisation basée sur les démarches métier

De nombreuses méthodes de capitalisation s'appuient sur le cycle de gestion des connaissances proposé par M. Grundstein (Grundstein & Barthès, 1996), (Grundstein, 2002).

Elles s'inscrivent essentiellement dans le cadre de capitalisation des connaissances d'experts ou d'acteurs métiers participant à des activités de projet. La formalisation des connaissances est souvent fastidieuse à la fin d'un projet puisqu'elle demande de formaliser les connaissances cruciales de l'ensemble des acteurs métier du projet.

Notre travail de recherche se positionne dans un cadre différent. Nous faisons les hypothèses suivantes : les acteurs métiers utilisent un outil numérique qui est le support de leurs activités collaboratives et qui permet d'enregistrer leurs actions élémentaires. Des connaissances cruciales sont incorporées dans l'ensemble de ces actions qui constituent le processus collaboratif réellement mis en œuvre.

La première hypothèse est de plus en plus vérifiée dans les entreprises qui ont instrumenté leurs activités de conception avec des outils numériques.

La seconde hypothèse vient du fait que la nature et l'enchaînement des activités de conception ont une relative invariance pour un certain type de projets ou d'études. Les types de projet ou d'étude peuvent dépendre des pratiques de l'entreprise (formalisées sous forme de processus institutionnalisés ou de référence) ou du lien du type d'étude avec une méthode de conception normalisée comme l'Analyse Fonctionnelle ou l'AMDEC. C'est particulièrement vrai dans le contexte industriel de notre travail de recherche.

Ce chapitre est structuré autour de trois contributions complémentaires que nous proposons dans ce travail de thèse :

- un modèle du double cycle de capitalisation des connaissances générées dans des activités collaboratives instrumentées,
- une structure d'un système de gestion des démarches métier, permettant de supporter et mettre en œuvre ce double cycle,
- une ontologie du domaine étudié recouvrant à la fois une approche organisationnelle (avec les concepts rôles, métiers, processus, etc.) et un modèle de traces (avec les concepts d'outil, d'entité, etc.).

3.1 Vers un double cycle de capitalisation des connaissances

Dans cette section, nous rappelons d'abord le cycle de gestion des connaissances proposé par M. Grundstein, puis nous proposons d'étendre ce cycle en un double cycle dans le cas de l'enrichissement de démarches métier à partir de la construction de processus opérationnel reflétant le déroulement réel d'une étude.

3.1.1 Le cycle de gestion des connaissances

Parmi les travaux sur la gestion des connaissances, le cycle de gestion des connaissances proposé par M. Grundstein (Figure 15) a été largement utilisé en entreprise et en recherche (Grundstein & Barthès, 1996) (Zacklad & Grundstein, 2001) (Grundstein, 2002). Ce cycle met en évidence cinq facettes importantes de la gestion des connaissances, ainsi que des activités et des acteurs différents nécessaires pour la gestion des connaissances.

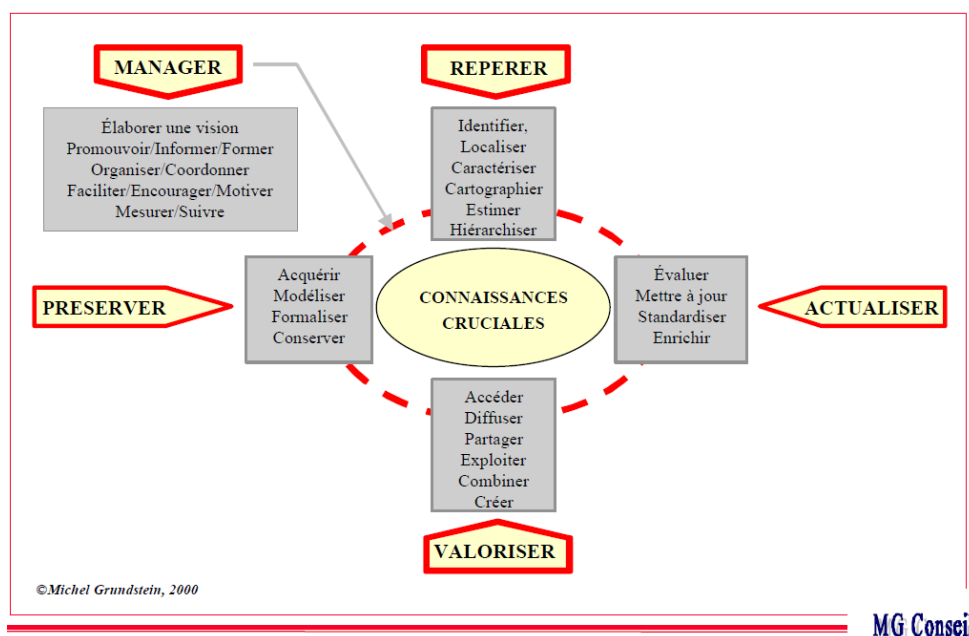


Figure 15. Cycle de gestion des connaissances selon (Grundstein, 2002)

La première facette concerne le repérage des connaissances cruciales, c'est-à-dire les savoirs et savoir-faire qui sont nécessaires au déroulement des processus clés qui constituent le cœur des activités de l'entreprise : il faut les identifier, les localiser, les caractériser, en faire des cartographies, estimer leur valeur économique et les hiérarchiser.

La deuxième facette concerne la préservation des connaissances : il faut les acquérir auprès des porteurs de connaissances, les modéliser, les formaliser et les conserver.

La troisième facette concerne la valorisation des connaissances : il faut les mettre au service du développement et de l'expansion de l'entreprise c'est-à-dire les rendre accessibles selon certaines règles de confidentialité et de sécurité, les diffuser, les partager, les exploiter, les combiner et créer des connaissances nouvelles.

La quatrième facette concerne l'actualisation des connaissances : il faut les évaluer, les mettre à jour, les standardiser et les enrichir au fur et à mesure des retours d'expériences, de la création de connaissances nouvelles et de l'apport de connaissances externes.

La cinquième facette concerne plus généralement ce qui est communément appelé « Management des connaissances » visant à organiser, gérer et amplifier l'utilisation et la création des connaissances dans les organisations. Il s'agit en particulier de sensibiliser et encourager les acteurs concernés, encourager le partage des connaissances, élaborer des indicateurs permettant d'assurer le suivi et la coordination des actions engagées ...

3.1.2 Proposition d'un double cycle de capitalisation des connaissances

Dans cette section, nous proposons d'étendre le cycle de capitalisation traditionnel pour tenir compte de la possibilité offerte par les outils numériques de tracer les actions des utilisateurs et pour bien distinguer la phase de réalisation d'un projet et la phase d'enrichissement des connaissances capitalisées par les métiers.

Rappelons ici la définition des traces proposée dans l'introduction (section 1.3.1)

Les traces d'activité sont des représentations explicites et persistantes des actions réalisées et des informations manipulées par les acteurs métier lorsqu'ils accomplissent leurs activités de conception.

Ainsi une trace est l'enregistrement d'une action d'un utilisateur, instrumentée par un outil et portant sur une entité particulière, à une date particulière.

Nous appelons processus opérationnel, l'enchaînement logico-temporelle des activités, réalisées au cours d'un projet particulier.

Nous appelons démarche métier l'enchaînement logico-temporelle d'activités (ou scénario), jugées pertinentes et préconisées pour la réalisation efficace d'une étude correspondant à un certain type d'études.

Dans le contexte de notre travail de recherche, nous distinguons deux cycles centrés vers les métiers ou vers le projet :

- un cycle pour les « démarches métiers » qui sont relatives à des classes de projets passés semblables (classe d'expériences) et qui sont capitalisées dans les métiers,
- un deuxième pour les « processus opérationnels », où chaque processus opérationnel a été mis en œuvre réellement dans un projet particulier (à chaque projet réalisé, une nouvelle expérience est générée).

L'initialisation de ce dispositif est réalisée par l'expert métier et concerne le cycle concernant les démarches métiers. Il va ACQUERIR et FORMALISER les premières versions des démarches métiers à partir de ses expériences passées (contenant souvent de nombreuses connaissances tacites), de celles de ses collaborateurs et/ou de connaissances issues de normes d'ingénierie préconisant des activités génériques de conception portant sur des objets intermédiaires ou finaux. Le formalisme recommandé proviendra de la norme de

modélisation BPMN car il permet de partager la même sémantique entre les différents acteurs métiers et il correspond à un standard dans les outils de visualisation des processus.

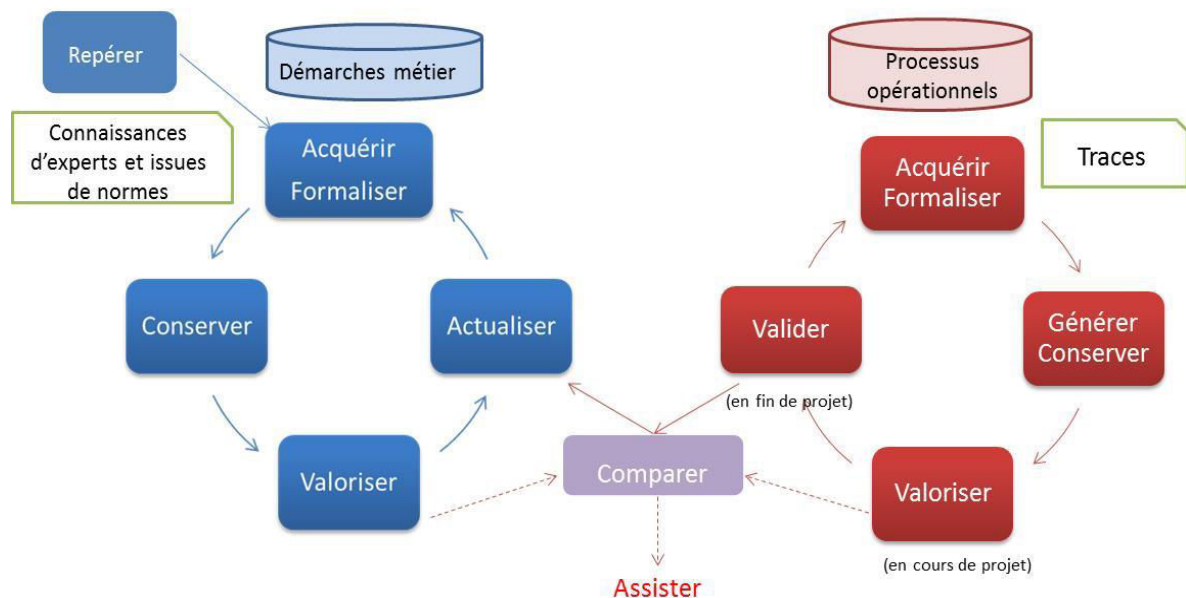


Figure 16. Double cycle de capitalisation des connaissances pour des activités instrumentées

Un système à base de connaissances va permettre de CONSERVER les démarches métiers. Ce système va ensuite les VALORISER en les mettant à disposition des utilisateurs, soit en consultation, soit en assistance.

Les experts qui peuvent définir et valider les démarches métiers peuvent varier selon le champ couvert par une démarche métier et la politique de l'entreprise :

- L'expert d'un métier lorsque la démarche métier ne concerne qu'un métier,
- Un expert KM qui peut être le garant de la mise à jour des démarches métiers et/ou de l'évolution de l'ontologie des démarches métiers. Dans ce dernier cas, une compétence particulière en informatique peut être requise pour faire évoluer l'ontologie.
- Un groupe d'experts lorsque la démarche métier nécessite la collaboration de plusieurs métiers ou lorsque la démarche métier est relative à une méthode de conception transversale à plusieurs métiers (comme l'AMDEC).
- Les experts conseil de la société qui diffuse l'outil numérique utilisé. Il pourra initialiser les démarches métier de l'entreprise et aider l'entreprise à faire évoluer l'ontologie métier en fonction des besoins propres de l'entreprise concernée.

Par la suite, nous désignerons cet expert ou ce groupe d'experts, par « expert méthodologique ».

Pour ACTUALISER une démarche métier, l'expert méthodologique pourra utiliser l'expérience provenant d'un nouveau projet une fois terminé. La problématique majeure réside habituellement dans l'oubli des tâches réalisées ainsi que leur enchaînement, quand arrive l'heure du bilan d'un projet. Peu d'ingénieurs prennent le temps d'enregistrer ou de

tracer les tâches qu'ils ont réalisées et de les lier aux résultats obtenus. Nous avons relevé lors de l'étude bibliographique que les entretiens avec les experts sont longs et ne permettent pas de capitaliser de façon fiable sur le processus opérationnel réellement mis en œuvre.

Dans le cas d'activités de conception outillées (ou instrumentalisées), il est envisageable de tracer et de mémoriser les actions réalisées sur l'outil afin de construire le processus opérationnel. Ce processus opérationnel pourra alors être utilisé pour une éventuelle actualisation de la démarche métier correspondante. Il reviendra à l'expert méthodologique de VALIDER l'enrichissement de la démarche métier.

Un système à base de traces va permettre d'ACQUERIR des données correspondant aux actions réalisées par les utilisateurs (ou acteurs métiers) de l'outil. Une ontologie peut FORMALISER la correspondance entre les actions dans les traces et les tâches des processus opérationnels. Un ensemble de règles de transformation des traces permet de GENERER (ou construire) le processus opérationnel mis en œuvre dans le cadre du projet. Il peut être alors CONSERVER dans un formalisme semblable à celui de la démarche métier.

En cours de projet, il pourra être VALORISE, en étant consulté par des acteurs métiers.

Cependant, tant qu'il n'a pas été validé par l'expert méthodologique concerné, le processus opérationnel a un statut de donnée. En fin de projet, il prend un statut de « connaissance » quand il est ACTUALISE et VALIDE par l'expert métier. Celui-ci peut alors l'annoter, l'enrichir avant de le versionner et de le CONSERVER.

Un point clé de ce dispositif de retour d'expérience concerne une facette qui n'a pas été identifiée par Grundstein, à savoir COMPARER. Deux approches peuvent co-exister :

- en cours de projet, pour fournir une assistance au concepteur,
- à la fin du projet, pour aider à enrichir la démarche métier.

En cours de projet, la comparaison peut être servir de base à une assistance pour guider le concepteur avec la démarche métier préconisée en fonction de l'avancement dans le processus opérationnel. Cette comparaison peut être visuelle, réalisée par le concepteur. Elle pourra être automatisée et fournir une aide proactive permettant après chaque tâche de proposer au concepteur la réalisation d'une nouvelle tâche suivant la démarche métier concernée.

A la fin du projet, la comparaison peut aider à enrichir la démarche métier en fonction du processus opérationnel validé (qui peut être considéré comme l'expérience du projet terminé). Cette comparaison peut être visuelle, réalisée par l'expert méthodologique. Elle pourra être automatisée. Si des écarts sont constatés, l'expert méthodologique pourra décider de modifier, de compléter ou d'annoter la démarche métier. Cette activité revient aussi à utiliser une nouvelle expérience, pour la comparer avec d'autres expériences similaires et pour en tirer de nouvelles leçons qui permettront d'enrichir la démarche métier. Les modifications de la démarche métier peuvent concerner :

- l'enchaînement de tâches et d'activités, ou la répétition d'une (ou plusieurs) tâche(s), par exemple, pour représenter un travail progressif et itératif,

- l'ajout d'une nouvelle tâche dans la démarche métier en cas d'évolution de l'outil méthodologique, ici, le logiciel TDC (l'ontologie devra alors évoluer pour assurer la correspondance).

Dans la section suivante, nous proposons une structure globale d'un système de gestion des démarches métiers à partir de la construction de processus opérationnels.

3.2 Approche globale de gestion des démarches métiers

Comme nous l'avons expliqué précédemment, les connaissances liées à la démarche métier mise en place dans un projet de conception doivent être gérées afin d'améliorer l'efficacité du processus de conception dans un contexte d'ingénierie collaborative et simultanée.

Dans cette section, nous présentons une approche globale de gestion des démarches métier qui pourra servir de base à un système à base de connaissances. Elle s'inspire de deux types d'approches : modélisation des processus de conception et réutilisation des connaissances métiers.

Le premier type d'approches a pour but de modéliser et caractériser les processus pour permettre aux entreprises de développer leurs produits de manière plus efficace. Plusieurs travaux (Mantelet, 2006; Mougenot, 2008; Ten Bosch, Van der Wolf, & Van der Hoeven, 1995) abordent la représentation du processus de conception dans le cadre d'une application informatique destinée à guider les concepteurs tout au long de leur projet : à partir de modèles de processus prédéfinis, adaptables en cours de projet, les acteurs archivent leurs activités, les outils employés, mais aussi les contraintes de conception, en vue d'une gestion multi-vues.

Pour le second type d'approches, des travaux récents visent à faciliter la gestion et la réutilisation de connaissances issues de métiers différents (Lahoud, Monticolo, & Gomes, 2010) (Serrafero, Gomes, Bonnivard, & Jézéquel, 2006).

En utilisant le premier type d'approche, nous souhaitons permettre aux experts d'identifier les connaissances pertinentes à capitaliser dans les démarches métiers de l'entreprise. A partir de cette identification, en utilisant le deuxième type d'approches, nous pouvons les structurer et proposer une structure des concepts du domaine. Cette démarche ascendante aboutit à la définition d'une ontologie permettant de représenter les démarches métiers. Cette approche est complétée par une définition des concepts, de leurs relations et attributs constituant l'ontologie du domaine des processus de conception. La sémantique et le vocabulaire, décrits dans l'ontologie, permettent au système de gestion des connaissances de structurer les données enregistrées et de supporter des processus d'inférence. Cette ontologie sera détaillée dans la section 3.3.

Un système à base de traces génère le processus opérationnel du projet de conception en utilisant l'ontologie et des règles de transformation. Une mesure de similarité entre le processus opérationnel et la démarche métier peut permettre de générer les écarts.

Notre approche s'appuie sur la conception d'un système intelligent capable de gérer des démarches métiers à travers les outils méthodologiques utilisés. Les services attendus sont les suivants :

- Générer et conserver le processus opérationnel correspondant à une étude particulière.

- Valoriser les processus opérationnels par leur consultation par les acteurs métiers.
- Valider les processus opérationnels à la fin d'une étude.
- Formaliser et conserver les démarches métiers créées par les experts.
- Valoriser les démarches métier par leur consultation ou leur réutilisation pour fournir une assistance proactive à un acteur métier.
- Actualiser (enrichir) une démarche métier grâce à l'expérience acquise dans une étude de même type (calcul des écarts entre le processus opérationnel et la démarche métier).

La Figure 17 illustre notre approche.

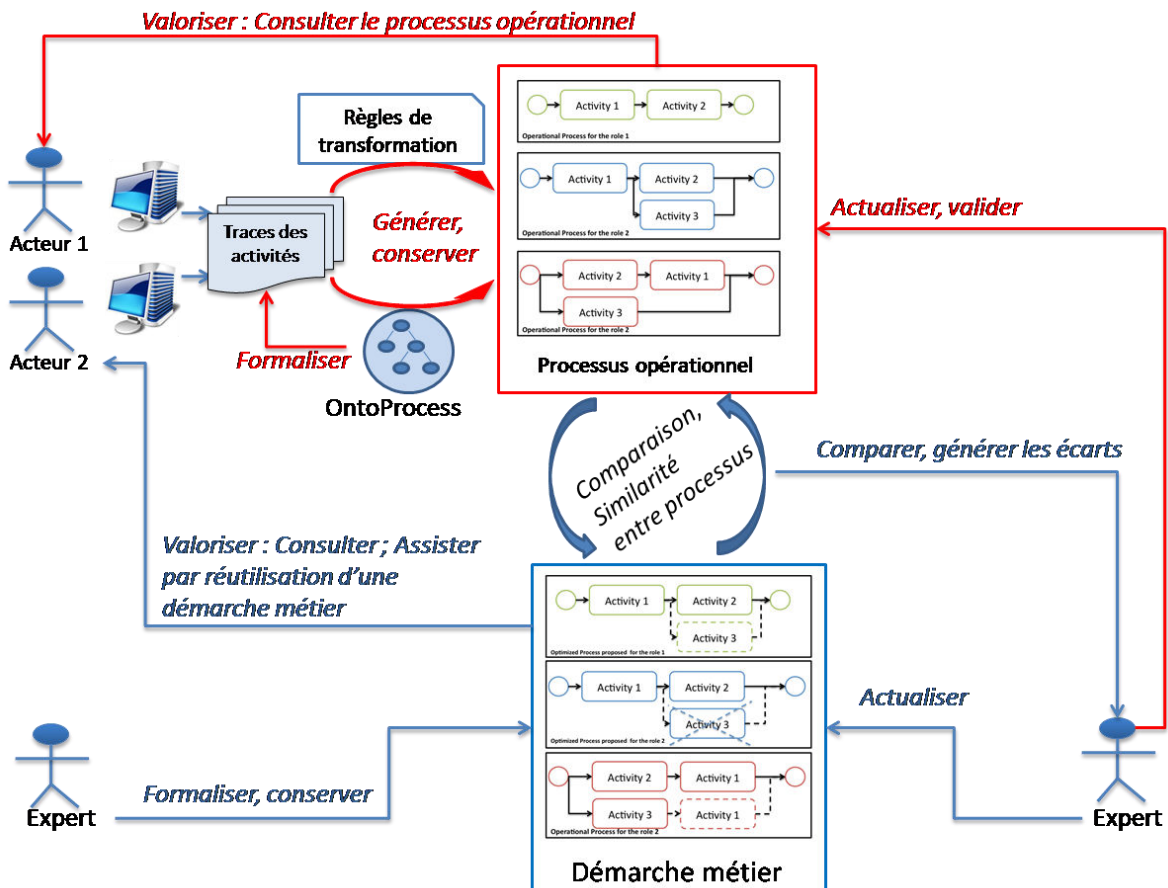


Figure 17. Approche globale de gestion des démarches métier

La modélisation de la connaissance est une étape fondamentale pour assurer le bon déroulement des cycles de capitalisation des connaissances. Cela consiste à identifier, interpréter et structurer les connaissances pour pouvoir ensuite les formaliser, sauvegarder, partager et manipuler. Le choix du formalisme à utiliser pour la structuration et la représentation de connaissances doit permettre de capturer explicitement la sémantique de ces connaissances.

Notre choix s'est porté sur une représentation basée sur une ontologie parce que l'ontologie permet de construire un modèle des connaissances et peut supporter des mécanismes d'inférence sur ces connaissances.

Dans le paragraphe suivant, nous allons présenter l'ontologie OntoProcess qui prend en compte le contexte organisationnel et identifie les concepts du domaine ainsi que leurs relations.

3.3 Ontologie du domaine des démarches métier et des processus opérationnels : OntoProcess

Notre approche de gestion des connaissances s'appuie sur une ontologie des démarches métier et des processus opérationnels. Nous proposons de construire une ontologie de domaine pour définir une terminologie et des relations permettant de représenter ces connaissances. L'ontologie représente un vocabulaire consensuel qui définit le sens des concepts et des relations entre ces concepts. Ce vocabulaire peut être associé à un modèle qui décrit le contenu d'une base de démarche métier.

3.3.1 Etape de construction de l'ontologie OntoProcess

Dans cette section, nous expliquons brièvement comment nous avons construit l'ontologie OntoProcess.

La conception d'OntoProcess suit les instructions de (Gandon, 2002) basée sur la comparaison des éléments des méthodologies de conception réalisée par (Fernandez, Gomez-Perez, & Juristo, 1997) et (Gomez-Perez, Fernandez, & De Vicente, 1996) d'une part et le travail sur les fondations théoriques des ontologies de (Bachimont, 2000) et (Kassel, 2002).

Les étapes vers la conception de l'ontologie (Gandon, 2002) sont :

- L'analyse des scénarios d'application : les scénarios sont des descriptions textuelles des activités et des interactions concernant les cas d'utilisation d'une application. L'analyse des scénarios consiste à identifier les objectifs, les limites, les usages et les utilisateurs de l'application. Il en résulte le document de spécifications des exigences qui décrit l'usage et l'utilité du futur système conçu et implémenté. Ici, les scénarios sont illustrés dans le double cycle de gestion de connaissances que nous avons présenté ci-dessus.
- Les scénarios sont ensuite utilisés pour introduire les questions de compétences : ces questions spécifient les exigences de l'ontologie.
- Analyse et collection des données par l'intermédiaire d'entretiens structurés ou non structurés, par observations, par brainstorming, par questionnaires, par analyse informelle (identification des concepts) ou formelle (identification des attributs et relations entre les concepts) des textes, analyse des traces.
- Etude linguistique et association sémantique : définition des concepts identifiés dans la phase de collecte de données. La définition des concepts peut être incluse dans les scénarios de l'application déjà déterminés.
- Conceptualisation et association ontologique : passage de la définition de sens des concepts à la formalisation des concepts. Cela consiste en une organisation des concepts sémantiquement similaires et en l'établissement de leurs relations.

- Définition de la taxonomie : le rôle principal de la taxonomie est de faciliter la compréhension de l'ontologie. La taxonomie est une classification basée sur les similarités. Elle présente les trois types d'inférences : classification / identification, généralisation / spécialisation (généralisation = catégorisation). Il existe trois approches pour la population et la structuration de l'ontologie sous forme de taxonomie :
 - L'approche Top-Down où on détermine les concepts du haut niveau et ensuite on les spécialise,
 - L'approche Middle-Out où l'on détermine les concepts noyaux et ensuite on les spécialise et les généralise.
 - L'approche Bottom-Up où l'on détermine au début les concepts du niveau de taxonomie le plus bas et ensuite on les généralise.

Ces approches sont basées sur la définition des critères qui distinguent ou rassemblent les concepts.

- Formalisation et opérationnalisation des concepts : La taxonomie est une représentation formelle mais son expressivité n'est pas suffisante pour la représentation de l'ontologie. La version informelle ne disparaît pas après la formalisation. En effet, l'ontologie doit être intelligible autant par l'humain que par la machine.

3.3.2 Modèle conceptuel correspondant à OntoProcess

Notre approche de conception de l'ontologie OntoProcess est basée sur une étape d'analyse des ontologies existantes du domaine.

Les concepts suivants sont fréquemment utilisés : métier, projet, acteur, rôle, activité.

Dans notre travail, nous parlerons d'étude dans un sens similaire à celui de projet : la réalisation d'une étude pourra se faire dans le cadre d'un projet. Cependant toutes les études ne seront pas forcément menées sous forme de projet. Le concept d'étude est donc le plus générique ici.

Notre ontologie est composée de trois parties permettant de décrire :

- le métier,
- la structuration préconisée d'une étude
- et la réalisation d'une étude.

Le Métier

Le modèle conceptuel de la Figure 18 décrit les concepts clés pour décrire les métiers.

Deux types d'« Acteur » peuvent être distingués : l'« Acteur Métier » et l'« Expert Méthodologique ».

Un « Département » est constitué d'« Acteurs Métier ». Chaque « Acteur métier » est rattaché, à un moment donné, à un seul métier qui correspond à un corpus de connaissances et de compétences semblables.

L'« Expert Méthodologique » n'est pas rattaché directement à un métier car il doit avoir une vision transversale du projet et une connaissance de l'implication de différents « Métiers » dans le projet de développement. L'« Expert Méthodologique » formalise les

expériences liées à ces « Métiers ». Il doit maîtriser les outils utilisés et les méthodes adaptées (lien volontairement non représenté entre « expert méthodologique » et « outil »).

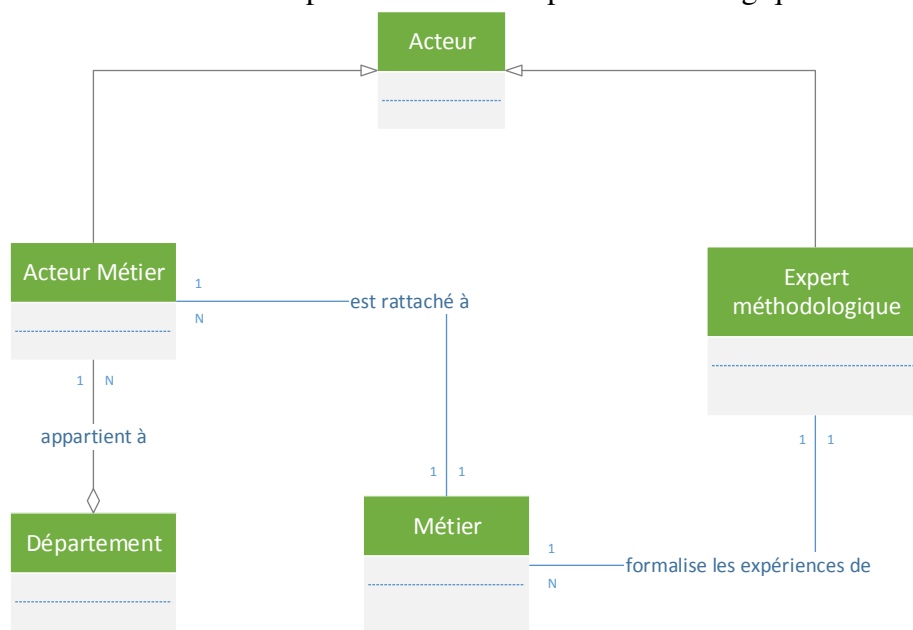


Figure 18. Modèle conceptuel du point de vue des métiers

La structuration préconisée d'une étude

Le modèle conceptuel de la Figure 19 présente les concepts clés pour comprendre la structuration d'un projet. La structuration préconisée du projet décrit les bonnes pratiques de développement relatives à certains types d'études. Dans le cadre de ce mémoire, nous rappelons ici la définition d'une « Démarche Métier » un enchaînement logico-temporelle d'« Activités » (ou scénario), pertinent pour la réalisation efficace d'une « Etude » correspondant à un certain type d'études.

Exemple d'étude : Faire la conception préliminaire d'un produit X, en réalisant les activités suivantes : Faire l'Analyse Fonctionnelle d'un produit X, définir la structure organique d'un produit X, faire l'AMDEC produit X.

Une étude du type « faire la conception préliminaire d'un produit Y » peut utiliser la même démarche métier que pour celle du produit X.

Pour une meilleure visibilité, plusieurs niveaux de décomposition d'une démarche métier sont envisagés : activité, tâche.

Une « activité » peut se décomposer en activités de plus bas niveau et en « tâches ». La tâche sera le niveau élémentaire de décomposition. C'est une activité que l'on ne peut plus décomposer.

Exemple de décomposition d'activité : l'activité « Faire l'Analyse Fonctionnelle d'un produit » se décompose en activités : rechercher les fonctions, caractériser les fonctions ... Rechercher les fonctions n'est pas décomposable : c'est une tâche. L'activité « Caractériser les fonctions » peut se décomposer à son tour : ce n'est pas une tâche.

Une activité est dite générique si elle n'est pas rattachée à un métier. Une activité est dite spécifique si elle est instanciée pour un métier donné.

Exemple : l'activité « Définir l'architecture organique d'un produit » est générique, les activités « Définir l'architecture organique d'un produit pour la partie électrique » ou « Définir l'architecture organique d'un produit pour la partie mécanique » sont spécifiques.

Un « rôle » est une abstraction d'un comportement dans un contexte précis et confère un statut dans l'organisation (Rodriguez 2005). Généralement, un « rôle » est responsable de plusieurs « activités ». Un rôle est relié indirectement à un métier.

Exemples de rôle relié à un métier : Spécificateur électrique, spécificateur mécanique, Chargé d'études électriques, chargé études de risques process injection ...

Notons que nous aurions pu définir le concept de rôle générique pour typer les rôles métier, par exemple, spécificateur ou concepteur. Cependant, ce concept n'est pas apparu comme étant utile dans notre étude.

Dans une approche organisationnelle, des rôles différents peuvent collaborer pour réaliser une étude donnée en suivant les préconisations d'une « Démarche métier ».

La « Démarche métier » détermine comment les « Activités » sont affectées aux « Rôles ». Une « Etude » selon son type requiert une certaine « Démarche métier ». La « Démarche métier » décrit ainsi l'enchaînement des activités préconisé ainsi que les allocations activité/rôle recommandées.

Exemple : le rôle « spécificateur électrique » appartient au métier « ingénierie électrique » rattaché au département « Bureau d'études électriques » et est responsable de l'activité « faire l'Analyse Fonctionnelle électrique ».

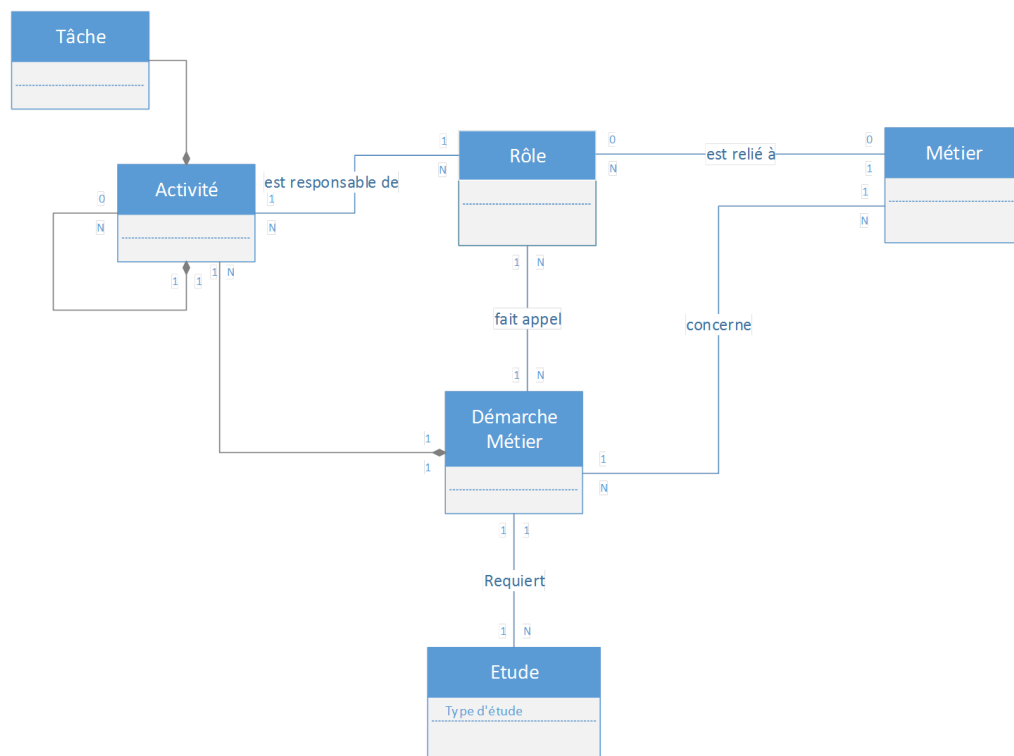


Figure 19. Modèle conceptuel du point de vue étude

La démarche métier d'un type d'études peut servir de schéma directeur pour planifier le projet correspondant à cette étude. Elle peut aussi permettre de guider les acteurs métier au cours d'une étude.

Exemple : un projet de développement complet d'un nouveau produit est composé de différentes phases (par exemple, étude de faisabilité, conception préliminaire, conception détaillée, industrialisation), chaque phase appelant les activités de type « faire une analyse fonctionnelle », « définir la structure organique », « faire l'AMDEC produit » Selon le choix de l'entreprise, différentes stratégies d'identification des démarches métier sont envisageables : soit une démarche métier est associée à l'ensemble de ce type de projet (considérée alors comme la réalisation d'une longue étude), soit une démarche métier est définie pour chaque phase. Un projet d'amélioration d'un process existant peut concerner une étude de type « faire une AMDEC Process d'un produit existant ».

Lien entre la structuration préconisée de l'étude et le métier

L'« Acteur métier » peut posséder plusieurs « rôles » au cours d'une étude. Le rôle donne à l'acteur métier le droit de réaliser les activités sous responsabilité de ce rôle.

Une étude de développement d'un produit fait intervenir des acteurs maîtrisant des métiers différents (ingénierie mécanique, ingénierie électronique, architecture système, ergonomie, etc.). La démarche métier décrit le processus d'implication de ces acteurs dans l'étude collaborative.

C'est l'expert méthodologique qui est responsable de définir, d'évaluer, de valider et de mettre à jour les démarches métier de l'entreprise. Il doit maîtriser les outils utilisés, pour pouvoir définir et valider les démarches métiers. Et à cet égard, il se base sur ses expériences, les stratégies de l'entreprise, les normes et les méthodologies de conception adoptées, et aussi les conseils provenant des consultants travaillant pour les éditeurs des outils utilisées au sein de l'entreprise.

La modélisation d'une démarche métier est initialisée par un expert méthodologique à partir de sa propre expérience ou ses connaissances de l'état de l'art (par exemple, guide de conception au sein de l'entreprise, normes sur l'Analyse Fonctionnelle, sur l'AMDEC ...). Cette modélisation sera ensuite enrichie grâce à notre système intelligent après l'achèvement de chaque nouvelle étude, validée par l'expert méthodologique responsable de cette démarche métier et enfin versionnée.

La réalisation d'une étude

La troisième partie du modèle conceptuel (Figure 20) identifie les concepts relatifs à la réalisation d'une étude.

L'acteur métier, en fonction de son rôle, réalise un ensemble d'« Actions » de conception qui sont supportées par un « Module » de l'« Outil » concerné.

Dans notre cas, l'outil correspond à la suite TDC et un module peut être TDC Need, TDC structure ou TDC FMEA.

Les « Actions » créent, modifient ou suppriment des « Entités ».

Exemple d'actions et d'entités : « Modifier le niveau », « créer un composant » sont des actions possibles portant respectivement sur une entité « niveau » et sur une entité « composant ».

Nous avons introduit dans notre modèle la notion de tâche qui nous a permis de faire le lien entre la description du réel (« Action ») et le requis (« tâche »). Elle permet de passer des

actions aux activités. Une tâche correspond à une action informatique de type créer, modifier ou supprimer, portant sur une entité.

Le « processus opérationnel » est construit à partir de l' « historique de traces ». Il est composé des enchaînements des activités réellement produites. Le « processus opérationnel » passe de l'état de données à l'état de connaissances après la « validation » par l'expert méthodologique.

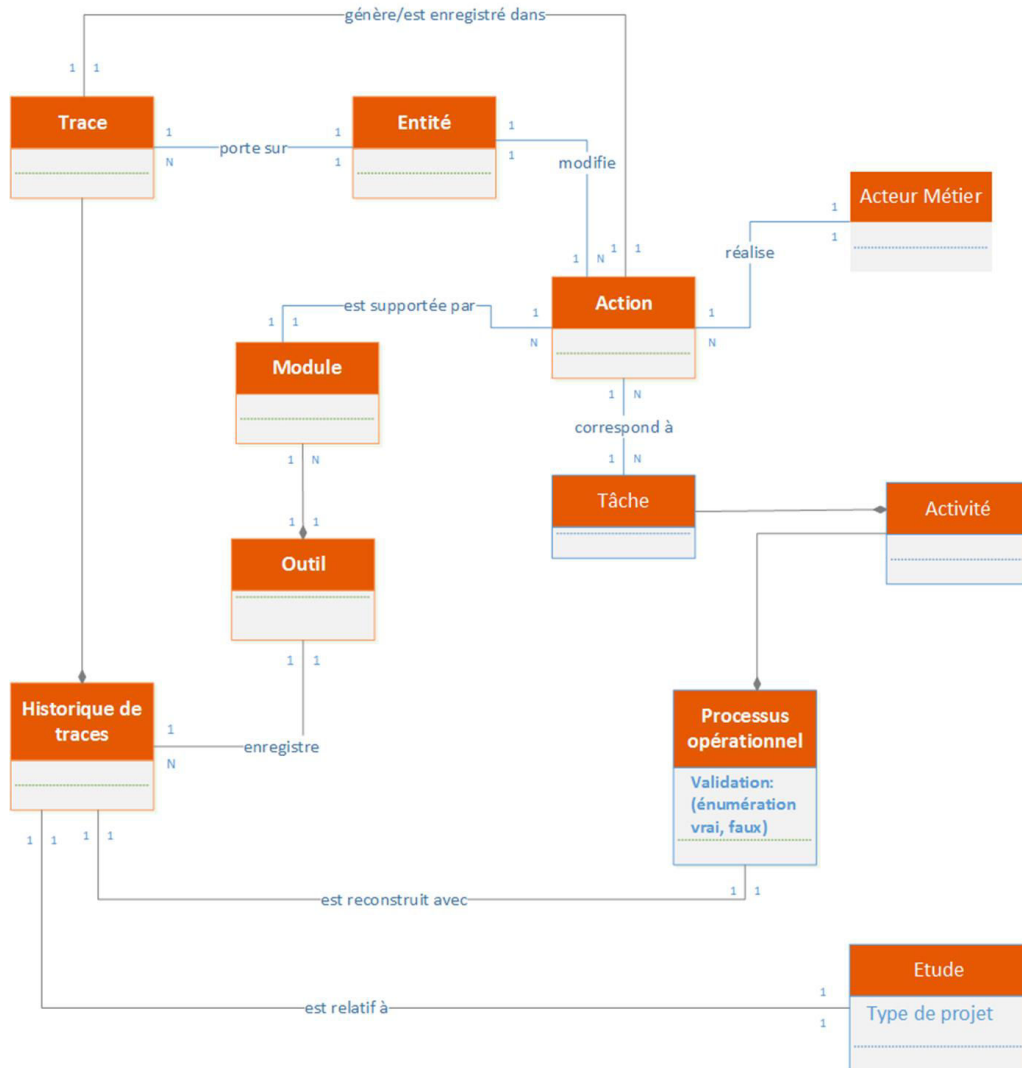


Figure 20. Modèle conceptuel de la réalisation d'une étude

Lien entre la structuration préconisée d'une étude et sa réalisation

Une « action » correspond à une « tâche » que l'on attend que l' « acteur métier » réalise. Nous faisons l'hypothèse qu'un rôle utilise un seul module méthodologique pour réaliser les activités dont il est responsable. Un « acteur métier » peut posséder différents rôles au cours d'une étude.

Exemple : Jean, ingénieur électrique, du département « Bureau d'études électrique », a deux rôles dans une étude : « spécificateur électrique » et « chargé d'études électriques ». Il doit réaliser les activités correspondantes en utilisant, respectivement, le module TDC Need

et le module TDC structure. En prenant le rôle « chargé d'études électriques », Jean a accès au module TDC structure où il peut définir l'architecture organique du produit.

Le module TDC structure lui donne la possibilité de modifier certaines entités de l'Analyse Fonctionnelle. Cependant, cette utilisation devra être très limitée car cela peut poser des problèmes de responsabilité d'un rôle par rapport à une activité, et par la suite, introduire des incohérences dans le cahier des charges fonctionnelles qui ne serait pas détectée.

Lien entre la réalisation d'une étude et le Métier

L' « Expert Méthodologique » définit et valide les démarches métiers. Après analyse d'un nouveau processus opérationnel et des écarts avec la démarche métier suivie, il peut décider de mettre à jour et enrichir la « démarche métier » concernée.

Vue globale de l'ontologie OntoProcess

La Figure 21 présente une vue globale de l'ontologie OntoProcess, à l'aide d'un diagramme de classes UML. Notons qu'à ce niveau de modélisation, nous devrions parler encore de modèle conceptuel.

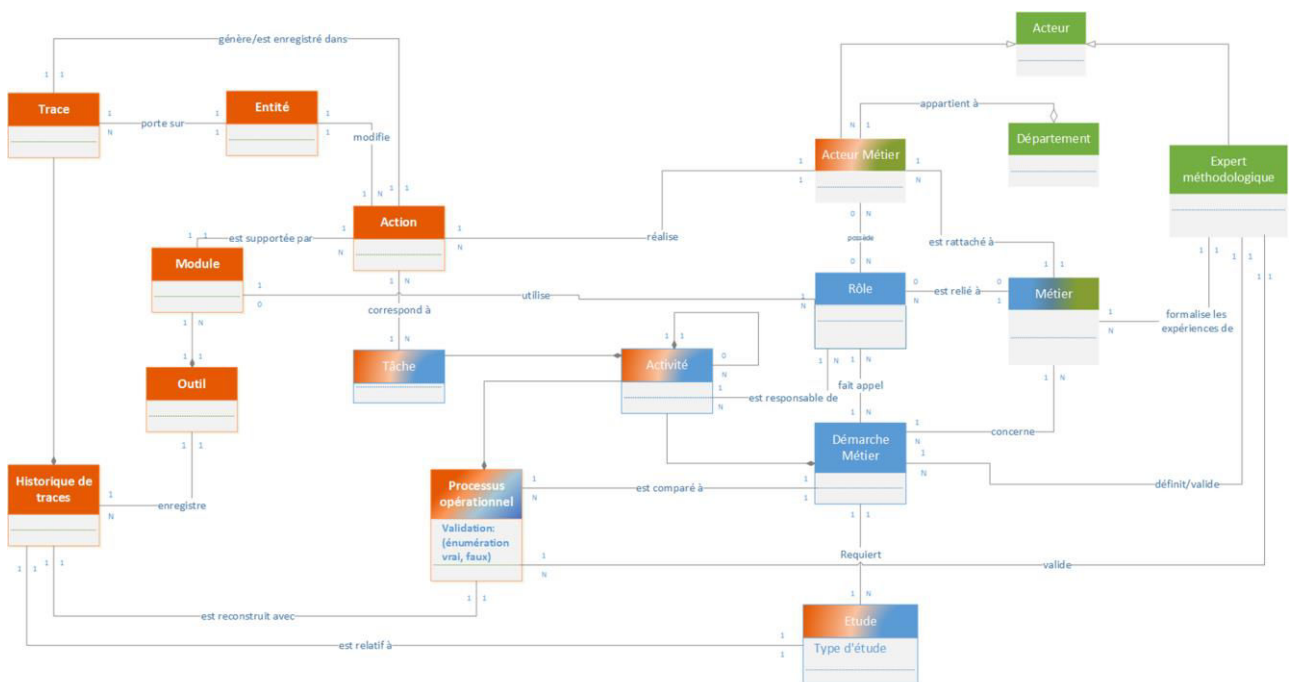


Figure 21. Vue globale de du modèle conceptuel OntoProcess

Autres concepts liés au management de projet

Ces concepts ne sont pas représentés dans ce modèle conceptuel car ils sont liés au management de projets.

Projet : engagement temporaire visant à créer un produit, un service ou un résultat.

Classe de projets : Les projets qui sont menés au sein des services Recherche et Développement d'une entreprise peuvent être regroupés par classe. En effet, des invariants peuvent être identifiés entre ces projets considérés comme semblables par les managers ou experts : même décomposition en phase, mêmes processus de conception mis en œuvre, mêmes rôles concernés... Deux projets de développement d'un nouveau produit sont relativement semblables et appartiennent à la même classe de projets. Mais un projet de développement est mené différemment d'un projet AMDEC Process d'un produit existant. Ils n'appartiennent pas à la même classe de projets.

Phase de projet : Une phase est un regroupement des activités entre deux jalons qui se caractérise par la production d'un livrable, un résultat tangible dont l'évaluation constitue un point de décision sur l'avancement du projet (par exemple, décisions de reprise de quelques tâches de la phase, de passage à la phase suivante, de réorientation du projet, etc.).

Revue de projet : Les fins de phase constituent les jalons sur le déroulement du projet, et donc les points de décisions stratégiques. Les revues de projet ont pour but de fournir au management une vision objective de l'état du projet. Elles consistent à analyser les résultats du projet (qui sont des entrées pour la phase suivante) et fournir au management les éléments de décision). Au niveau temporel, il est préférable d'avoir une séquence entre les phases (le projet progresse).

3.3.3 Extension d'OntoProcess relative aux activités et entités de conception spécifiques à un domaine

Chaque entreprise peut posséder son propre vocabulaire concernant les activités à réaliser dans ses différentes études. Cependant, l'introduction d'un outil supportant ces activités peut permettre de standardiser ce vocabulaire, en particulier, lorsque cet outil a lui-même été conçu pour être le support à des méthodes de conception normalisées, par exemple, l'AMDEC ou l'Analyse Fonctionnelle.

Dans ce travail de thèse en collaboration avec la société TDC, la suite méthodologique commercialisée par l'entreprise supporte principalement quatre méthodes : l'Analyse fonctionnelle externe, l'analyse fonctionnelle interne (ou analyse structurelle), l'AMDEC produit et l'AMDEC process. Les concepts manipulés par cet outil sont conformes à ceux qui sont décrits dans les normes décrivant ces méthodes.

Ainsi dans la suite de ce travail, nous pouvons assurer que les extensions de l'ontologie OntoProcess que nous proposons conservent un caractère générique. Ils vont concerner des taxonomies d'activités permettant de décrire la décomposition des activités jusqu'au niveau des tâches. Les entités manipulées par les tâches et faisant l'objet d'actions qui les modifient sont identifiées et vont permettre d'inférer les tâches réalisées à partir des traces enregistrées. Les étapes de la construction d'un processus opérationnel, reposant en partie sur l'ontologie OntoProcess, seront décrites au Chapitre 5.

3.3.3.1 Principes d'utilisation d'OntoProcess pour faire le lien entre action, entité, trace et tâche

Dans cette section, nous souhaitons présenter les principes d'utilisation d'OntoProcess pour transformer des traces en processus opérationnels, à partir d'un petit exemple. Cela nous

permettra de mieux comprendre les éléments d'OntoProcess dont nous avons besoin pour réaliser la transformation.

Dans la Figure 22, on distingue quatre parties qui sont inter-reliées :

- En haut, sur la gauche, c'est un extrait de l'ontologie des traces et sur la droite, un extrait de l'ontologie des processus opérationnels.
- En bas, sur la gauche, ce sont les instances de traces correspondant aux classes de l'ontologie, et sur la droite, les instances de processus.

Comme le définit l'ontologie des traces, une action porte sur une entité de conception (par exemple, phase de vie, fonction de service, etc.). L'entité est identifiée par son ID. L'action a plusieurs attributs : sa date de réalisation, la valeur qu'elle attribue à l'entité, l'acteur métier qui la réalise, le module dans lequel elle est réalisée, et son type (MAJ, etc.)

Dans l'exemple des instances des traces, il y a deux entités : une phase d'ID 570 et une fonction de service d'ID 571. La fonction de service 571 appartient à la phase 570. Une action de type MAJ attribue la valeur « Montage Assemblage » à la phase de vie 570. Elle est réalisée à la date 14:50:37T12-18-2008, par l'acteur métier YIG, dans le module TDC Need.

Une action de type MAJ attribue la valeur « Résister aux environnements pendant le stockage des composants et S/E » à la fonction de service 571. Elle est réalisée à la date 15:56:28T14-01-2009, par l'acteur métier YIG, dans le module TDC Need.

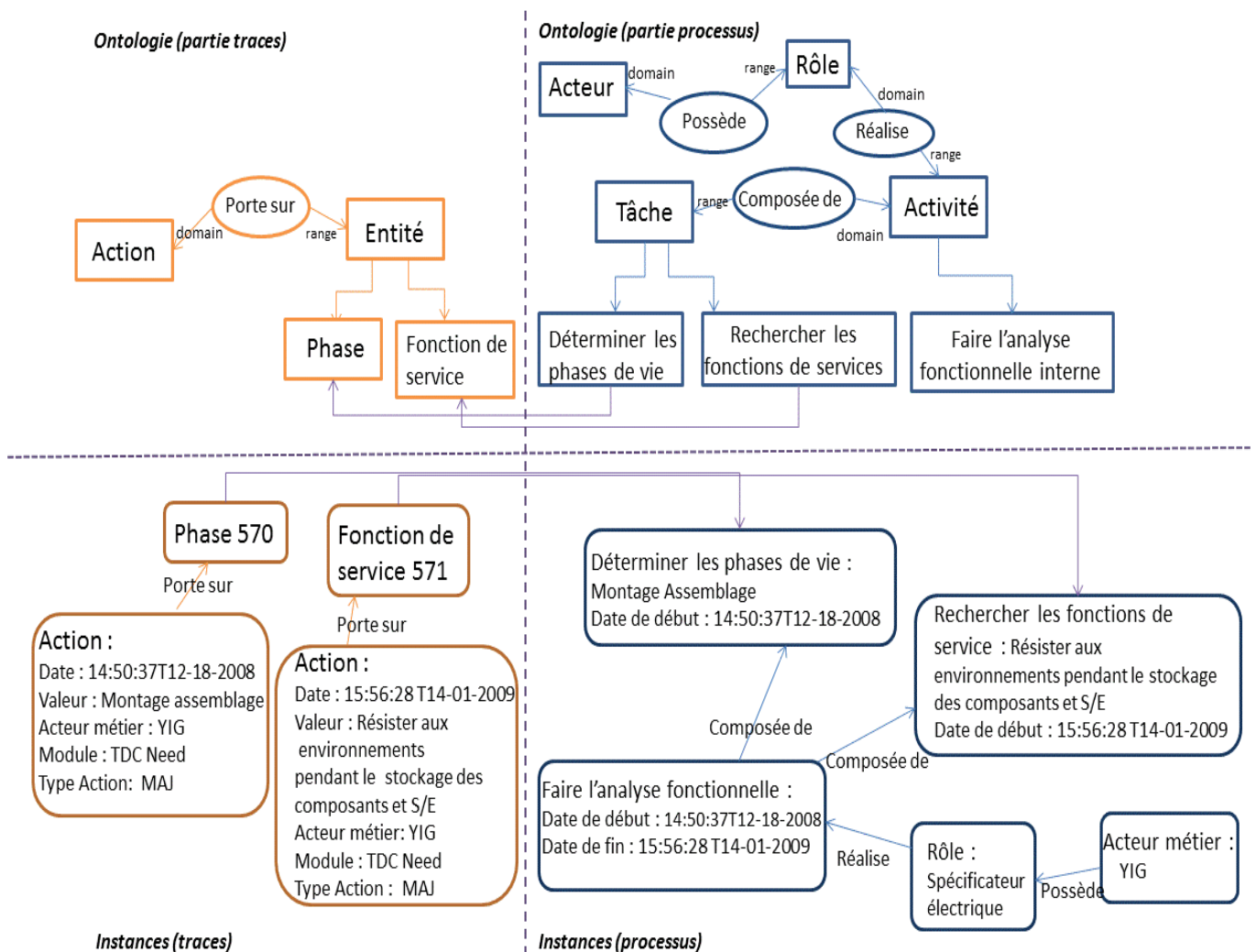


Figure 22. Exemple d'instances de traces et de processus à partir de l'ontologie OntoProcess

Ces traces sont pauvres en sémantique. C'est en utilisant l'ontologie des processus opérationnels que l'on enrichit les traces avec une nouvelle sémantique pour les rendre exploitable. La partie de l'ontologie des processus opérationnels que nous présentons dans la Figure 22 définit que l'acteur métier possède des rôles et qu'il réalise certaines activités selon ces rôles. Elle exige d'identifier les différents acteurs qui collaborent et de trouver pour chacun les rôles qu'il possède et les activités et les tâches qu'il réalise. Un acteur est identifié par son nom. Les activités peuvent être décomposées en des tâches atomiques (non décomposables).

L'extension d'OntoProcess présenté dans la Figure 24 concerne les activités d'analyse fonctionnelle. Chaque activité d'analyse fonctionnelle est décomposée en des activités en tâches, par exemple, déterminer les phases de vie et déterminer les fonctions de service.

L'ontologie permet donc d'identifier et de construire les instances d'activités à partir des actions réalisées sur les entités. Par exemple, une activité « Déterminer les fonctions de service » correspond à des actions sur les entités « fonction de service », et une activité « Déterminer les phases de vie » correspond à des actions sur les entités « phase de vie ».

Les activités sont regroupées pour chaque acteur pour leur permettre de retrouver la séquence de tâches qu'ils ont réalisées et leur permettre de prendre du recul sur ces activités (aspect réflexif du praticien). Le rôle de l'acteur est déduit à partir du module qu'il a utilisé et du métier auquel il appartient. Dans l'exemple, les deux actions sont réalisées par le même acteur YIG. YIG a utilisé le module TDC Need, donc YIG a un rôle générique de spécificateur. Par ailleurs le métier auquel appartient YIG a été renseigné : ingénierie électrique. Le rôle spécifique de YIG est donc « spécificateur électrique ». Si YIG utilisait le module TDC Structure, il aurait alors le rôle de « concepteur électrique ».

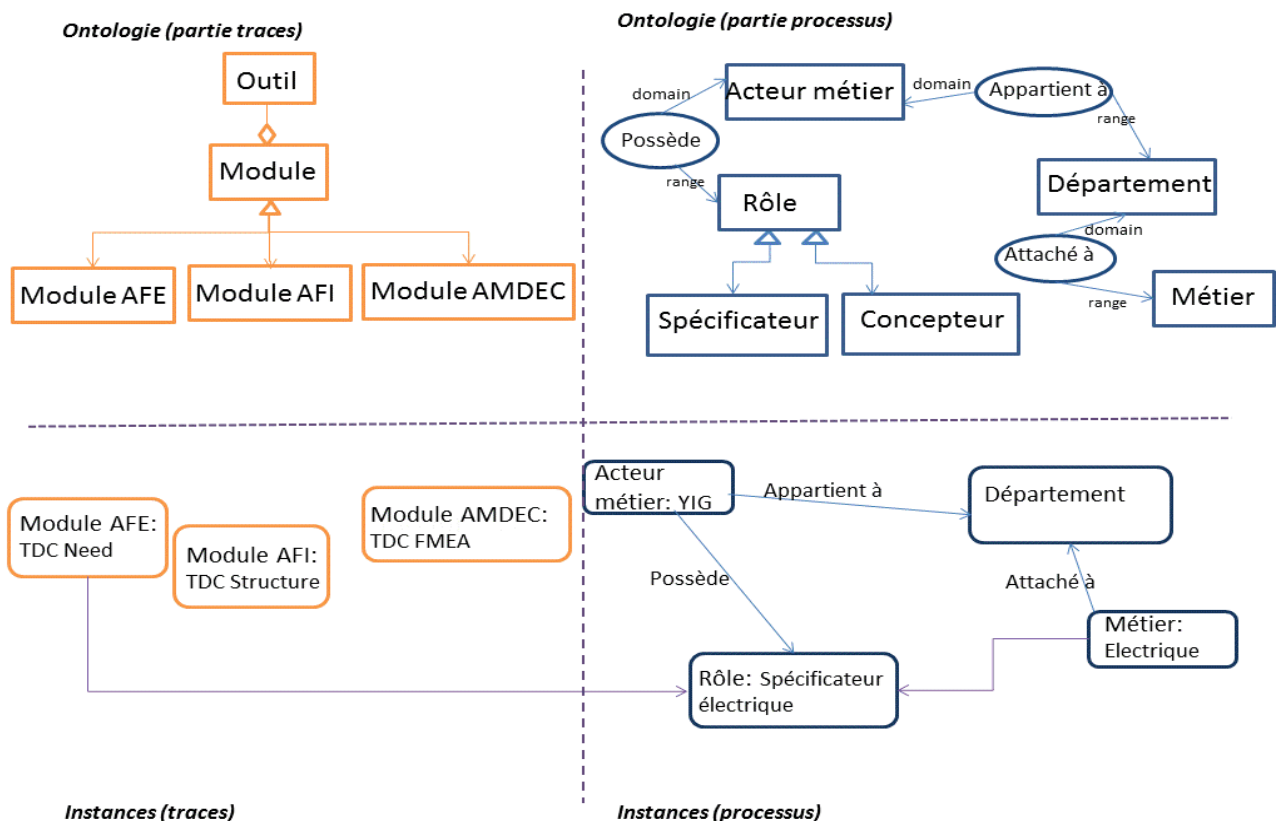


Figure 23. Exemple d'instances trace et rôle métier à partir de l'ontologie OntoProcess (suite de Figure 22)

Il est à noter que l'expert méthodologique doit pouvoir choisir le niveau de généralité dans la modélisation des démarches métier : soit générique en utilisant directement les activités représentées dans l'ontologie (les instances des activités de l'ontologie représentées dans la démarche métier porteront alors les mêmes intitulés que dans l'ontologie), soit spécifique à un type d'étude en identifiant les entités modifiées.

3.3.3.2 Extension de l'ontologie OntoProcess pour des modules spécifiques

Les figures ci-dessous montrent les différentes décompositions d'activités supportées par les modules de l'outil TDC :

- analyse fonctionnelle externe (Figure 24),
- analyse fonctionnelle interne ou structurelle (Figure 25),
- AMDEC produit (Figure 26),
- AMDEC Process (Figure 27).

Nous les présentons sous forme d'organisation hiérarchique dans laquelle tous les concepts héritent de la classe « activité ». Dans les démarches métiers de l'entreprise et les processus opérationnels construits, on retrouve des instances de ces concepts d'activités.

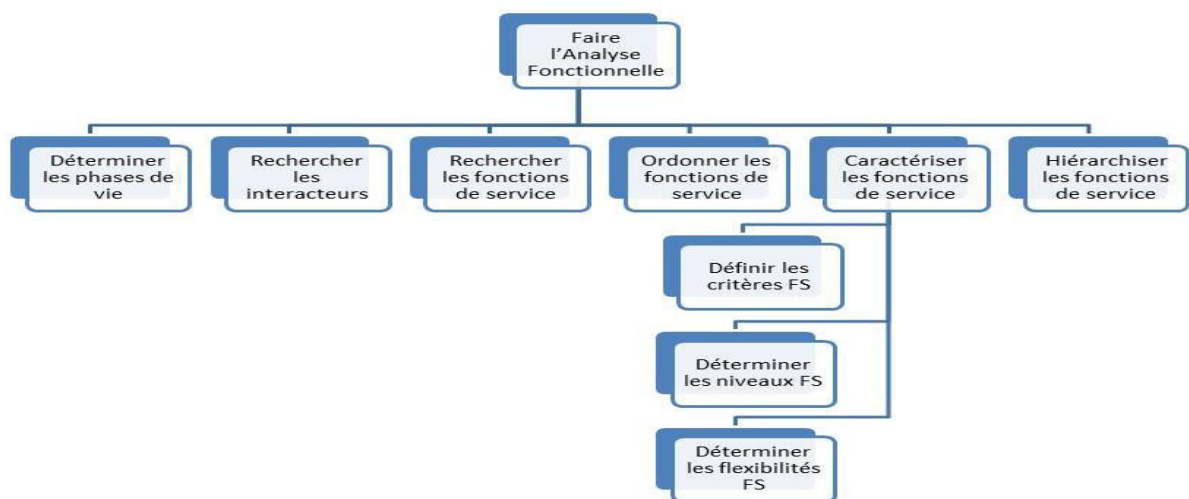


Figure 24. Extension de l'Ontologie OntoProcess pour les activités d'Analyse Fonctionnelle Externe

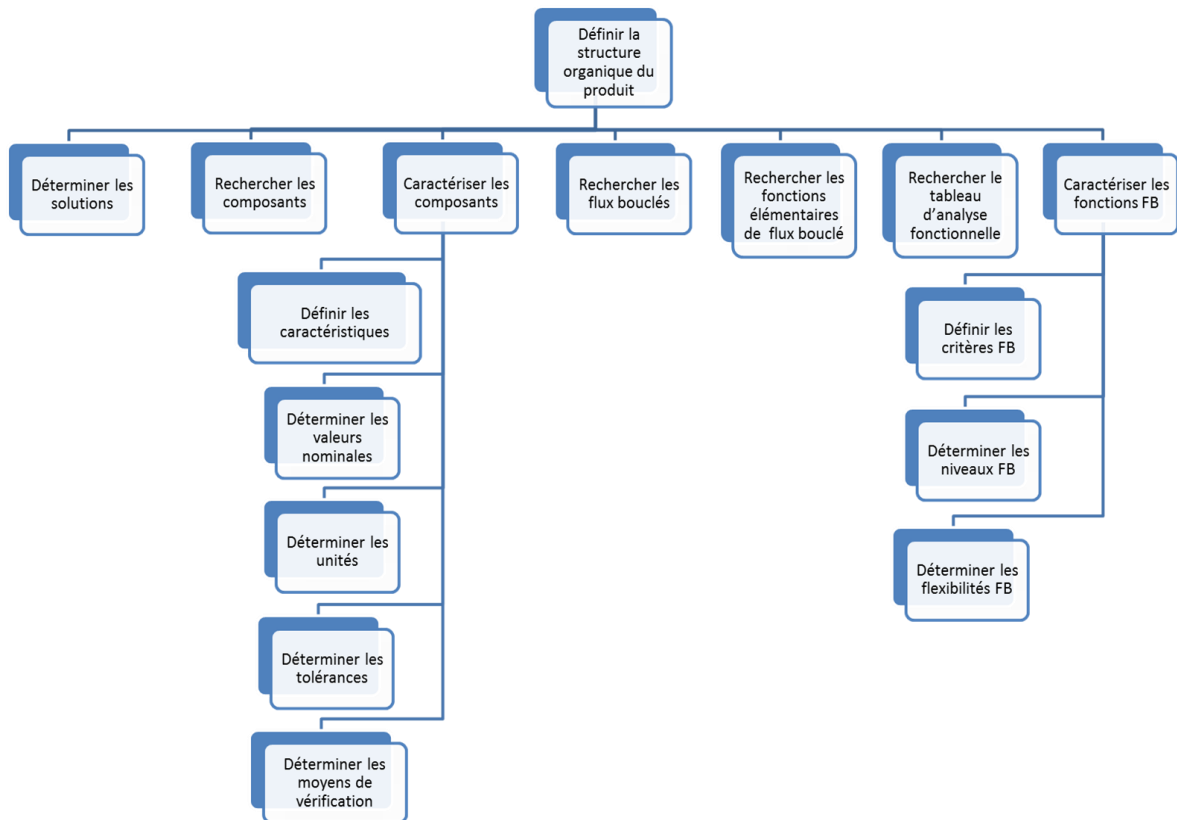


Figure 25. Extension de l'ontologie OntoProcess pour les activités d'Analyse Fonctionnelle Interne

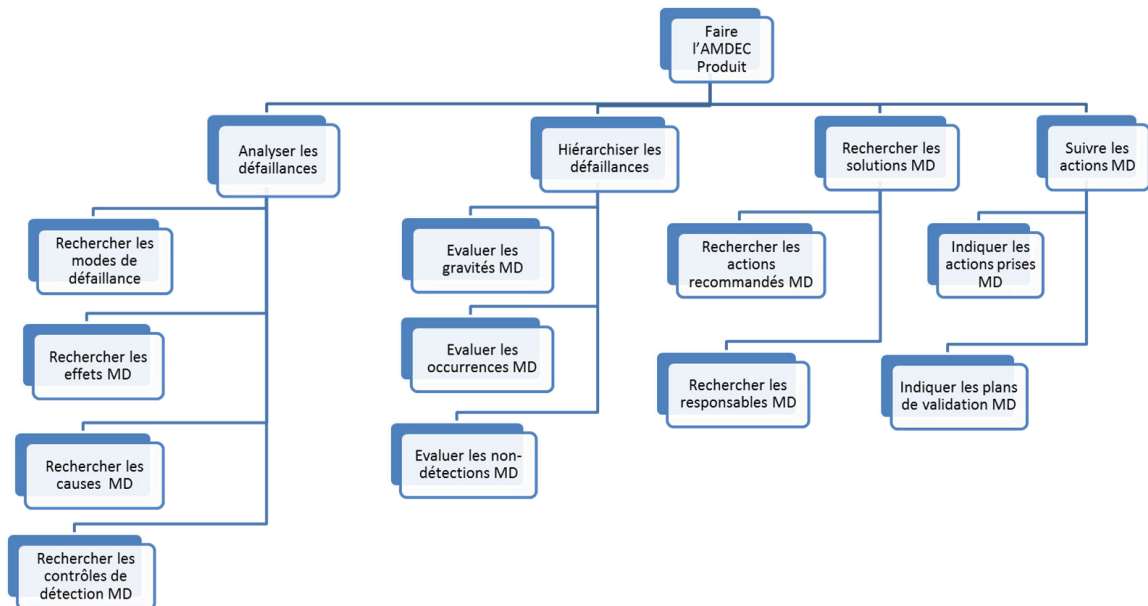


Figure 26. Extension de l'Ontologie OntoProcess pour les activités d'AMDEC Produit

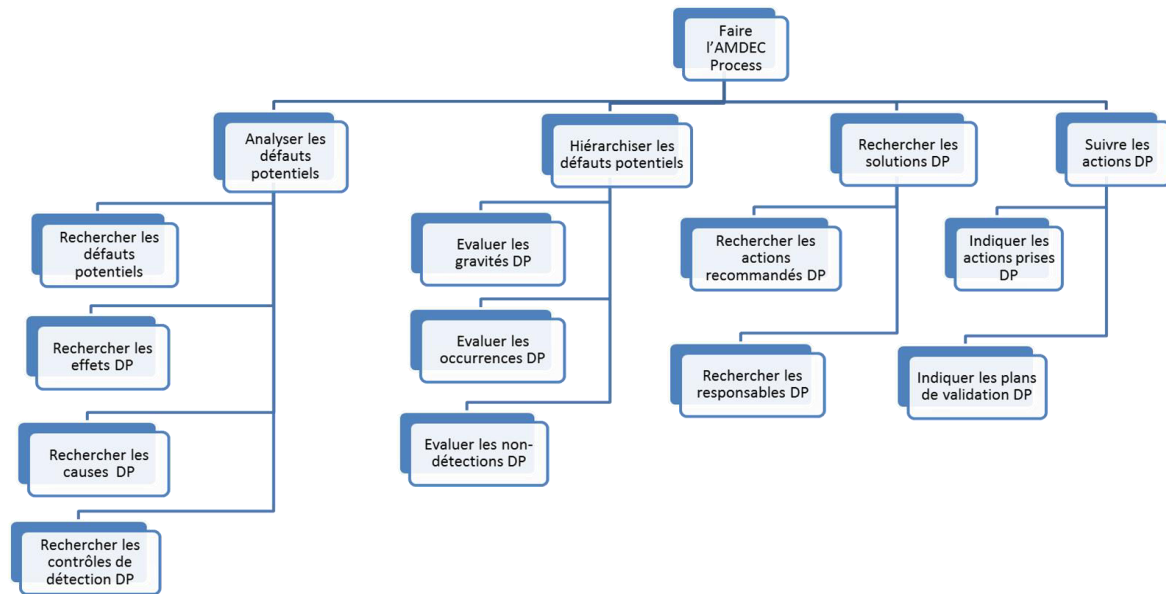


Figure 27. Extension de l'Ontologie OntoProcess pour les activités d'AMDEC Process

Nous avons représenté sous forme de tableau les correspondances qui permettent d'inférer les **activités** à partir des **entités** collectées dans les **traces**.

A titre d'exemple, le Tableau 3 concerne l'analyse fonctionnelle externe et le Tableau 4, l'AMDEC produit. Ces correspondances seront utilisées pour les transformations des traces en processus opérationnels.

Tâches dans OntoProcess	Entités dans les traces	Entité supérieure dans les traces
Déterminer les phases de vie	Phase de vie	
Rechercher les interacteurs	Interacteur	—
Rechercher les fonctions de service	Fonction de service	—
Définir les critères FS	Critère	Fonction de service
Déterminer les niveaux FS	Niveau	Fonction de service
Déterminer les flexibilités FS	Flexibilité	Fonction de service

Tableau 3. Table de tâches pour l'analyse fonctionnelle externe

Tâches dans OntoProcess	Entités dans les traces	Entité supérieure dans les traces
Rechercher les modes de défaillances	Mode de défaillance	Composant
Rechercher les effets MD	Effet	Mode de défaillance, Composant
Rechercher les causes MD	Cause	Mode de défaillance, composant
Rechercher les contrôles de détection MD	Contrôle de détection	Mode de défaillance, composant
Evaluer les gravités MD	Gravité	Mode de défaillance, composant
Evaluer les occurrences MD	Occurrence	Mode de défaillance, composant
Evaluer les non-détections MD	Détection	Mode de défaillance, composant
Rechercher les actions recommandées MD	Action recommandée	Mode de défaillance, composant
Rechercher les responsables MD	Responsable	Mode de défaillance, composant
Indiquer les actions prises MD	Action prise	Mode de défaillance, composant
Indiquer les plans de validation MD	Plan de validation	Mode de défaillance, composant

Tableau 4. Table des tâches de l'AMDEC Produit

La Figure 28 et Figure 29 montrent une partie de l'ontologie OntoProcess représentée sous Protégé, en décrivant les classes, les propriétés et les attributs.

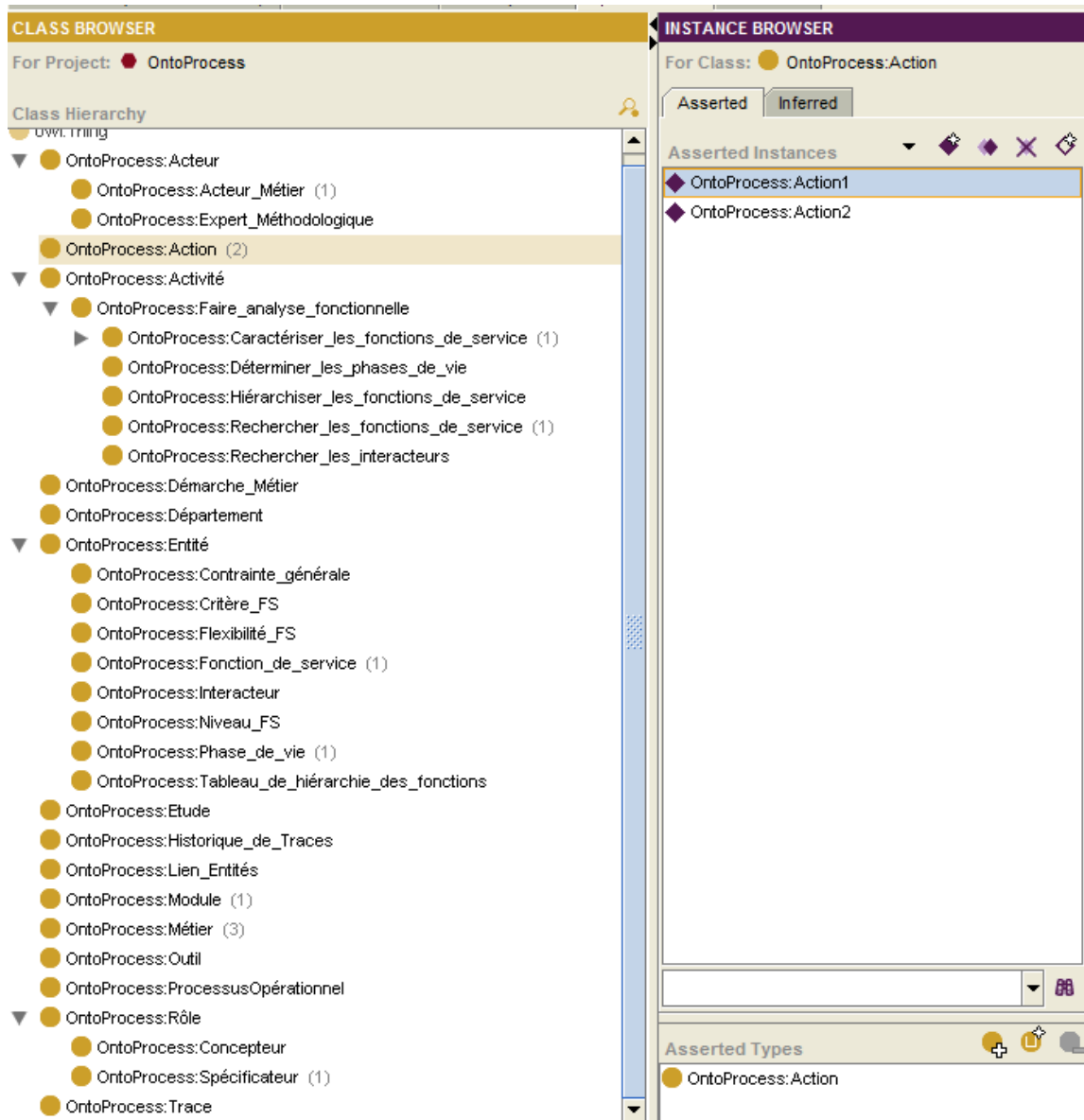


Figure 28. Classes de l'Ontologie OntoProcess représentées sous Protégé

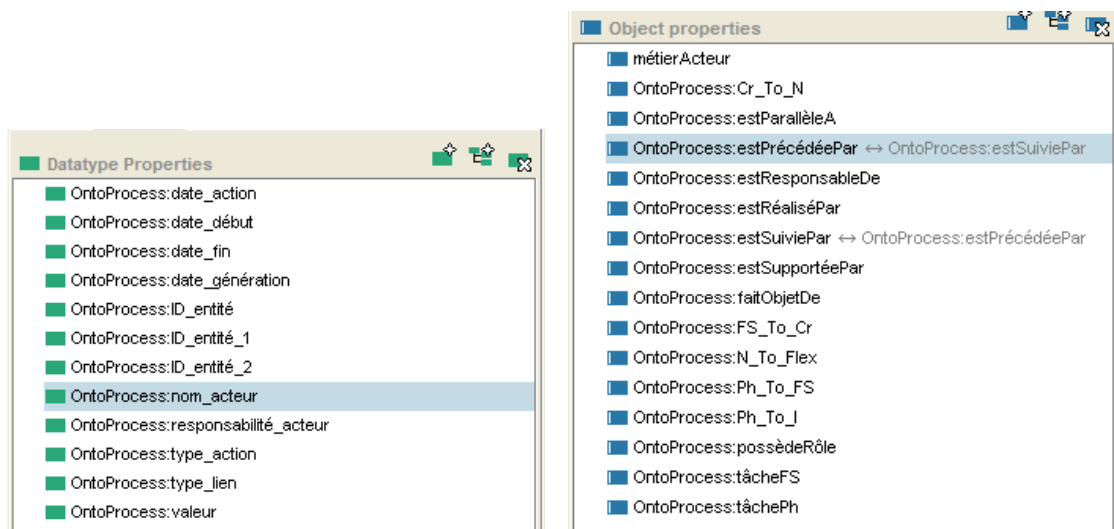


Figure 29. Propriétés et attributs utilisés dans OntoProcess, représentée sous Protégé

3.4 Conclusion

Dans ce chapitre, nous avons proposé un double cycle de capitalisation des connaissances pour des activités instrumentées. Il met en relation d'un côté, des processus opérationnels construits à partir de traces brutes enregistrées de l'activité et d'un autre côté, des démarches métiers qui sont de bonnes pratiques de conception de l'entreprise. L'enrichissement d'une démarche métier peut être réalisé, soit manuellement par visualisation et prise de recul sur le processus opérationnel pour en dégager de nouvelles activités à réaliser, soit assister par un système intelligent qui peut comparer ce processus et la démarche métier afin de proposer des améliorations. Nous proposons alors une approche globale de gestion des démarches métier, s'appuyant sur un système à base traces et proposant une assistance proactive à l'utilisateur.

La construction d'un processus à partir de traces nécessite le développement d'une ontologie du domaine. Dans notre recherche, nous nous sommes intéressés à des méthodes de conception supportées par la suite TDC, à savoir l'analyse fonctionnelle externe, l'analyse structurelle, l'AMDEC produit et l'AMDEC Process. L'implémentation de ces méthodes dans la suite TDC respectant les normes AFNOR relatives à ces méthodes, notre travail de recherche ne perd pas en généralité. Nous avons proposé une ontologie OntoProcess recouvrant le domaine couvert par ces quatre méthodes, proposant un modèle de traces lié à la réalisation d'une étude, et une modélisation organisationnelle s'appuyant sur une vue métier et une vue étude. Cette ontologie permettra d'une part, de retrouver les correspondances entre des instances de traces et des instances de processus, et d'autre part, d'assister l'expert méthodologique dans l'élaboration des démarches métier.

Actuellement, nous avons choisi de ne pas modéliser les relations logico-temporelles entre les activités dans OntoProcess : les enchaînements entre activités sont définis dans les démarches métiers par les experts métiers et ce, pour chaque type d'étude. Ce choix est lié au fait que l'on ne souhaite pas restreindre, ou trop contraindre, les possibilités d'enchaînement dans une démarche métier et laisser ainsi chaque entreprise libre de faire ses choix.

Ensuite, se pose la question de donner la possibilité à l'expert de créer ses propres activités, comme par exemple des activités de test, non prévues par les normes et qui permettraient de modifier le statut d'une information (comme un critère par exemple) de « non validée » à « validée ».

Deux approches peuvent être étudiées :

- permettre à l'expert méthodologique de créer une nouvelle activité dans une démarche métier et assurer la mise à jour automatique de l'ontologie. Cette option ne nécessite pas d'expertise sur les ontologies pour l'expert.
- modifier l'ontologie avec l'ajout de cette activité, pour permettre ensuite à l'expert de l'utiliser. Dans ce cas, il y aura recours à un expert sur les ontologies. L'expert est ainsi contraint d'utiliser les activités définies dans l'ontologie (respectant les normes). L'outil peut lui proposer les activités possibles, par un menu déroulant par exemple.

Quelle que soit l'approche choisie, un problème délicat de cohérence de cette nouvelle ontologie va se poser ainsi que des difficultés de mise à jour du logiciel pour l'éditeur, chaque client ayant alors des versions différentes d'ontologie. Des approches de type fusion d'ontologie sont à étudier pour tenter de répondre à ce challenge.

Chapitre 4. Architecture du système multi-agents MPPAS

L'adéquation des systèmes intelligents et particulièrement des systèmes multi-agents (SMA) pour traiter des problèmes d'ingénierie des connaissances nous a conduit à concevoir un SMA pour répondre à la problématique de la gestion des démarches métiers en entreprise. Nous présentons dans cette section une architecture de SMA pour faciliter la gestion des traces d'activité, la construction de processus opérationnels et la réutilisation de démarches métier. Le SMA a pour objectif d'analyser des traces des activités en fonction des rôles des acteurs métier. Il construit ensuite des processus opérationnels liés aux différents rôles. Il analyse et compare les processus opérationnels afin de définir une démarche métier, c'est-à-dire un processus de référence (reconnu comme une bonne pratique) pour un type d'étude.

Dans un premier temps, nous décrivons l'approche organisationnelle que nous avons choisie pour concevoir le SMA. Cette approche permet de détailler les rôles des agents, leurs objectifs et actions ainsi que le système de coordination entre les agents. Notre système multi agents est un MAIS (Multi Agent Information System) (Gandon et al., 2002) : il est conçu pour analyser et exploiter des informations et des connaissances. Les informations sont des traces d'activités situées dans un contexte organisationnel (rôle des acteurs métiers, actions réalisées, projet en cours, etc.). Les connaissances exploitées sont les démarches métier et les processus opérationnels validés.

4.1 Introduction

Les démarches métier définissent des guides opérationnels pour la réalisation des projets dans l'entreprise. Nos travaux de recherche s'intéressent aux démarches métier supportant les méthodologies de conception de produit. Les acteurs métier réalisent les activités définies dans les processus en utilisant, créant et partageant leurs savoirs et savoir-faire. Le projet d'ingénierie représente un environnement hétérogène (diversité professionnelle des acteurs métier), distribué (différentes entités informationnelles) et dynamique. En effet, les informations, les activités et les rôles des acteurs évoluent au fur et à mesure des projets.

Dans cet environnement complexe, il est nécessaire d'utiliser une approche où le système de gestion des connaissances liées aux démarches métier puisse avoir la capacité d'être :

- *Réactif*, c'est-à-dire capable de réagir lorsqu'une nouvelle activité est réalisée afin de la tracer et de construire un processus opérationnel.
- *Pro actif*, c'est-à-dire pouvoir assister les acteurs métier à choisir l'activité métier pertinente par rapport à son contexte de travail.

- *Flexible*, c'est-à-dire être capable de s'adapter aux changements de l'environnement tels que l'apparition de nouvelles activités dans le processus, de nouveaux rôles, etc.

Les trois caractéristiques présentées précédemment correspondent aux propriétés des systèmes multi-agents (SMA). Nous proposons ainsi un système intelligent, basé sur l'utilisation d'agents logiciels, qui à partir de la traçabilité des activités métier, identifie les rôles des acteurs métier, construit le processus opérationnel réalisé par ces acteurs et les compare à la démarche métier correspondante. L'objectif du SMA est donc d'apporter une assistance à la formalisation des démarches métier et à leur réutilisation. En fonction de ces objectifs, nous présentons dans la section 4.3 de ce chapitre l'architecture du SMA.

La traçabilité des activités réalisées par les acteurs métier permet d'identifier les rôles des acteurs, les entités informationnelles qu'ils manipulent et bien entendu, de comprendre qu'elles sont les activités qu'ils ont accomplies dans un projet. Afin d'interpréter et de différencier les entités informationnelles capturées dans les traces d'activités, les agents ont besoin d'utiliser un vocabulaire et une sémantique précise permettant de définir les concepts utilisés et leurs relations. Ces derniers sont définis dans l'ontologie présentée au chapitre 3 facilitant la compréhension de l'ensemble des entités informationnelles pour décrire les processus opérationnels utilisés lors des projets de conception de produit.

Par ailleurs, afin de construire un processus opérationnel ou de comparer deux processus, il est nécessaire d'utiliser des règles de construction et de comparaison. Ces règles seront décrites en détail dans le chapitre suivant mais nous expliquons dans la section 4.3.4 de ce chapitre comment s'organisent (coopèrent, négocient, collaborent) les agents pour se répartir les tâches de construction et de comparaison.

4.2 Méthodologie de conception du système multi-agents

Pour concevoir notre système multi-agents appelé MPPAS (Management of Professional Processes by Agents System) nous proposons une méthodologie inspirée des méthodes de conception de SMA existantes. Dans cette section, nous présentons les différentes méthodes de conception existantes (section 4.2.1) ainsi que la méthode que nous avons mise en œuvre (section 4.2.2).

4.2.1 Méthodologies pour la conception de SMA existantes

Les méthodologies pour la conception de SMA prennent en compte toutes les phases du cycle de vie d'une application orientée agents (Iglesias, Garijo, & Gonzalez, 1999). Nous présentons dans cette section quatre d'entre elles qui proposent des processus de développement, de l'analyse des besoins à l'implémentation. D'autres méthodologies existent, certaines sont basées sur une approche différente où le SMA est vu comme un ensemble de groupes d'agents jouant des rôles précis et ayant la faculté d'interagir entre eux pour résoudre leurs objectifs. Ces approches ont donné naissance à des modèles organisationnels qui nous ont servi pour la présentation de notre approche de modélisation organisationnelle (section 4.2.2).

4.2.1.1 La méthodologie Gaia

En 2000, (Wooldridge, 2000) propose la méthodologie Gaia dédiée à la spécification et au développement de SMA (Zambonelli, Jennings, & Wooldridge, 2003). La spécification de SMA à l'aide de Gaia s'effectue par le suivi d'un processus permettant de détailler le modèle du système à concevoir. La Figure 30 présente le processus composé de 'modèles' à appliquer pour chaque activité de développement du SMA.

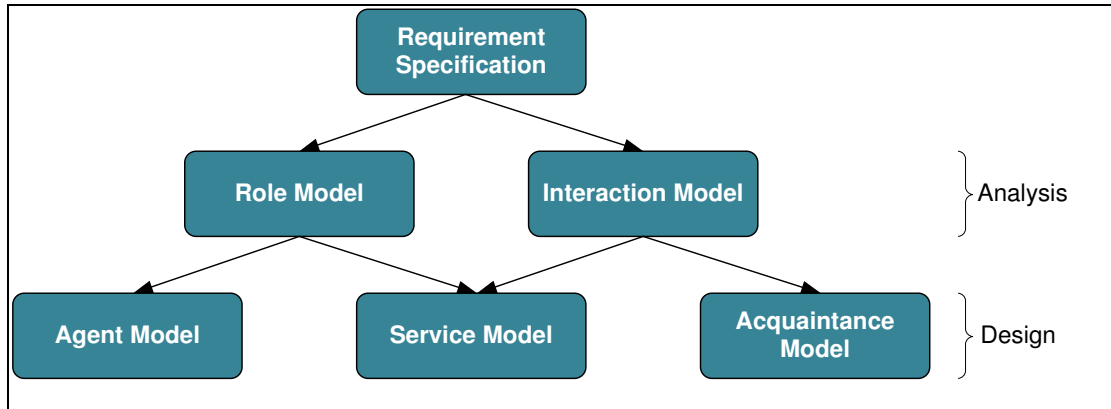


Figure 30. Le processus utilisé dans Gaia (Wooldridge, 2000)

Avec la méthodologie Gaia, le SMA est vu comme une organisation dans laquelle les agents assurent différents rôles. Ces rôles ont systématiquement des interactions entre eux. L'étape d'analyse est dédiée à la compréhension des organisations à travers deux modèles : le modèle des rôles et le modèle des interactions.

Un rôle est défini par quatre attributs : ses responsabilités, ses permissions, ses activités et ses protocoles de communications avec les autres rôles. Les responsabilités décrivent les fonctionnalités du rôle. Les permissions sont les droits que possède le rôle qui lui permettent d'accomplir ses responsabilités. Les permissions déterminent les ressources accessibles par le rôle et les droits qu'il détient pour les manipuler (lecture, modification...). Les activités sont les tâches que peut réaliser le rôle sans interagir avec les autres rôles. Les protocoles définissent les interactions entre les rôles.

Le modèle d'interaction est composé d'une série de définitions de protocole de communication. Il existe autant de protocoles que d'interactions entre rôles. Le modèle est centré sur la nature et l'objectif des interactions sans déterminer l'ordre des échanges de messages. Le protocole est une description de l'interaction avec son initiateur, le destinataire répondant à l'interaction, les informations en entrées ainsi que les informations en sorties.

L'objectif de la méthode Gaia est de transformer les modèles « rôles » et « interaction » décrits dans la phase d'analyse en un niveau suffisamment bas d'abstraction pour pouvoir appliquer des méthodes d'implémentation de système classiques. La phase d'analyse de Gaia détermine comment dans une société, les agents coopèrent pour atteindre leurs objectifs et qu'elles informations ils ont besoin pour les réaliser.

La phase de conception de Gaia est centrée sur la description de trois modèles : le modèle agent, le modèle des services et le modèle relationnel. Le premier modèle décrit les types d'agents que le système utilise en leur associant un ou plusieurs rôles. Le modèle de service spécifie les services associés à chaque rôle. Le modèle relationnel contient les liens de communication entre les différents types d'agents.

4.2.1.2 La méthodologie OperA

La méthodologie OperA (Dignum et al., 2004) est centrée sur la spécification formelle des groupes d'agents ; système est vu comme une organisation ou une société d'agent. L'approche de modélisation d'OperA consiste en deux phases principales : la première phase est dédiée à la spécification du modèle *organisationnel* qui comprend la définition de la structure organisationnelle et du comportement global du système. Dans la seconde phase, on spécifie la population d'agents peuplant la structure organisationnelle ainsi que les conditions spécifiques nécessaires à la mise en œuvre des rôles dans les organisations. Cette seconde étape est réalisée à travers la construction de deux différents modèles appelés modèle 'Social' et modèle d'Interaction'.

Le modèle *organisationnel* spécifie les caractéristiques d'une société d'agent à travers quatre structures :

- *La structure sociale.* Elle décrit les objectifs de la société, ses rôles et les modèles de coordination;
- *La structure dédiée aux interactions.* Elle détaille les interactions par des scripts qui représentent les tâches d'une société avec les actions coordonnées de plusieurs rôles. Elle donne un ordonnancement des exécutions des scripts lequel spécifie le déroulement des interactions entre les rôles;
- *La structure normative.* Elle décrit les normes de la société d'agent à partir de normes sur les rôles et les interactions. Les normes sont définies comme des séries de conditions à remplir pour exécuter un objectif du rôle ou une interaction ;
- *La structure communicative.* Elle spécifie les actes de communication et les concepts du domaine à travers des ontologies.

Dans le modèle *social* de la méthodologie OperA l'exécution des rôles des agents est fixée par des contrats 'sociaux' qui décrivent les capacités et les responsabilités des agents dans la société. Les contrats sont des règles qui définissent les conditions d'exécutions des rôles. L'utilisation de contrats pour décrire l'activité du système permet d'associer les objectifs des organisations avec les désirs des agents.

Dans le modèle *interactionnel* les interactions sont créées dynamiquement par les rôles des agents à partir de scripts d'interaction spécifiés dans le modèle organisationnel. Un rôle négocie les conditions d'interactions avec les autres rôles. Ces scripts d'interaction sont également spécifiés à l'aide de contrats.

La modélisation d'un système à l'aide d'OperA est donc réalisée en deux phases distinctes ; une première pour modéliser une société d'agents et l'autre pour rendre les agents de cette population (organisation) prêts à interagir entre eux en exhibant des comportements cohérents régis par un certain nombre de règles (contrats).

4.2.1.3 Conception d'un SMA avec PASSI

La méthodologie PASSI (Process for Agent Societies Specification and Implementation), proposée par Cossentino (Cossentino, 2005) décrit un processus de la prise en compte les besoins, de la conception jusqu'à l'implémentation d'un SMA orienté sociétés d'agents. La méthodologie intègre des modèles et concepts pour la conception à partir de la programmation orientée objets et des approches de l'intelligence artificielle.

Le processus permet de construire cinq modèles (Figure 31) :

- *System Requirement Model* présente les besoins du système. Il fournit une description des fonctionnalités du système, à partir d'une approche de décomposition des problèmes selon le paradigme agent. Quatre activités contribuent à définir le modèle ; (i) la description des besoins où le système est présenté en terme de fonctionnalités, (ii) l'identification des agents où des agents sont introduits pour répondre aux besoins identifiés, (iii) l'identification des rôles où interactions entre les agents sont décrits en spécifiant leurs rôles, (iv) la spécification des tâches où la planification de chaque agent est décrite ;
- *Agent Society Model* contient la spécification des rôles, des interactions, des dépendances et une ontologie. C'est la phase où le paradigme agent est pleinement exploité. Elle est construite à l'aide de trois composants : (i) une description de l'ontologie de domaine en terme de concepts, prédicats et actions, (ii) une ontologie dédiée aux communications des agents, décrivant en terme de référence aux éléments de l'ontologie de domaine, de protocole et de langage de communication, (iii) une description des rôles joués par les agents avec la spécification de leurs dépendances ;
- *Agent Implementation Model* décrit l'architecture en terme de classes et méthodes à deux différents niveaux d'abstraction : multi et mono agent. Ce modèle se construit selon deux activités successives ; la définition de la structure du SMA et la définition des comportements des agents ;
- *Code Model* contient le code source. Cette section permet la réutilisation des programmes déjà implémentés;
- *Deployment Level* spécifie la répartition des parties du système (agents) en fonction des machines ou en fonction des plateformes logicielles utilisées. Ce niveau possède également une composante dédiée aux tests. Ces derniers sont partagés en deux activités : les tests dédiés aux agents et les tests dédiés aux sociétés d'agents. Les premiers vérifient chaque comportement des agents par

comparaison avec les besoins identifiés en début de méthode. Les seconds tests sur les sociétés sont réalisés par vérification des résultats globaux du SMA.

La méthodologie PASSI permet de concevoir des SMA complexes. Afin de pouvoir traiter les petits projets, les auteurs ont conçu la méthode « Agile PASSI » (Chella, Cossentino, Sabatucci, & Seidita, 2004). Cette méthode conserve la structure itérative de PASSI mais permet d'utiliser un processus plus rapide centré sur le code plutôt que sur la documentation des modèles.

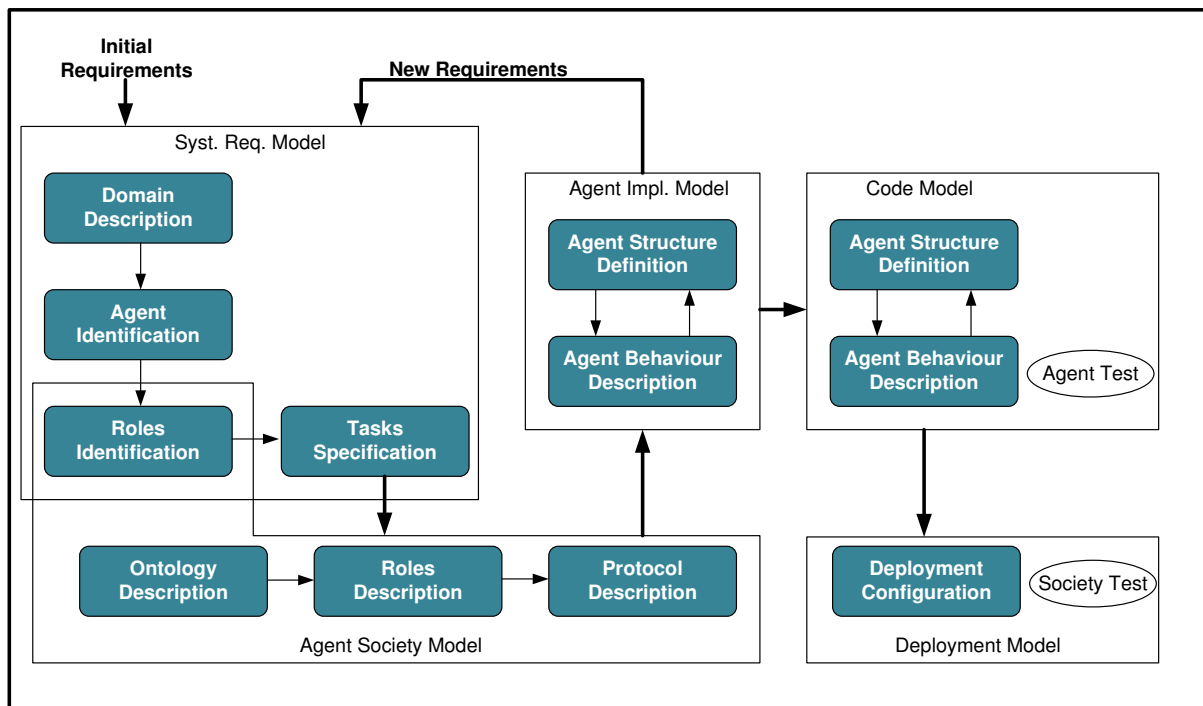


Figure 31. La méthodologie PASSI (Cossentino, 2005)

4.2.1.4 La méthodologie TROPOS

TROPOS (Bresciani, Perini, Giorgini, Giunchiglia, & Mylopoulos, 2001), (Giunchiglia, Mylopoulos, & Perini, 2002) est une méthodologie de développement de système informatique orienté agents. La spécificité de la méthode repose sur la phase de définition des besoins où les intentions des utilisateurs sont identifiées et analysées. Cette phase est fortement inspirée par le cadre du travail de Yu (E. Yu, 1995).

Tout le système de spécification des besoins est basé sur les objectifs que le système doit atteindre ainsi que sur l'identification des acteurs (humains et agents) qui permettent d'atteindre les objectifs du système. Le processus proposé par la méthode est simplement l'analyse des buts pour chaque acteur et leurs spécifications en termes d'algorithmes.

Le langage conceptuel de modélisation utilisé en phase d'analyse est dérivé de i^* et fournit une notation graphique et une série de techniques pour analyser les objectifs. La

notation permet également d'élaborer des spécifications formelles et informelles. Les concepts principaux du langage sont l'acteur, l'objectif, la tâche et les ressources :

- L'*acteur* peut représenter un rôle ou une série de rôles joués par un agent dans une structure organisationnelle donnée ;
- L'*objectif* décrit un intérêt stratégique des acteurs. Deux principaux types d'objectifs sont proposés, les 'hardgoals' et les 'softgoals'. Les premiers peuvent être atteints alors que les seconds ne sont pas forcément satisfaits. Ces derniers sont utiles pour représenter comment un état peut être atteint ;
- La *tâche* spécifie la façon de réaliser un ensemble d'actions. Cela peut être un moyen de satisfaire un objectif ou un 'softgoal' ;
- Une *ressource* est une entité informationnelle utilisée pour une tâche donnée ou pour atteindre un objectif spécifique.

La phase de conception utilise la spécification des acteurs pour établir la structure organisationnelle du système, c'est-à-dire l'identification des relations entre les agents et leurs propriétés. La phase d'implémentation utilise un ensemble d'outils orientés UML.

TROPOS couvre toutes les phases d'un processus de développement logiciel de l'analyse des besoins à l'implémentation du code source. Les phases du processus sont itératives. Cette méthodologie est adaptée au développement de SMA dont les besoins sont difficilement identifiables par une conception à l'aide d'objectifs secondaires (softgoals).

4.2.1.5 Autres méthodologies organisationnelles

Comme nous l'avons présenté auparavant, les approches traditionnelles pour la spécification de système multi-agents sont centrées sur la définition individuelle de l'agent et des tâches qu'il peut réaliser. D'autres approches prennent en compte le système de manière globale en utilisant un point de vue organisationnel pour décrire l'aspect social, c'est-à-dire le rôle des agents ainsi que leurs relations à travers le type d'interactions qu'ils ont. Plusieurs travaux de recherche ont abordé des points de vue différents et complémentaires pour l'utilisation d'approche organisationnelle dans la conception du système intelligent. Zambonelli dans (Zambonelli, Jennings, & Wooldridge, 2001) présente les règles organisationnelles que doivent suivre les agents. Dignum dans (Dignum, 2004) ainsi que Esteva (Esteva, Rodriguez-Aguilar, Sierra, Garcia, & Arcos, 2001) décrivent les normes et contrats que doivent respecter les agents dans une organisation alors que Parunak (Parunak & Odell, 2002) propose de centrer la conception du système sur les structures sociales des agents dans l'organisation. Un autre point de vue est celui du modèle AGR (Agent/Group/Role) qui met en évidence la notion de groupe (intra groupe et inter groupe) d'agents dans l'organisation.

Nous proposons dans la section suivante 4.2.2 une méthodologie de conception de système multi-agents basée, en partie, sur ces différentes approches organisationnelles.

4.2.2 Notre approche de modélisation organisationnelle et de construction des cartes d'identité des agents

Nous présentons une approche de modélisation organisationnelle pour concevoir le système multi-agents qui gèrera les démarches métier. Cette approche nous permet de décrire les aspects sociaux (rôles, interactions, objectifs communs, etc.) et coopératifs (interactions, coopérations) des agents. En effet, d'après la définition de Dignum (Dignum, 2004), les organisations d'agents peuvent être vues comme des ensembles d'entités et leurs interactions, régulées par des mécanismes d'ordre sociaux et créées par des acteurs autonomes pour accomplir un but commun.

Le point central de la conception d'un SMA à partir d'une approche organisationnelle est le rôle. Le rôle de l'agent représente sa position dans les organisations auquel il appartient. Nous pouvons décrire le rôle à l'aide de sa carte d'identité, c'est-à-dire en définissant l'ensemble des caractéristiques organisationnelles telles que :

- L'organisation auquel il appartient,
- Ses propres objectifs
- Les interactions qu'il peut avoir avec les autres agents,
- Les compétences qu'il possède, c'est-à-dire les tâches particulières qu'il accomplit
- Les connaissances dont il a besoin pour traiter l'information. Ces connaissances sont souvent décrites à l'aide d'une ontologie.

Nous proposons également de décrire notre système sous trois points de vue :

- *La structure organisationnelle* avec les missions des agents et leurs objectifs au sein de leurs organisations, leurs processus (liste et ordonnancement des tâches à accomplir), et la liste des résultats à fournir,
- *Le point de vue expertise de l'agent* avec la description de ses savoirs faire et responsabilités,
- *Le point de vue connaissances* où sont décrites les connaissances utilisées par le système et par chaque agent en fonction de ses activités.

Les différents points de vue permettent ainsi de mettre en corrélation la structure organisationnelle, la description des connaissances et la spécification des rôles des agents dont l'importance est précisée par Guizzardi (Guizzardi, 2006) : « Une organisation d'agents est une communauté de partage de connaissances dans laquelle les agents collaborent et échangent des connaissances pour réaliser leurs activités. »

Afin de représenter graphiquement les organisations, nous utilisons le formalisme RIOCK (Role Interaction Organisation, Competence & Knowledge) (D. Monticolo, 2008).

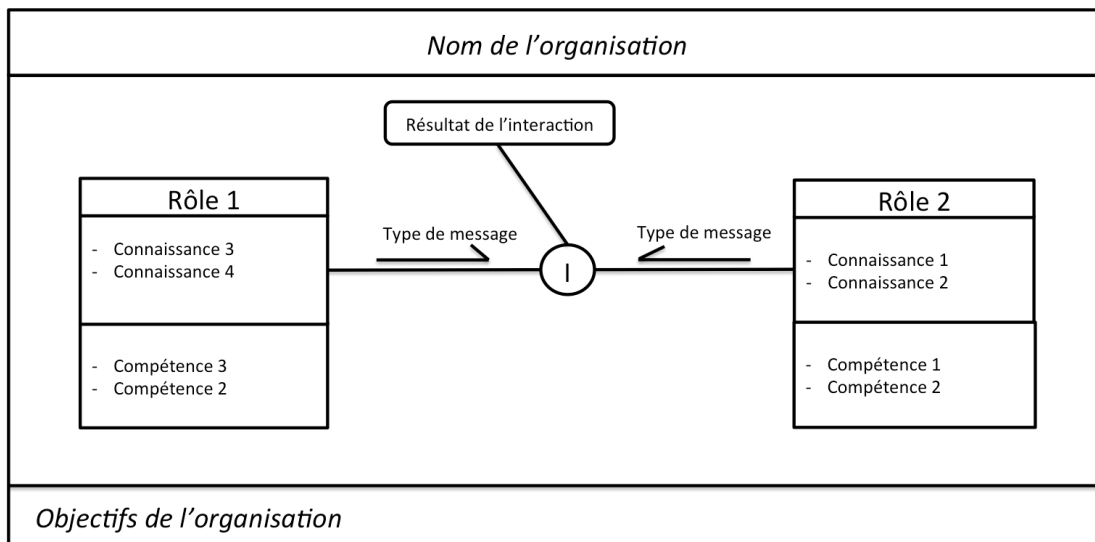


Figure 32. Modélisation d'une organisation à l'aide du formalisme RIOCK

En utilisant le formalisme RIOCK, l'organisation est représentée dans un cadre principal (Figure 32). Le nom de l'organisation est présenté dans la partie supérieure. Elle est composée de rôles représentés par des rectangles. Chaque rôle possède au moins une connaissance dont il a besoin pour réaliser les objectifs de l'organisation qui sont décrits dans la partie inférieure. De plus, pour chaque rôle, nous listons les compétences dont il est doté pour réaliser ces actions.

Dans une organisation, les rôles sont en interaction. L'interaction est illustrée par un cercle relié par des segments aux rôles. Sur chaque segment, nous décrivons le type de messages envoyés par chaque agent.

L'interaction entre les agents est de plusieurs types :

- *L'alliance* : Cette interaction représente un ensemble d'agents partageant les mêmes objectifs. Les agents vont coopérer pour accomplir les objectifs de l'organisation. Ils vont mettre de côté leurs propres objectifs. L'organisation perdurera tant que les agents n'ont pas satisfait ses objectifs. Avec le formalisme RIOCK, l'alliance est représentée par un A dans le cercle de l'interaction.
- *La coalition* : Les agents partagent des informations pour pouvoir accomplir leurs tâches. Ils gardent ainsi leurs propres objectifs et possèdent des objectifs différents. Cependant ils ont au moins un objectif commun avec l'organisation. Un agent peut sortir de la coalition si ses propres objectifs sont atteints. L'organisation perdure tant qu'il y a plusieurs agents en interaction et que les objectifs de l'organisation ne sont pas atteints. Avec le formalisme RIOCK, la coalition est représentée par un C dans le cercle de l'interaction.
- *Le leadership* : Cette interaction exprime le fait qu'un agent pilote l'organisation et distribue les tâches aux autres agents pour accomplir les objectifs de l'organisation. Chaque agent de l'organisation partage au moins un objectif commun avec l'organisation. Le leader possède le plus d'objectifs dans l'organisation. Avec le formalisme RIOCK, l'alliance est représentée par un L dans le cercle de l'interaction.

4.3 Architecture globale du système

Dans cette section, nous présentons l'architecture globale du SMA.

4.3.1 Identification des organisations d'agents

Le système multi-agents a pour objectif de construire les processus utilisés dans l'entreprise et d'aider les acteurs métier à optimiser leurs processus. La Figure 34 présente l'ensemble des fonctionnalités que doit accomplir le système intelligent, et ce, en cohérence avec la section 3.2.

Le système doit :

- Assurer la traçabilité des actions des acteurs métier,
- Construire les processus opérationnels en fonction des rôles des acteurs métier,
- Créer et mettre à jour les démarches métier,
- Comparer les processus opérationnels avec les démarches métier,
- Aider les acteurs métier à exécuter leurs processus (assistance).

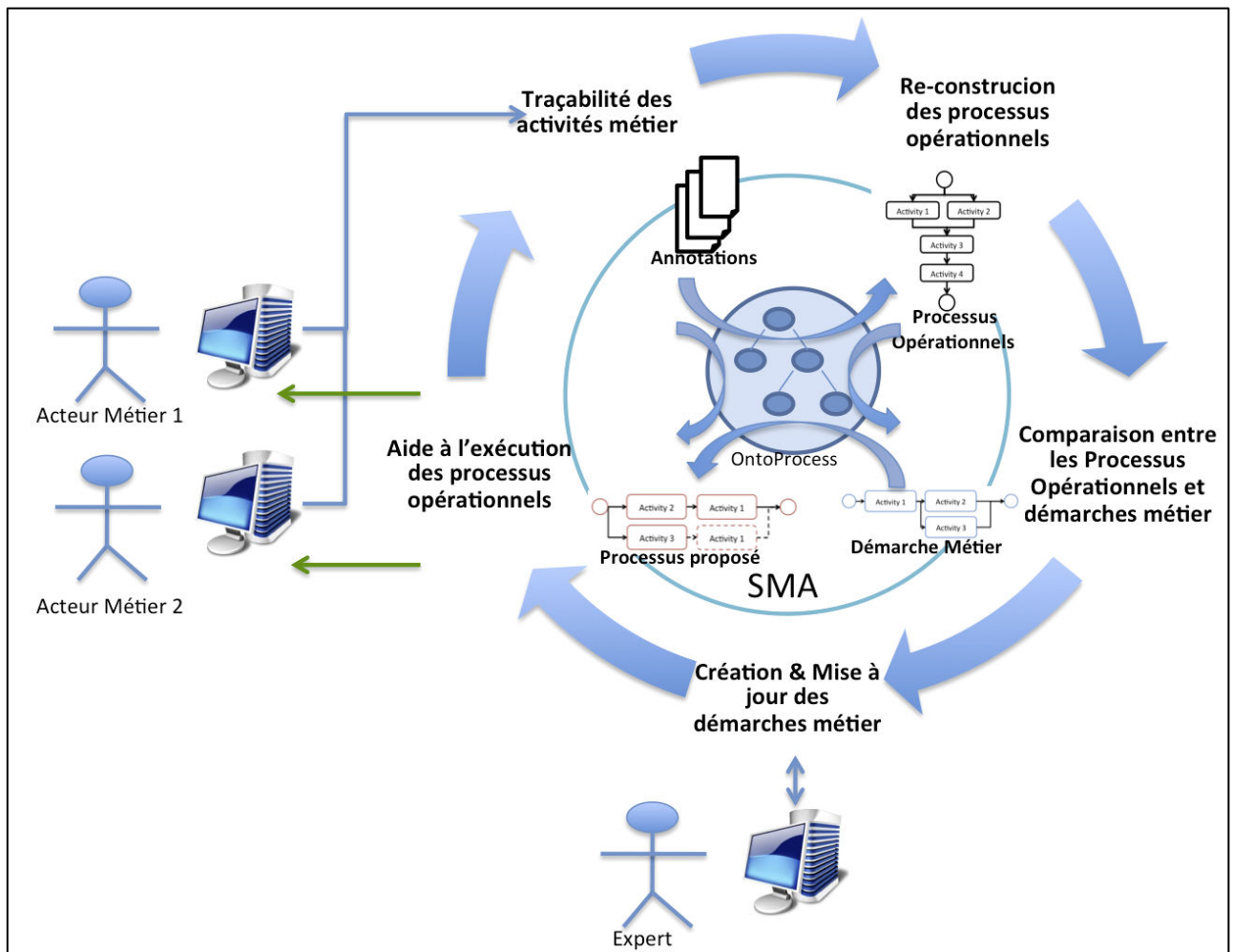


Figure 33. Liste des fonctionnalités que doit assurer le système multi-agent

Les entités créées et manipulées par le système multi-agents sont :

- les traces d'activités,
- les annotations permettant d'enrichir ces traces en les contextualisant,
- l'ontologie OntoProcess permettant au système de comprendre les concepts identifiés dans les traces,
- les processus opérationnels et les démarches métier formalisés avec le langage BPMN.
- les écarts entre les processus opérationnels et les démarches métier.

Dans les sections suivantes, nous détaillons l'architecture du système en présentant les différents types d'agents avec leurs rôles (section 4.3.2), leurs propriétés (section 4.3.3) et leur organisation (section 4.3.4).

4.3.2 Niveau individuel

A partir des fonctionnalités du système intelligent présenté à la Figure 33, nous pouvons déduire les caractéristiques individuelles des agents, c'est-à-dire leurs rôles, leurs expertises et interactions. Nous présentons donc la carte d'identité de chacun des rôles des agents du système intelligent.

4.3.2.1 Le rôle « Extracteur de traces »

L'agent ayant le rôle d'Extracteur de Traces (ET) a pour premier objectif de capturer les historiques des actions des acteurs métier dans la suite logicielle TDC. L'agent construit ses propres requêtes SQL permettant d'interroger la base de données de TDC. Il récupère les actions réalisées en fonction des acteurs métier, de leurs rôles, des dates de réalisation, des entités informationnelles manipulées ou créées, etc.

A partir du résultat des requêtes SQL, l'agent construit des traces des actions. Afin de connaître la structuration des traces, l'agent ET utilise le modèle de traces (Figure 53) de l'ontologie OntoProcess, qui sera détaillé dans la section 5.3.1.

Organisation	Gérer les traces d'activités métier
Rôle	Extracteur de Traces (ET)
Objectifs	Extraire les traces des activités métier Créer les fichiers XML où sont formatées les traces. (base de traces)
Interaction	Agent ARAO
Compétences	Création de requêtes SQL, Création de fichier XML
Connaissances	Accès à la base de données, Nom du projet, modèle de traces (Ontologie OntoProcess)
Interface	Sans interface
Connexions	Base de données, Ontologie OntoProcess
Nombre	Un agent ET par projet

Tableau 5. Carte d'identité du rôle « Extracteur de Traces »

L'agent ET n'a pas d'interface particulière puisqu'il exécute des tâches de traitement de l'information (Figure 34). Il y a un agent ET par projet afin de gérer l'ensemble des tâches du projet.

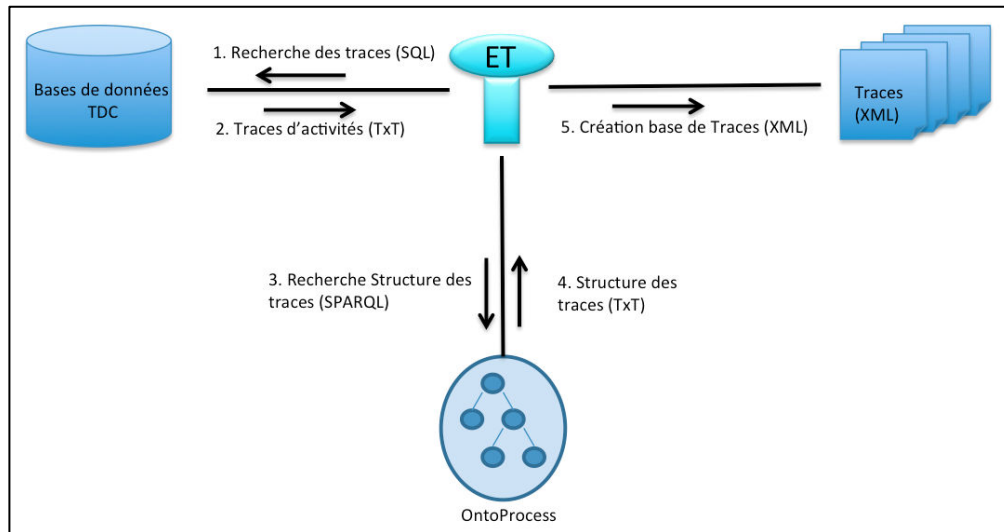


Figure 34. Opérations réalisées par l'agent ET

4.3.2.2 Le rôle « Analyseur et Rédacteur d'Annotations Organisationnelles »

Les principaux objectifs de l'agent « Analyseur et Rédacteur d'Annotations Organisationnelles » (ARAO) est d'analyser les traces des activités des acteurs métier et de les annoter.

Organisation	Gérer les traces d'activités métier
Rôle	Analyseur et Rédacteur d'Annotations Organisationnelles (ARAO)
Objectifs	Analyse les traces des activités métier Annoter les traces selon leurs contextes organisationnels
Interaction	Agent ET, Agent CPO
Compétences	Lecture des fichiers XML, Détection du contexte Organisationnel, création de fichier RDF
Connaissances	Modèle Conceptuel Organisationnel (Ontologie OntoProcess)
Interface	Sans interface
Connexions	Base de traces, Ontologie OntoProcess
Nombre	Un agent ARAO par projet

Tableau 6. Carte d'identité du rôle « Analyseur et Rédacteur d'Annotations Organisationnelles »

Afin d'analyser les traces des activités métier, l'agent ARAO a besoin d'utiliser l'ontologie OntoProcess afin d'identifier les activités réalisées par les acteurs, les relations entre elles et de déduire le rôle des acteurs métier.

Ces identifications vont permettre à l'agent ARAO d'annoter les traces avec un contexte organisationnel dans lequel sont décrits le rôle de l'acteur, le projet/étude auquel il participe ainsi que les tâches qu'il a réalisées.

L'agent ARAO a pour compétence la lecture des fichiers XML dans lesquels sont stockées les traces d'activité et la création de fichier RDF intégrant les annotations en fonction des contextes organisationnelles (Figure 35). L'agent n'a pas d'interface particulière puisqu'il assure du traitement de l'information. Il communique avec l'agent ET pour récupérer les traces d'activité. Il existe un agent ARAO par projet.

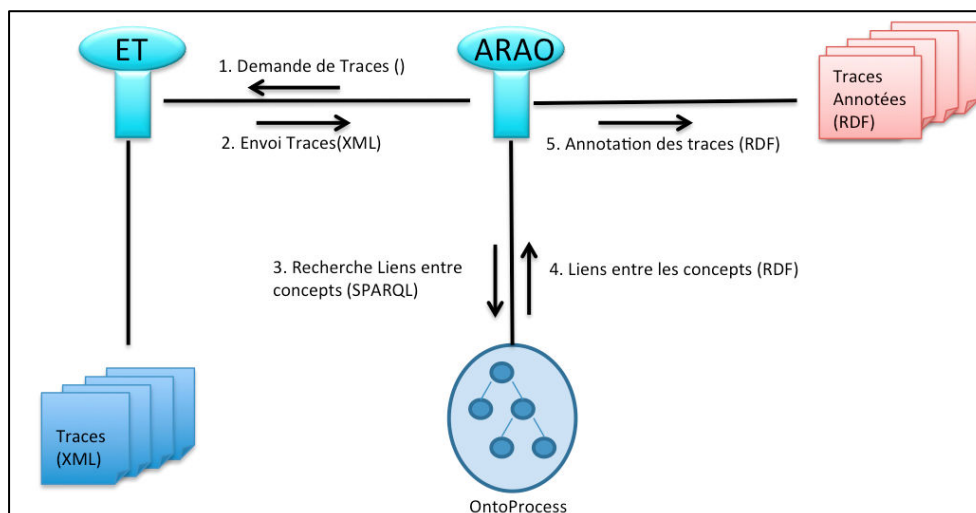


Figure 35. Opérations réalisées par l'agent ARAO

4.3.2.3 Le rôle « Constructeur de Processus Opérationnel »

Les objectifs principaux du rôle Constructeur de Processus Opérationnels (CPO), sont d'exploiter les traces des activités pour construire le processus opérationnel des acteurs métier.

Organisation	Gérer les Processus Opérationnels et Démarches Métier
Rôle	Constructeurs de Processus Opérationnels (CPO)
Objectifs	Interpréter les traces des activités métier Construire le processus opérationnel pour chaque rôle
Interaction	Agent ARAO, Agent CPODM
Compétences	Lecture des fichiers XML, Appliquer des règles de transformation, Construire des processus opérationnels, représenter graphiquement un processus opérationnel
Connaissances	Règles de transformation, Modèle conceptuel des connaissances (Ontologie OntoProcess), langage BPMN
Interface	Interface graphique
Connexions	Base de traces (RDF), Ontologie OntoProcess
Nombre	1 agent CPO par acteur métier

Tableau 7. Carte d'identité du rôle « Constructeur de Processus Opérationnel »

L'agent CPO communique avec l'agent ARAO pour obtenir les traces annotées. Il a besoin de l'ontologie OntoProcess pour comprendre ces traces, c'est-à-dire identifier les différentes entités d'information et leurs relations.

L'agent CPO utilise également des règles de transformation pour construire les processus opérationnels en fonction des traces annotées. Nous détaillons les règles de transformation dans le chapitre 5.

L'expertise de l'agent CPO porte sur l'application des règles de transformation et la construction de processus opérationnel au format BPMN (Figure 36). L'agent possède une interface graphique pour représenter les processus. Il existe un agent CPO par acteur métier.

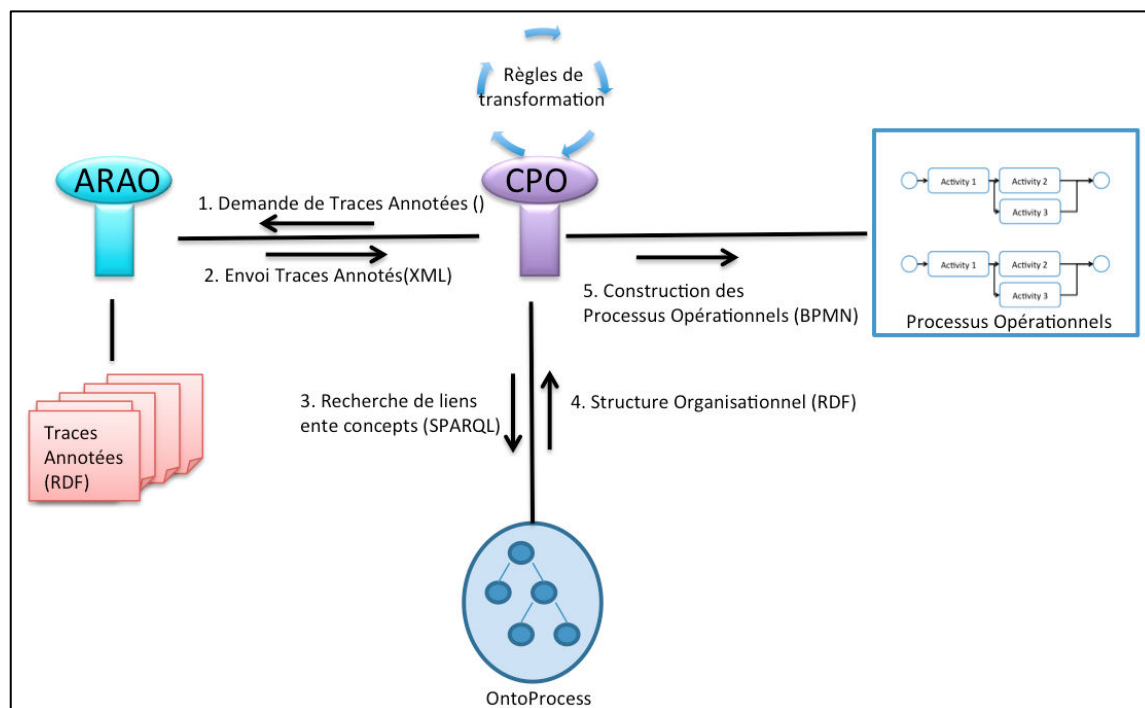


Figure 36. Opérations réalisées par l'agent CPO

4.3.2.4 Le rôle « Comparateur Processus Opérationnels et Démarche Métier »

L'agent ayant le rôle « Comparateur de Processus Opérationnels et Démarche Métier » (CPODM) a pour objectif de comparer les processus opérationnels générés par l'agent CPO avec la démarche métier.

Organisation	Gérer les Processus Opérationnels et Démarches Métier
Rôle	Comparateur PO/DM (CPODM)
Objectifs	Estimer les écarts entre PO/DM
Interaction	Agent CPO, Agent AEM
Compétences	Comparaison des processus Formaliser et stocker les écarts
Connaissances	Algorithmes de comparaison des processus
Interface	Sans interface
Connexions	...
Nombre	1 agent CPODM par rôle d'acteur métier

Tableau 8. Carte d'identité du rôle CPODM

L'agent CPODM communique avec l'agent CPO pour obtenir les processus opérationnels formalisés. Il utilise l'ontologie OntoProcess pour rechercher les différents liens entre les

concepts qui définissent les processus opérationnels. Il peut ensuite comparer les processus opérationnels avec la démarche métier formalisée par l'expert. Lorsqu'il a terminé la comparaison, l'agent stocke les écarts entre les processus opérationnels et la démarche métier.

Chaque processus opérationnel correspond à l'exécution d'un ensemble d'activités par les acteurs humains en fonction de leurs rôles (Figure 37). La démarche métier spécifie l'ensemble des processus pour un type d'étude en spécifiant les activités qui sont conduites par chacun des rôles de l'entreprise. L'agent peut donc comparer un processus opérationnel avec une démarche métier en utilisant les annotations organisationnelles qui spécifient le rôle de l'acteur.

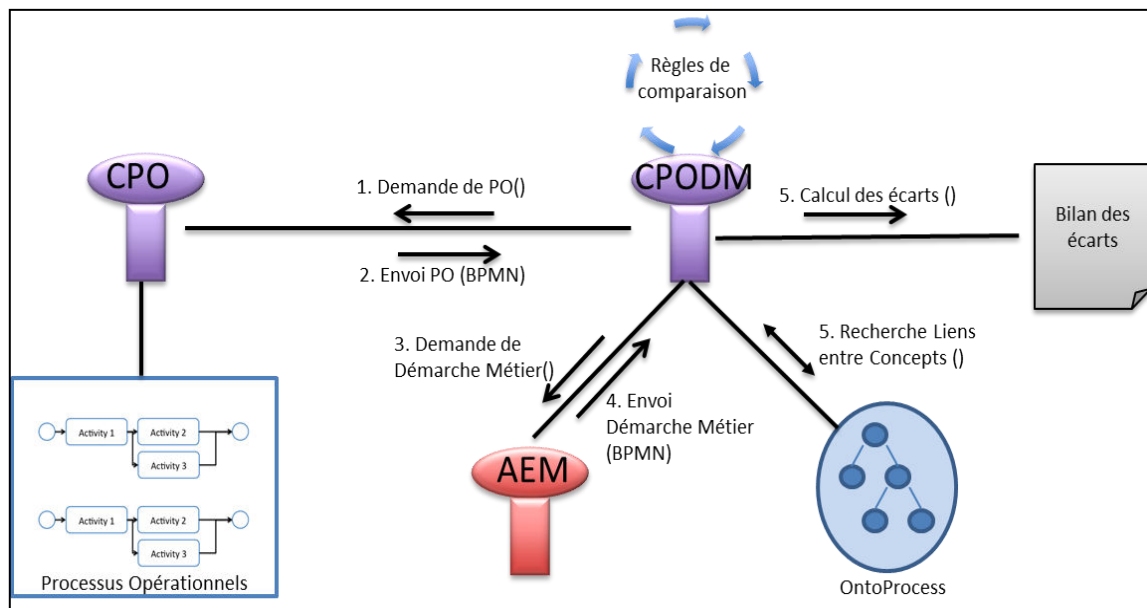


Figure 37. Opérations réalisées par l'agent CPODM

4.3.2.5 Le rôle « Assistant Expert Méthodologique »

L'agent ayant pour rôle Assistant Expert Méthodologique (AEM) a pour objectif d'assister les experts métier à formaliser et construire les démarches métier. Il existe un agent AEM pour tous les projets de l'entreprise.

Organisation	Assister les utilisateurs et Aide à la Décision
Rôle	Assistant Expert Méthodologique (AEM)
Objectifs	Construire les démarches métier par rôles d'acteur métier
Interaction	Agent APM, Expert Métier (Humain)
Compétences	Maitrise du langage BPMN Représenter graphiquement le processus
Connaissances	Langage BPMN Modèle conceptuel des activités métier (ontologie OntoProcess)
Interface	Interface de création de démarche métier
Connexions	...
Nombre	1 agent EM pour tous les projets

Tableau 9. Carte d'identité du rôle « Agent Expert Méthodologique »

L'agent AEM possède une interface permettant aux experts métier de l'entreprise de spécifier la démarche métier utilisée dans chacun des types de projets. L'agent utilise les concepts décrits dans l'ontologie OntoProcess pour guider l'utilisateur à créer des activités, liés un rôle ou plusieurs rôles à une activité, associer des entités informationnelles à une activité, et définir l'enchaînement des activités. L'agent utilise le format BPMN pour construire la démarche métier (Figure 38).

L'agent AEM gère les différentes démarches métier de l'entreprise. Il existe une démarche métier par type d'étude.

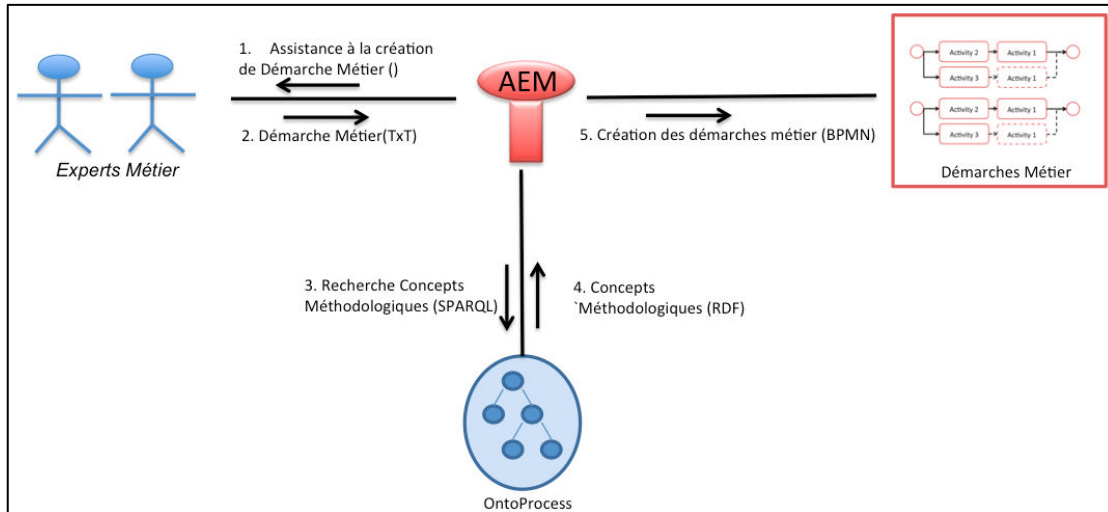


Figure 38. Opérations réalisées par l'agent « Assistant Expert Méthodologique »

4.3.2.6 Le rôle « Assistant Processus Métier »

Il existe un agent Assistant Processus Métier (APM) pour chaque rôle des acteurs métier. L'objectif de cet agent est d'assister à la réutilisation des démarches métier dans l'entreprise.

Organisation	Assister utilisateurs et Aide à la décision
Rôle	Assistant Processus Métier (APM)
Objectifs	Assister les acteurs métier à appliquer leurs processus
Interaction	Agent AEM, Agent CPO, Agent CPODM, Acteur Métier (Humain)
Compétences	Maitrise du langage BPMN Représenter graphiquement les écarts de processus Stratégie d'Aide à la Décision
Connaissances	Langage BPMN
Interface	Interface d'aide à la réalisation des processus
Connexions	...
Nombre	1 agent APM par rôle des acteurs métier

Tableau 10. Carte d'identité du rôle « Assistant Processus Métier »

L'agent Assistant Processus Métier communique avec l'agent AEM pour obtenir la démarche métier correspondant au projet en cours, avec l'agent CPO pour récupérer le processus opérationnel utilisé par l'acteur métier et avec l'agent CPODM pour récupérer les écarts entre la démarche métier et le processus opérationnel en cours.

Il existe un agent APM pour un acteur métier. L'agent monitoré les actions de l'acteur métier pour détecter le contexte organisationnel (nom du projet, activité conduite et rôle de l'acteur). Lorsqu'il a détecté le contexte, il est en mesure d'avertir l'acteur métier lorsqu'il réalise des activités ou un enchaînement d'activités différent de celui spécifié dans la démarche métier.

L'agent possède plusieurs stratégies d'aide à la décision lui permettant de savoir à quel moment il peut aider les acteurs métier (Figure 39). Il est en mesure de représenter les écarts entre le processus opérationnel réalisé par l'acteur métier et la démarche métier en utilisant le formalisme BPMN.

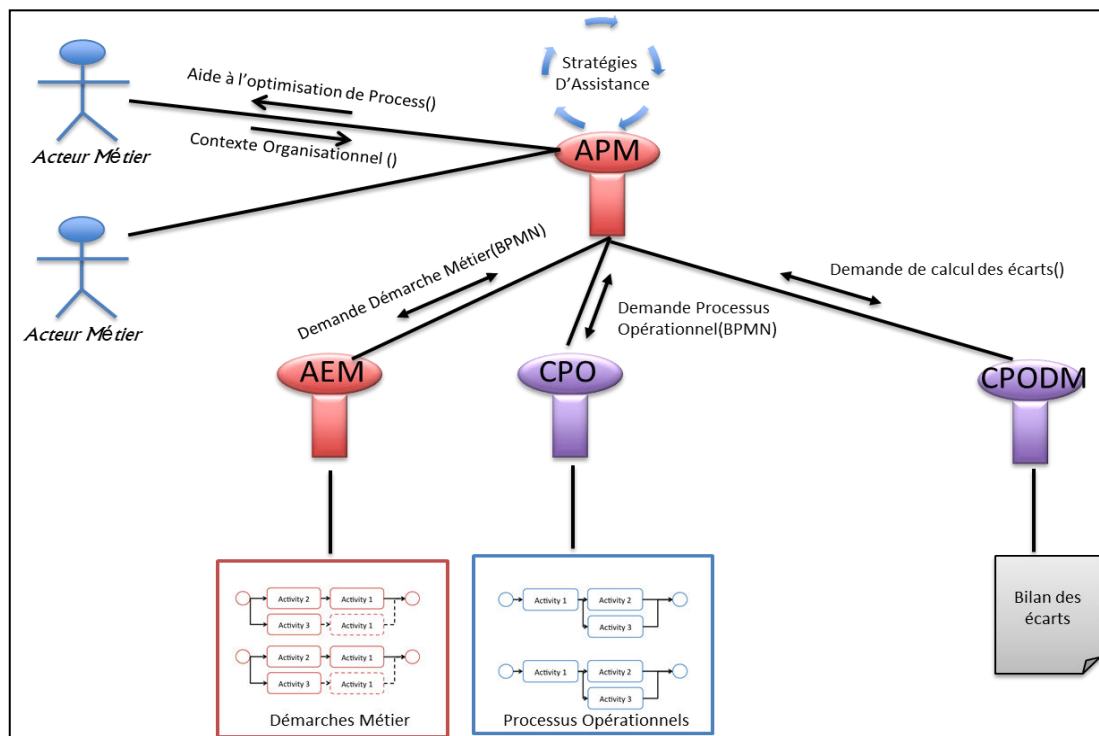


Figure 39. Opérations réalisées par l'agent « Assistant Processus Métier »

Dans la section suivante nous expliquons comment les agents évoluent à l'intérieur des organisations pour accomplir leurs objectifs et assurer l'objectif global du système.

4.3.3 Synthèse et comparaison des propriétés des rôles du système multi-agents

Le Tableau 11 présente les caractéristiques attribuées aux agents d'après la littérature. La première propriété des agents est d'être autonome.

Notre système démarre à partir d'un programme « Lanceur » qui active l'ensemble des agents. Chaque rôle est autonome, c'est-à-dire que chaque agent peut démarrer sa séquence d'action à réaliser. Les agents CPODM et APM ont besoin d'information pour réaliser l'ensemble de leurs séquences d'actions, ils démarrent donc leurs activités en interrogeant les autres agents du système en attendant de recevoir les réponses de ces derniers.

La seconde propriété est la réactivité. Dans notre système les agents ARAO, CPO, CPODM et AEM s'échangent des informations en fonction de leurs objectifs. Chaque agent sollicite un autre agent qui lui répond. De ce fait, les agents sont réactifs. L'agent Assistant Processus Métier appartient à une configuration différente puisqu'il réagit aux sollicitations

de l'utilisateur du système lorsque ce dernier souhaite obtenir une assistance à la réutilisation des démarches métier.

La propriété suivante est la pro-activité. Seul l'agent Assistant Processus Métier est pro actif. En effet il est le seul à proposer une assistance aux utilisateurs pro-active en détectant leurs contextes organisationnels (rôle de l'utilisateur, projet et activité en cours).

La quatrième propriété est la faculté des agents à communiquer les uns avec les autres. Cette faculté est décrite comme l'habilité sociale dans la littérature. Tous les agents du système communiquent les uns avec les autres pour s'échanger les informations dont ils ont besoin pour réaliser leurs actions. Le système utilise une configuration organisationnelle, les habilités sociales des agents sont décrites dans les organisations présentées dans la section suivante.

La cinquième propriété est la faculté à gérer ses objectifs. En effet chacun des 6 agents possède un ou plusieurs objectifs qu'il doit accomplir pour finaliser l'ensemble de ses actions. La configuration organisationnelle impose également aux agents de remplir les objectifs des organisations en complément de ses objectifs. Sless et al. (Sless, Hazon, Kraus, & Wooldridge, 2014) expliquent qu'un agent est versatile lorsqu'il traite plusieurs objectifs simultanément. C'est le cas de nos agents qui doivent gérer les objectifs des organisations auxquels ils appartiennent en même temps que leurs propres objectifs.

La représentation visuelle est utilisée par l'agent Assistant Expert Méthodologique et l'agent Assistant Processus Métier. Cette propriété signifie que ces deux agents sont dotés d'une Interface Homme Machine pour interagir avec l'utilisateur. L'agent AEM possède une interface pour que les experts métier puissent construire les processus de référence de l'entreprise c'est-à-dire les démarches métier. L'agent APM possède une interface qui lui permet de proposer des processus optimisés, c'est-à-dire des processus opérationnels enrichis en fonction des démarches métier. Ces démarches métier guident les utilisateurs lors de l'exécution de leurs processus opérationnels.

Un agent est cognitif lorsqu'il est en mesure d'expliquer aux autres agents ses intentions, ses objectifs ou ses plans d'exécution aux autres agents. Dans notre système, tous les agents, mis à part l'agent Extracteur de Traces, envoient des demandes à d'autres agents pour obtenir des informations dont ils ont besoin pour réaliser leurs actions. Cependant ils ne décrivent pas leurs intentions aux autres agents. On ne peut donc pas considérer que ces agents sont cognitifs.

La propriété suivante est le fait qu'un agent soit adaptatif c'est-à-dire qu'il puisse apprendre et puisse adapter ses actions en fonction des changements dans l'environnement dans lequel il se trouve. L'agent Extracteur de connaissances adapte ses requêtes en fonction des différentes sources d'informations pour extraire les traces. D'autre part, l'agent Assistant Processus Métier propose des processus opérationnels optimisés en fonction des actions des acteurs métier. Il s'adapte donc au contexte organisationnel de l'acteur métier.

Le système utilise une configuration organisationnelle : les agents vont travailler ensemble pour atteindre des objectifs communs. Ils s'entraident donc les uns avec les autres pour remplir les objectifs des organisations. De ce fait ils sont tous bénévoles (propriété 9) pour aider d'autres agents.

La propriété suivante est la rationalité. Chaque agent du système possède un ou des plans d'action bien définis en fonction de leurs objectifs ou des objectifs de l'organisation.

La propriété suivante est la flexibilité, c'est-à-dire que l'agent est capable dynamiquement d'adapter ses actions en fonction des changements de l'environnement. Seul l'agent APM est en mesure de s'adapter en fonction du changement du contexte organisationnel des acteurs métier pour proposer un processus approprié.

Nous avons choisi une configuration organisationnelle. Dans ce cadre, chaque agent va continuer d'assurer ses actions jusqu'à ce qu'il ait atteint ses propres objectifs et ceux de l'organisation à laquelle il appartient. Les agents possèdent donc la propriété de persistance.

Dans certains systèmes multi agents, les agents sont structurés à partir d'une architecture BDI (Beliefs Desire & Intentions). Dans ce genre d'architecture les agents possèdent une 'personnalité' et peuvent même présenter des états émotionnels. Dans notre cas, les intentions des agents sont décrites à travers leurs objectifs mais ils ne possèdent pas de personnalité qui décrirait leurs croyances ou leurs états émotionnels.

La dernière propriété est l'ubiquité qui correspond au fait d'être présent à différents endroits en même temps. Notre plateforme multi agents est un système centralisé où tous les agents s'exécutent sur le même serveur. Seuls les agents Assistant Expert Méthodologique et Assistant Processus Métier peuvent être considérés comme ayant la propriété d'ubiquité puisqu'ils communiquent avec les utilisateurs à travers une interface qui peut être visualisée à travers une application Web sur n'importe quel outil multimédia (PC, tablette ou smartphone).

	Extracteur de Traces (ET)	Analyseur et Rédacteur d'Annotations Organisationnelles (ARAO)	Constructeur de Processus Opérationnel (CPO)	Comparateur Processus Opérationnels et Démarche Métier (CPODM)	Assistant Expert Méthodologique (AEM)	Assistant Processus Métier (APM)
Autonomie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Réactivité	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Pro activité						<input type="checkbox"/>
Habilité sociale (habilité à communiquer)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Habilité à gérer ses objectifs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Représentation visual (IHM)					<input type="checkbox"/>	<input type="checkbox"/>
Cognitif						
Adaptatif	<input type="checkbox"/>					<input type="checkbox"/>
Bénévole	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Rationnel	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Flexibilité						<input type="checkbox"/>
Persistant	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Personnalité						
Ubiquité					<input type="checkbox"/>	<input type="checkbox"/>

Tableau 11. Propriétés des agents de notre SMA

4.3.4 Niveau organisationnel et communication

Afin de structurer le système, nous présentons en détail les trois principales organisations d'agents.

Une première organisation a pour objectif de gérer les traces issues du suivi des activités réalisées par les acteurs métier.

4.3.4.1 L'organisation « Gestion des Traces »

L'organisation « Gestion des Traces » a pour objectifs d'extraire les traces et de les capitaliser. Dans cette organisation nous avons défini deux rôles pour les agents : le rôle d'extracteur de traces et le rôle d'analyseur & rédacteur d'annotations organisationnelles. Le résultat obtenu dans cette organisation est la constitution de la base de traces au format RDF.

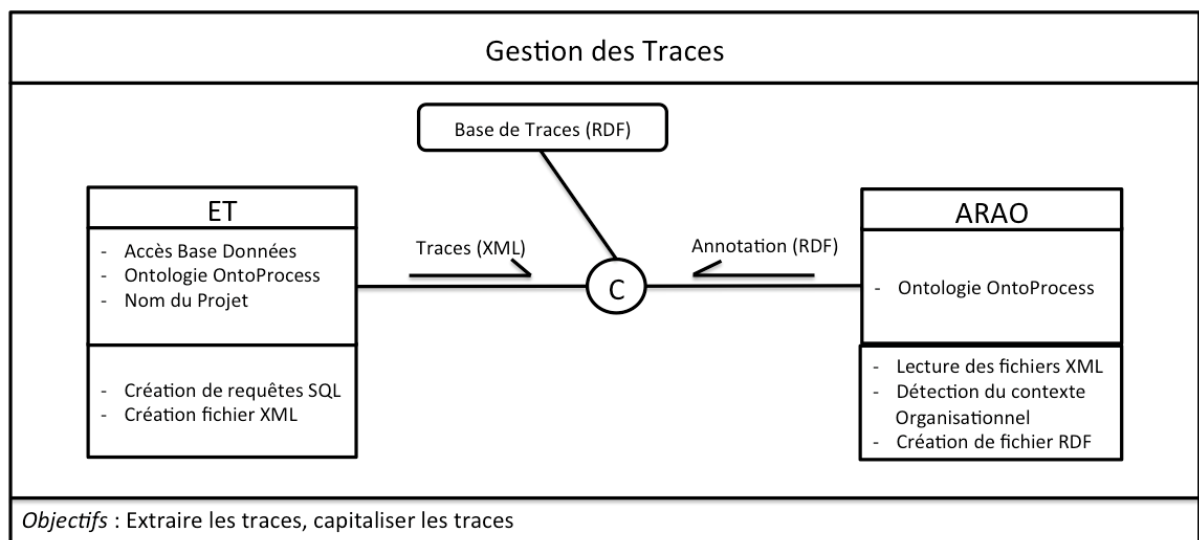


Figure 40. Organisation « Gestion des traces »

Le rôle « Extracteur de Traces » (ET) est dédié à l'extraction des traces à partir du suivi des activités métier. Le second rôle est dédié à l'analyse des traces et à leurs annotations. L'agent Extracteur de Trace a besoin des connaissances concernant l'Accès aux bases de données où sont stockées les traces, du nom du projet et de la spécification des concepts décrivant les entités stockées dans les traces. Cette dernière connaissance est déterminée par l'ontologie OntoProcess (cf. chapitre 3). L'agent ET possède donc les compétences de créer des requêtes SQL permettant les extractions de traces et de créer des fichiers XML (à partir de la structure de l'ontologie OntoProcess) pour stocker les traces.

Le second rôle est celui d'« Analyseur et Rédacteur d'Annotations Organisationnelles » (ARAO). Dans cette organisation l'agent ARAO a besoin de la définition des concepts et de leurs relations décrits dans l'ontologie OntoProcess pour annoter les traces capitalisées. Les compétences de ce rôle sont la lecture des fichiers XML, la détection du contexte organisationnel (nom du projet, rôle de l'acteur, entité informationnelle utilisée) et la création d'annotations dans des fichiers RDF.

L'interaction entre les deux rôles est de type « Coalition » : les agents n'ont pas les mêmes objectifs mais collaborent pour satisfaire les deux objectifs de l'organisation. En effet l'agent ET a pour objectif d'extraire les traces et de les formater/sauvegarder dans des fichiers XML. En satisfaisant ses propres objectifs, l'agent complète également l'objectif de l'organisation qui est d'extraire les traces. De même, lorsque l'agent ARAO a complété ses deux objectifs qui sont l'analyse des traces et leurs annotations et archivage, il complète également le second objectif de l'organisation.

Le diagramme de séquence présenté en Figure 41 décrit l'enchaînement des communications entre les deux agents, l'ontologie OntoProcess et la suite logicielle TDC où sont enregistrées les traces d'activité. Nous ne détaillons pas ici les méthodes utilisées dans la programmation des agents mais nous les citons. On observe que dans le cadre d'une coalition entre les deux agents, les objectifs de l'organisation sont atteints l'un après l'autre. En effet, l'agent ET capture les traces d'activité des acteurs métier et les formate en fonction de la structure (noms des concepts et leurs liens) définie dans l'ontologie OntoProcess. Nous présentons en détail le modèle de traces d'OntoProcess à la section 5.3.1. Ce travail permet d'atteindre le premier objectif de l'organisation. Le second objectif est atteint lorsque les agents ont échangé les traces et que l'agent ARAO les a analysées et annotées.

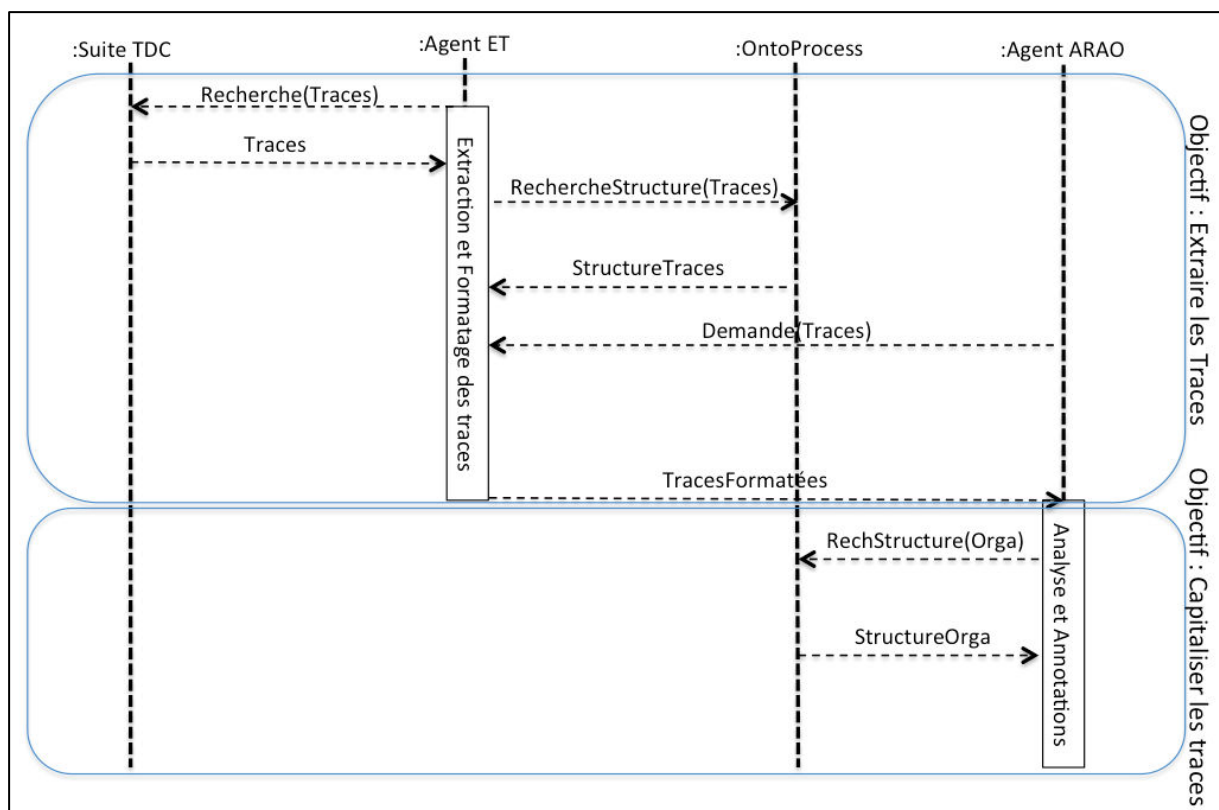


Figure 41. Diagramme de séquence de l'organisation « gestion des traces »

4.3.4.2 L'organisation « Gérer les Processus Opérationnels & les Démarches Métier »

L'organisation « Gérer les Processus Opérationnels & les Démarches Métier » est composée de deux rôles : le rôle de constructeur de processus opérationnel (CPO) et le rôle de comparateur de processus opérationnel et de démarche métier (CPODM).

L'agent CPO a pour objectif d'analyser les traces des activités afin de construire les processus opérationnels sous-jacents. Les processus opérationnels sont construits pour chaque rôle (chef de projet, concepteur, ergonomes, ingénieur méthode etc.) ayant participé au projet. Il existe donc autant d'agent CPO que de rôles d'acteur métier dans le projet.

L'agent CPODM compare les processus opérationnels avec les démarches métier. Les écarts seront ensuite utilisés pour aider les utilisateurs à optimiser leurs processus opérationnels.

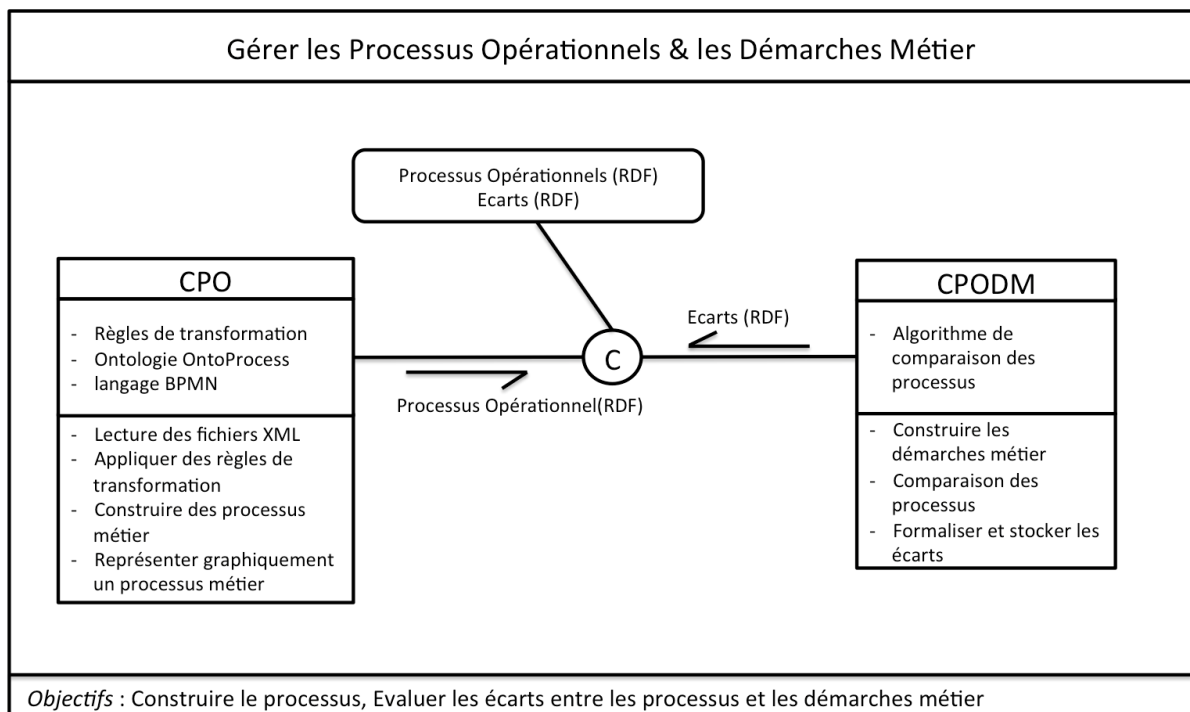


Figure 42. Organisation « Gérer les Processus Opérationnels & les Démarches Métier »

La Figure 42 présente l'organisation « Gérer les Processus Opérationnels & les Démarches Métier » avec les deux rôles d'agents en interaction. L'interaction est une coalition; les deux rôles partagent au moins un objectif commun avec l'organisation. En effet, l'un des objectifs du CPO est de construire le processus opérationnel et l'un des objectifs du CPODM est d'évaluer les écarts entre processus et démarche. Ces deux objectifs sont ceux de l'organisation.

La Figure 43 décrit les communications entre les agents.

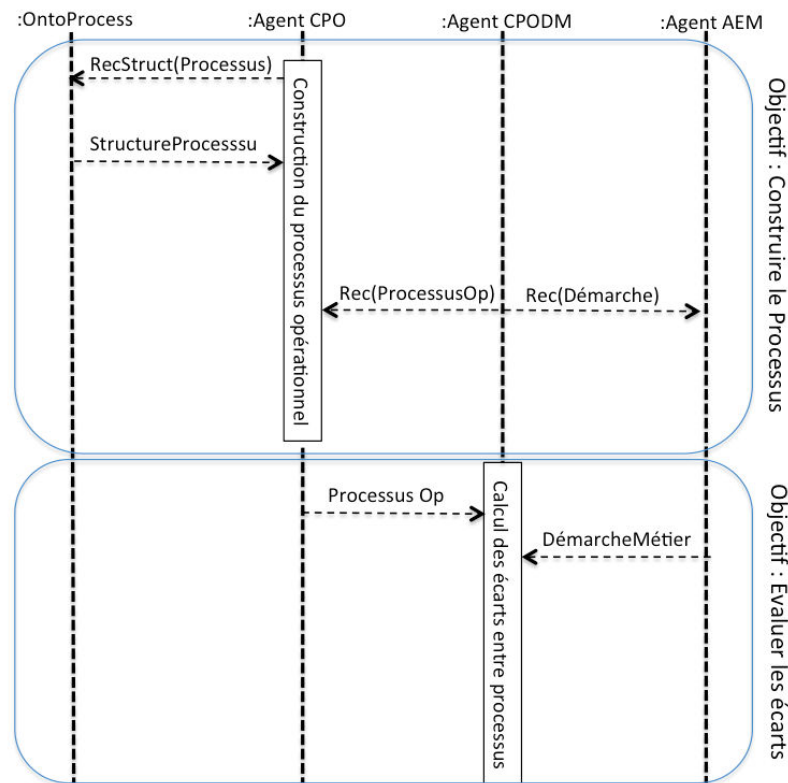


Figure 43. Diagramme de séquence de l'organisation

Trois méthodes sont utilisées : la recherche de la structuration d'un processus par l'agent CPO, la demande de Processus Opérationnel et la demande de démarche métier par l'agent CPODM. L'agent AEM n'appartient pas à l'organisation puisqu'il ne travaille pas pour accomplir les objectifs de cette dernière. Il ne fait que transmettre les démarches métier à l'agent CPODM.

4.3.4.3 L'organisation «Assister les utilisateurs et Aide à la décision»

Cette organisation a pour objectif d'assister les utilisateurs à la création des démarches métier (processus de référence) et à la réutilisation des processus opérationnels. En effet, les agents doivent guider les utilisateurs en proposant un séquençement des tâches à accomplir lorsque ces derniers travaillent sur leurs projets.

L'organisation «Assister les utilisateurs et Aide à la décision» est composée de deux rôles ; l'Assistant Expert Méthodologique qui est dédié à l'assistance à la création de démarches métier et l'Assistant Processus Métier qui a pour objectif de guider l'utilisateur dans l'application de la démarche métier.

L'agent AEM assiste les experts métier pour construire les démarches métier à l'aide d'une interface graphique. Cette interface permet à l'utilisateur de définir les activités, leurs séquençements ainsi que les flux d'informations et les acteurs impliqués. L'agent AEM est en charge de mettre en forme la démarche métier formalisée et de la stocker en suivant le formalisme BPMN.

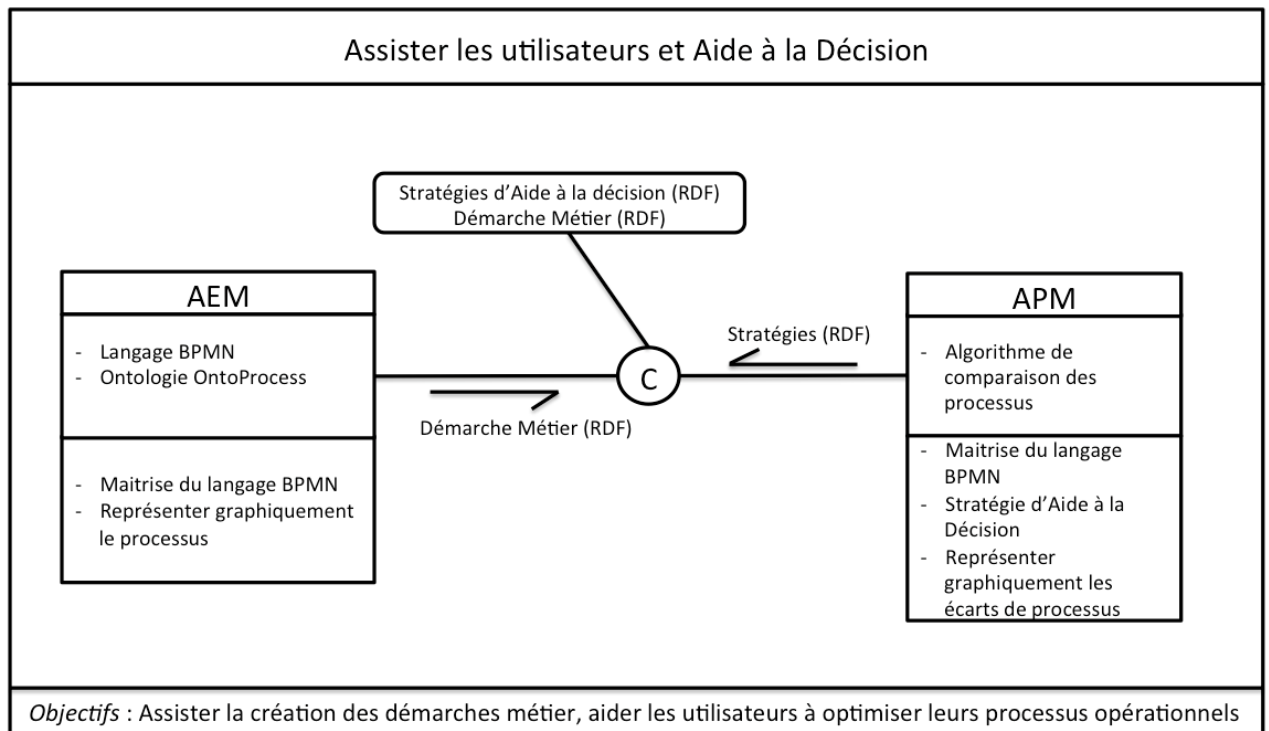


Figure 44. Organisation « Assister les utilisateurs et aide à la décision »

L'agent APM est également doté d'une interface graphique qui lui permet de proposer à l'utilisateur un processus optimisé en fonction de la séquence de travail que ce dernier est en train d'accomplir.

La Figure 44 présente les communications entre les acteurs métier et les agents AEM et APM. Le premier objectif de l'organisation est d'assister à la création de démarches métier. Pour ce faire, l'expert métier utilise l'interface de création des démarches métier gérées par l'agent AEM. L'expert métier peut choisir différents composants pour construire la démarche métier (processus de référence). Ces composants sont décrits par le standard BPMN 2.0 et sont regroupés en 6 catégories :

- Les 'Pool and lanes' pour décrire les niveaux et sous-niveaux du processus,
- Les 'Events' pour décrire les événements qui pourraient se produire tout au long du processus,
- Les 'Activities' pour décrire les activités structurant le processus,
- Les 'Artefacts' pour décrire les sources d'information (base de données, documents, etc.) liées aux tâches,
- Les 'Gateways' pour définir les possibilités d'enchaînement des tâches (séquentiel, parallèle, etc.).
- Les 'Flow' pour décrire le type de relations entre les tâches (séquence, association, etc.)

Lorsque l'expert méthodologique a terminé de choisir les composants pour construire la démarche métier, l'agent AEM utilise ces composants pour générer une version graphique de la démarche métier au format BPMN. Cette version graphique peut ensuite être ajustée par l'expert métier en modifiant certains composants. Après une série d'itérations pour ajuster et valider la démarche métier, l'agent AEM procède à son archivage.

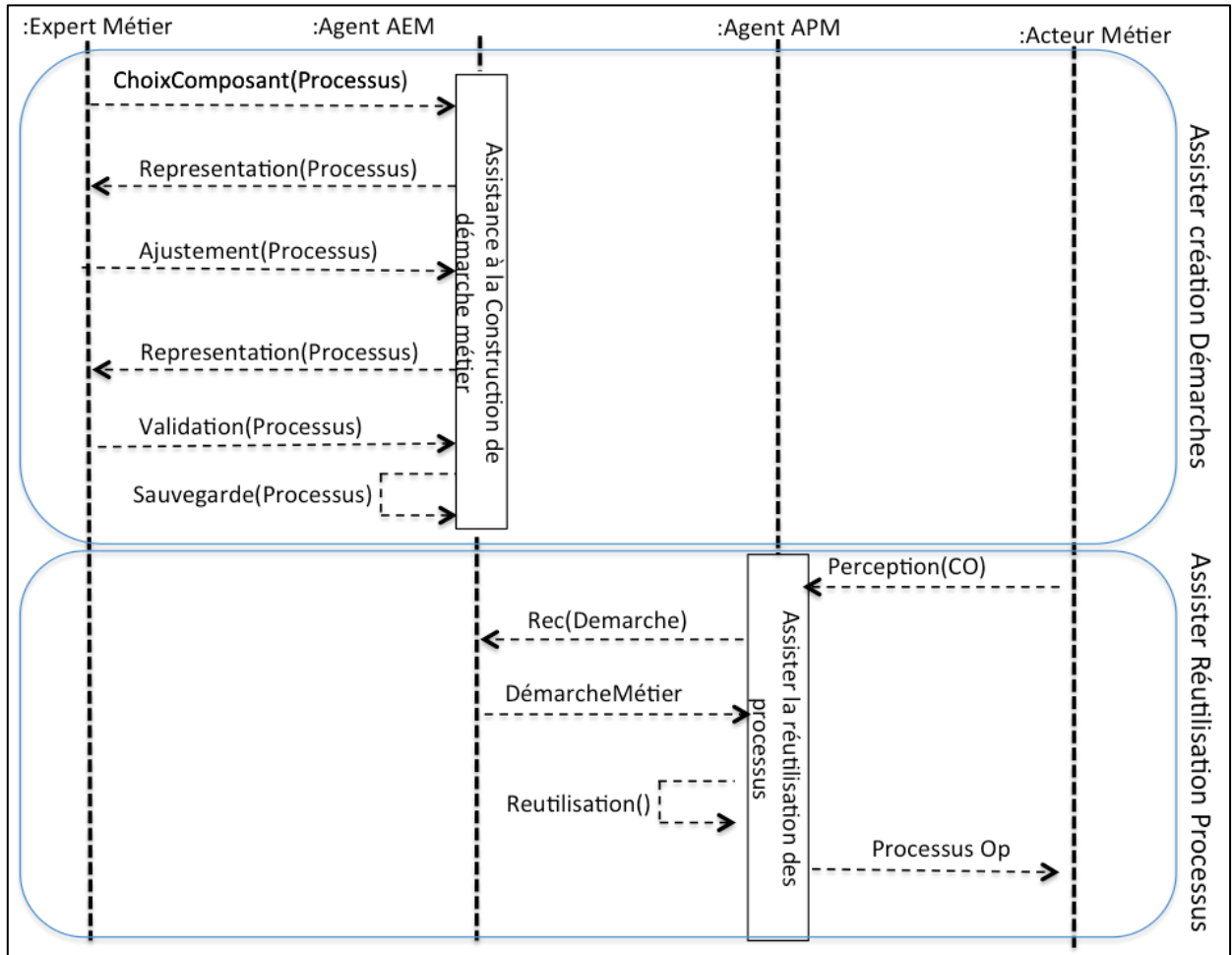


Figure 45. Diagramme de séquence de l'organisation « Assister les utilisateurs et aide à la décision »

Le second objectif de l'organisation est d'assister la réutilisation des démarches métier. Pour ce faire, l'agent APM détecte le contexte organisationnel (projet, rôle, activités) dans lequel l'acteur métier est en train de travailler. L'agent APM demande ensuite à l'agent AEM s'il existe une démarche métier déjà formalisée possédant les mêmes activités que celles du contexte organisationnelle correspondant à l'acteur métier. S'il existe déjà une démarche métier, l'agent APM consulte l'agent CPODM pour connaître les écarts entre le processus accompli par l'acteur et la démarche métier. Lorsque l'agent APM détecte des écarts, il informe l'acteur métier et lui envoie une représentation graphique d'un processus optimisé présentant les possibilités qu'il a pour améliorer son processus opérationnel en cours.

4.4 Conclusion

Dans ce chapitre, nous avons présenté quelques aspects du domaine des systèmes multi-agents ainsi que les méthodes de conception de SMA. Nous avons montré que l'évolution des méthodes de spécification des SMA, notamment celles basées sur des approches organisationnelles où les phénomènes sociaux (rôles, interactions, habiletés...) sont pris en compte, offrent de nouvelles possibilités pour la gestion des connaissances. Nous pensons

qu'une modélisation organisationnelle des processus complexes utilisés par les humains, apporte une plus grande efficacité aux SMA pour gérer les connaissances distribuées et hétérogènes. Ainsi, les sociétés, les rôles, les capacités et les interactions des agents doivent être inspirés des groupes, rôles, compétences et interactions des acteurs métier afin de concevoir un SMA capable d'identifier les connaissances à capitaliser, s'assurer une capitalisation semi-transparente et anticiper les besoins en connaissances des acteurs métier lors de l'exécution de leurs processus.

Nous avons identifié trois organisations d'agents :

- Une organisation dédiée à la gestion de traces et constituée des agents « Extracteur de Traces » responsable de l'extraction des traces et « Analyseur et Rédacteur d'Annotations organisationnel » responsable d'analyser le contexte organisationnel et de créer les annotations organisationnelles.
- Une organisation dédiée à la gestion des processus et des démarches métier et constituée des agents « Constructeur de Processus Opérationnel » responsable de la construction des processus opérationnels et « Comparateur de Processus Opérationnel et de Démarche métier » responsable de calculer les écarts entre les processus opérationnels et les démarches métiers.
- Une organisation dédiée à l'assistance des utilisateurs et l'aide à la décision et constituée des agents « Assistant Expert Méthodologique » qui a pour objectif d'assister à la création des démarches métier et « Assistant Processus Métier » responsable de guider l'acteur métier dans l'application de la démarche métier.

Après avoir présenté l'architecture du système MPPAS, nous présentons dans le chapitre suivant en détail les mécanismes de construction du processus opérationnel à partir des traces des activités métier.

Chapitre 5. Système à base de traces pour la construction de processus opérationnel

Dans ce chapitre, nous détaillons la conception et l'implémentation d'un module du système intelligent proposé : le système à base de traces (SBT) pour la construction d'un processus opérationnel.

La section 5.1 identifie des exigences concernant la construction d'un processus opérationnel.

La section 5.2 décrit l'architecture globale du système à base de traces. Cette architecture est réalisée à partir des architectures des SBT classiques de la littérature et s'appuie sur trois agents décrits précédemment :

- L'agent « Extracteur des Traces » (ET)
- L'agent « Analyseur et Rédacteur d'Annotations Organisationnelles » (ARAO) qui permet d'analyser les traces des activités des acteurs métier et de les annoter, et
- L'agent « Constructeur de Processus Opérationnels » (CPO), qui permet d'exploiter les traces des activités métier pour construire les processus opérationnels pour chaque acteur métier.

Dans la section 5.3, le module de collecte des traces collaboratives est détaillé. En particulier, nous détaillons la partie de l'ontologie OntoProcess qui est relative au modèle des traces et qui est le cœur du module de collecte.

Dans la section 5.4, nous représentons la structure générale du module de transformation du SBT et les règles de transformation mises en place dans ce module.

La section 5.5 présente une comparaison d'outils de visualisation et revient sur les difficultés techniques rencontrées pour implémenter le module de visualisation des processus opérationnels construits.

5.1 Exigences sur la construction du processus opérationnel

Le système à base de traces est au cœur du système intelligent que nous avons spécifié dans le Chapitre 4. Il a pour but d'extraire et analyser les traces numériques générées par les actions des utilisateurs sur les outils méthodologiques afin de construire un processus opérationnel représentant le déroulement réel des tâches et activités d'un projet.

5.1.1 Missions et finalités du SBT

L'une des missions du SBT est d'assurer la construction des processus opérationnels et leur visualisation (ou consultation) par les acteurs métiers et les experts méthodologiques, à

partir des traces enregistrées par l’outil informatique, et en réponse à une requête d’un utilisateur.

Les deux finalités du SBT, assurées au niveau du système intelligent global, est de contribuer à l’enrichissement et la mise à jour des démarches métier, ainsi que l’assistance proactive des utilisateurs.

Un processus opérationnel, partiellement construit en cours de projet, fournit une représentation de l’état des activités des acteurs métier. Il peut contribuer à la proposition d’une assistance aux utilisateurs en faisant référence à la démarche métier.

Le processus opérationnel construit par le SBT n’est pas considéré comme une expérience tant qu’il n’a pas été interprété, analysé et éventuellement validé.

5.1.2 Scénario d’utilisation lié à la demande de visualisation du processus opérationnel

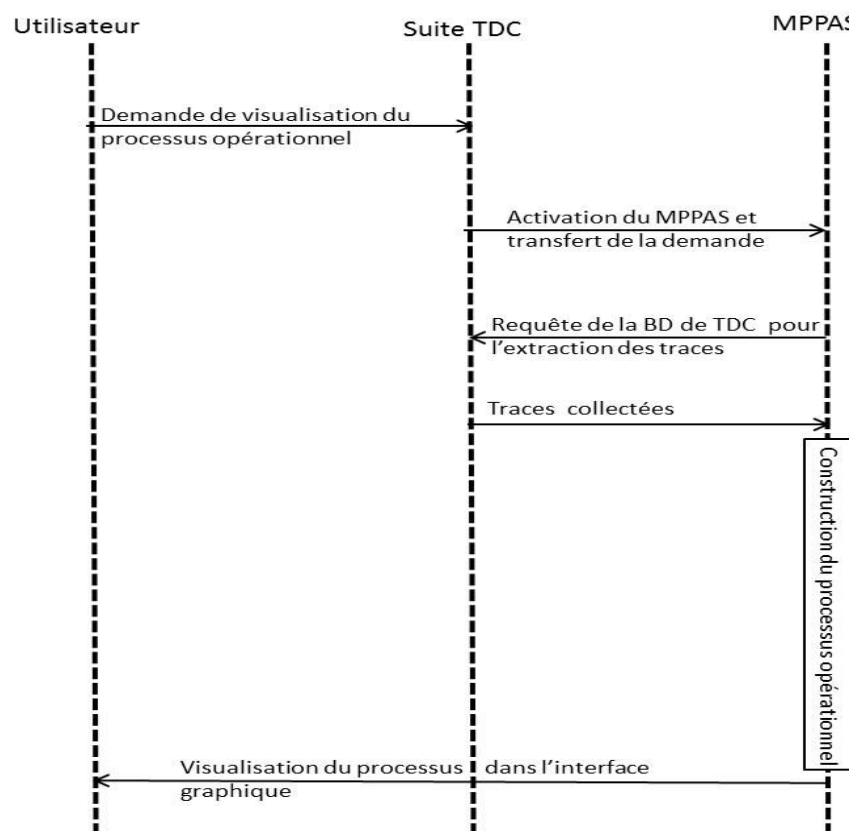


Figure 46. Scénario de construction et de visualisation d’un processus opérationnel

Le scénario d’utilisation (Figure 46) correspondant à une demande de visualisation (et donc de construction) du processus opérationnel par l’utilisateur peut être décrit ainsi :

- Un utilisateur envoie une demande de visualisation du processus opérationnel via une interface graphique de TDC,
- La suite TDC active le système intelligent MPPAS et demande la visualisation
- Le système intelligent envoie une requête à la base de données de la suite TDC pour recevoir les fichiers de trace correspondant à l’étude,
- La suite TDC envoie les fichiers de traces brutes,

- Le système intelligent génère le processus opérationnel et lance sa visualisation dans l'interface graphique.

Une contrainte de durée de génération est à considérer afin d'avoir une réponse dans un temps raisonnable pour l'utilisateur (en moins d'une minute).

Dans les sections suivantes, nous décrivons des exemples d'utilisation du logiciel de conception et nous représentons le processus opérationnel correspondant que nous souhaitons voir construit par le SBT. Ces « cas test » nous permettront de vérifier progressivement que l'implémentation du SBT respecte ces exigences.

5.1.3 Cas test 1

Pierre, ingénieur au sein du département électronique, réalise l'activité d'analyse fonctionnelle dont il est responsable, en utilisant l'outil de spécification (pour nous, TDC Need) et en prenant le rôle de « spécificateur électronique ». Le processus opérationnel construit montre que Pierre a réalisé cette activité en une seule reprise. Dans la représentation du processus opérationnel, nous proposons une synthèse du résultat de chaque activité réalisée. La synthèse du résultat de l'analyse fonctionnelle réalisée par Pierre est présentée sous forme de liste des fonctions de service (ici, fonctions F1, F2, F3, F4,...) que Pierre a identifiées à partir des besoins du client. Les dates où l'activité a commencé et a terminé sont disponibles aussi pour affichage.

L'analyse du processus opérationnel montre que lorsque Pierre a terminé son activité d'analyse fonctionnelle, il a abandonné son rôle de spécificateur électronique et s'est procuré le rôle de « chargé d'étude électronique » pour réaliser l'activité de définition de la structure du produit.

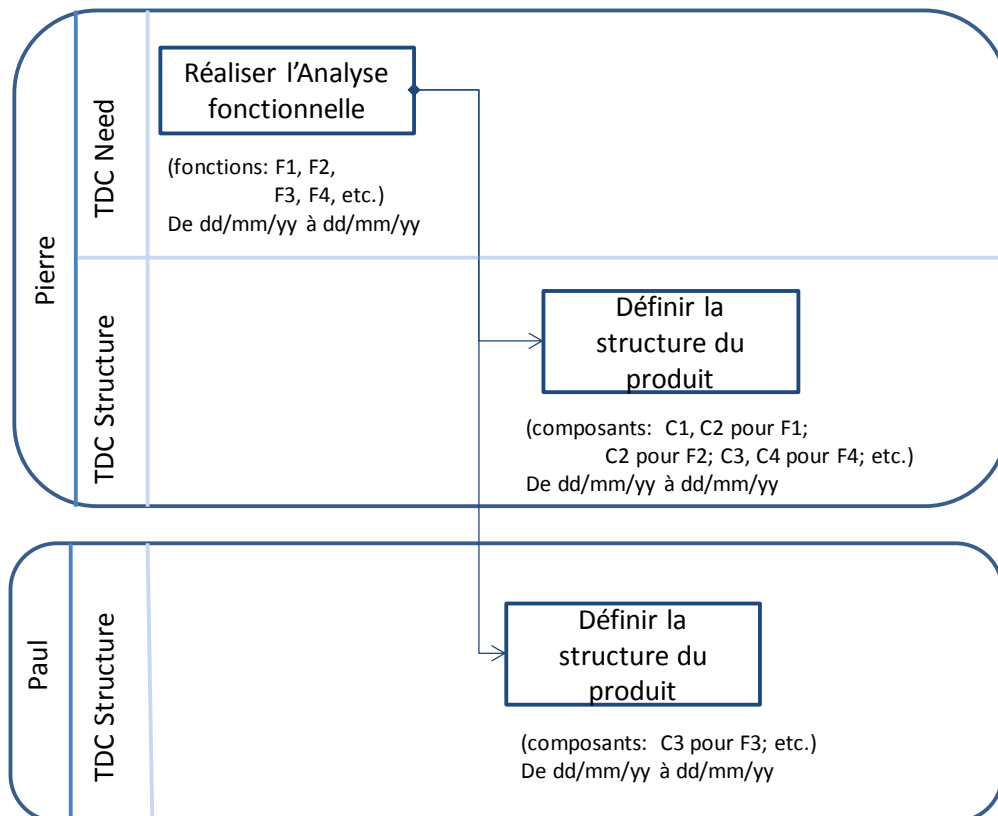


Figure 47. Exemple de processus opérationnel lié au cas test 1

Paul, un technicien du département mécanique, a collaboré avec Pierre pour réaliser cette activité et a pris le rôle de « chargé d'étude mécanique » dans ce projet. La Figure 47 montre le résultat de l'activité de chaque acteur séparément ainsi que la liste des composants créés par chacun. Un tableau d'allocation entre les composants et les fonctions est également généré et visualisé sous la forme C_j pour F_i .

Le positionnement des activités dans l'éditeur de processus respectera les aspects temporels. Comme Paul travaille sur la matérialisation de la fonction F3 qui a été spécifiée préalablement par Pierre, une dépendance logico-temporelle peut être représentée entre ces activités de deux acteurs métier différents. Cependant, il n'y a pas de liens entre les activités « définir la structure du produit » de Pierre et de Paul, ce qui signifie que les activités sont en concurrente. Notons qu'une dépendance peut exister dans la réalité et avoir été traitée lors d'une réunion de travail, n'ayant évidemment pas laissé de traces dans les outils de conception.

5.1.4 Cas test 2

Dans la deuxième démarche, les deux activités « Réaliser l'analyse fonctionnelle » et « Définir la structure du produit » sont réalisées en plusieurs itérations jusqu'à ce que l'on obtienne une définition complète d'une solution qui répond au cahier de charge.

La Figure 48 montre que la définition de la structure du produit ne commence que lorsque l'analyse fonctionnelle a atteint une certaine maturité. Ensuite, chaque itération d'analyse fonctionnelle est suivie d'une itération de définition de la structure du produit.

Il est donc utile de voir le résultat de chaque itération de ces activités. L'analyse du processus opérationnel construit peut révéler la logique suivie dans le projet : par quelles fonctions débute la spécification, quand peut débuter la définition de la structure de la solution, quelles sont les fonctions identifiées ou affinées à la seconde itération, quel est l'affinement de la structure lors de la nouvelle itération ? ...

L'intérêt de cette représentation vient du fait qu'elle montre l'enchaînement réel des itérations des activités. Mais cela peut devenir compliqué d'identifier le bon niveau de granularité pertinent, quand les itérations se multiplient dans une étude.

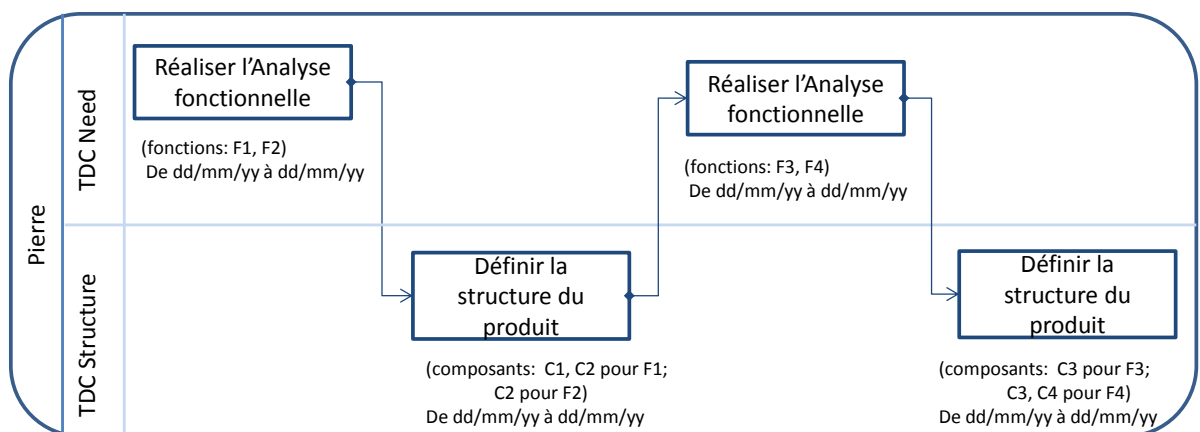


Figure 48. Exemple de processus opérationnel lié au cas test 2

5.1.5 Cas test 3

Dans les deux exemples précédents, nous avons présenté des processus opérationnels simples, avec des activités à un seul niveau de détail.

La Figure 49 montre les tâches réalisées par Pierre dans le projet de conception en tenant compte de deux niveaux d'agrégation des activités tels que définis dans l'ontologie. Dans cet exemple, le processus opérationnel est construit avec plus de détails que dans le premier exemple.

La réalisation de l'analyse fonctionnelle se décline en deux activités. En plus des fonctions créées au cours de la sous-activité « Rechercher les fonctions », la sous-activité « Caractériser les fonctions » permet d'afficher l'ensemble de critères que Pierre a créés pour les fonctions.

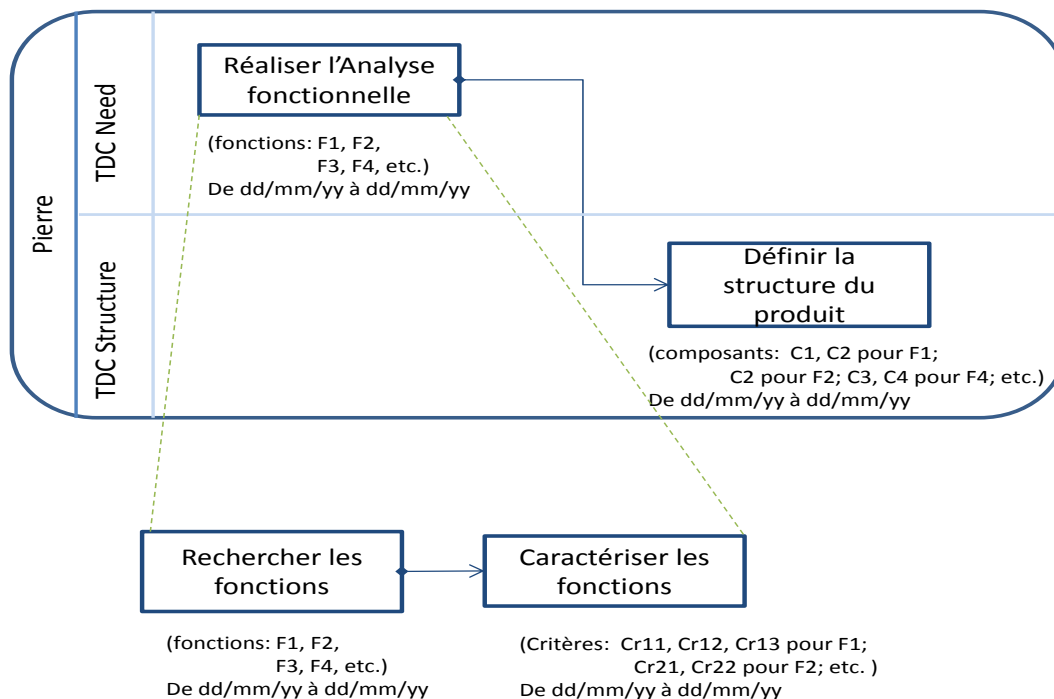


Figure 49. Exemple de processus opérationnel lié au cas test 3

Cet exemple permet d'illustrer un choix concernant l'agrégation de tâches et d'activités : nous utiliserons l'ontologie OntoProcess pour déterminer le niveau supérieur d'une tâche ou activité lorsque nous procéderons à une agrégation. Ce choix provient de notre volonté de conserver un niveau de détail homogène dans une représentation de processus. L'utilisateur pourra facilement accéder un niveau de détail plus fin.

5.1.6 Cas test 4

La représentation du processus opérationnel dans la Figure 50 montre que Pierre a repris la caractérisation des fonctions après avoir défini la structure du produit. Cela peut s'expliquer par le fait que la solution créée n'a pas satisfait tous les critères définis auparavant, ce qui a poussé Pierre à redéfinir les critères en question. La deuxième itération de l'activité « Réaliser l'analyse fonctionnelle » correspond donc à une amélioration de la caractérisation des fonctions.

En construisant le processus opérationnel à partir des traces, il est nécessaire de regrouper les actions tout en s'assurant que les données sur les regroupements ont du sens pour l'utilisateur qui visualise le processus opérationnel construit. Pour cela, le SBT doit :

- conserver les dates de toutes les actions,
- prendre en compte la dernière valeur de l'entité dans l'itération de l'activité construite,
- dans la liste des entités manipulées au cours de l'itération, utiliser des couleurs différentes pour distinguer les entités créées des entités modifiées,
- pour des itérations d'une activité, lister les fonctions modifiées lors de cette activité et accéder à l'état courant de la spécification ou définition (par exemple, sous la forme d'un tableau d'analyse fonctionnelle avec fonctions, critères, niveaux, flexibilité).

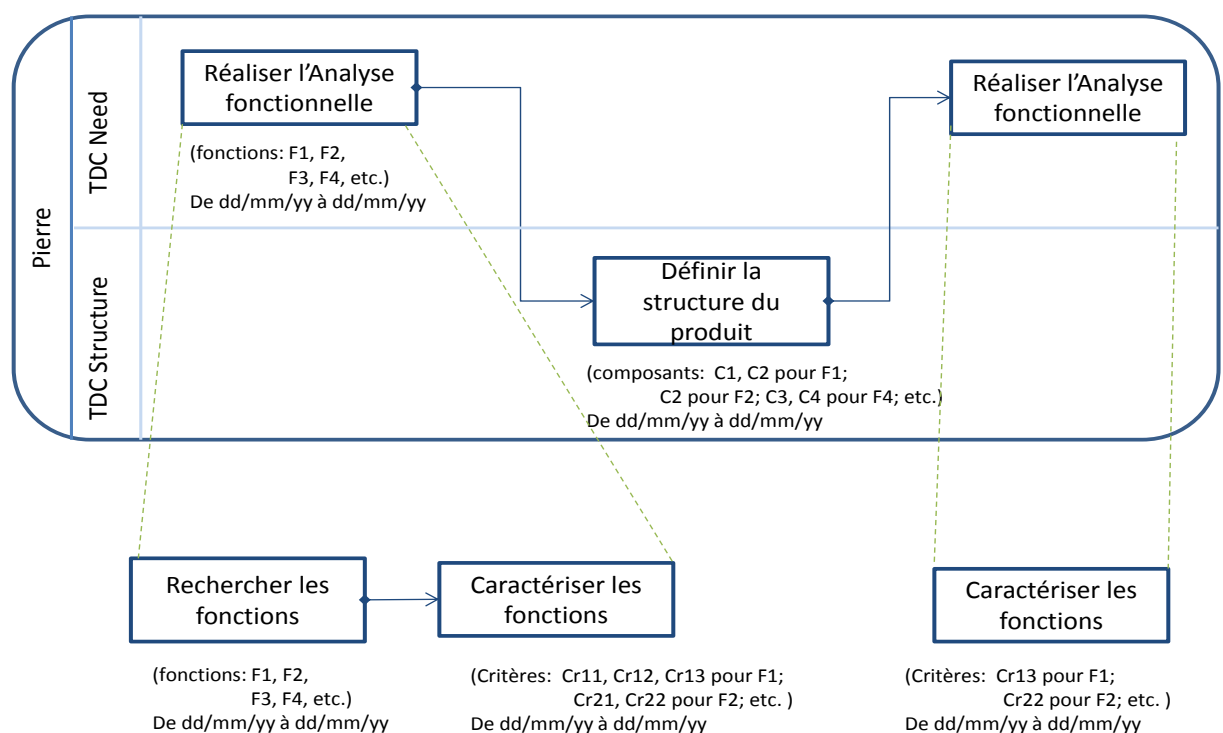


Figure 50. Exemple de processus opérationnel lié au cas test 4

Les quatre cas test présentés dans cette section ont pour objectif d'illustrer les exigences de l'utilisateur concernant la construction du processus opérationnel. Ils permettront de nous guider dans la définition de l'architecture et l'implémentation du SBT. Enfin, ils seront utiles pour vérifier que le SBT développé répond bien à ces exigences basiques.

5.2 Architecture du SBT pour la construction du processus opérationnel

Rappelons d'abord que l'architecture d'un Système à Base de Traces que propose (Settoui et al., 2006a) est composée de trois sous-systèmes (ou modules) : le système de collecte, le système de transformation et le système de visualisation.

Nous avons adapté l'architecture du SBT pour nos travaux. Le système de visualisation permet de visualiser les nouvelles connaissances ou inscriptions de connaissance extraites à

partir des traces, qui portent sur la situation actuelle d'une expérience et se concrétisant par un ensemble d'activités tracées par les SBT.

Les connaissances visualisées par le SBT sont aussi sujet d'analyse et d'interprétations permettant d'avoir des retours à partir des expériences passées. Dans les travaux basés sur le SBT de (M.-H. Karray et al., 2014) , le système de visualisation a été remplacé par un système d'auto-apprentissage qui permet d'inférer de nouvelles règles de connaissances sur les activités des processus gérées par la plateforme de maintenance. Le système d'exploitation que nous proposons comprend un système de visualisation et communique avec un système d'aide à l'enrichissement et un système d'aide à la décision, assurés par d'autres agents.

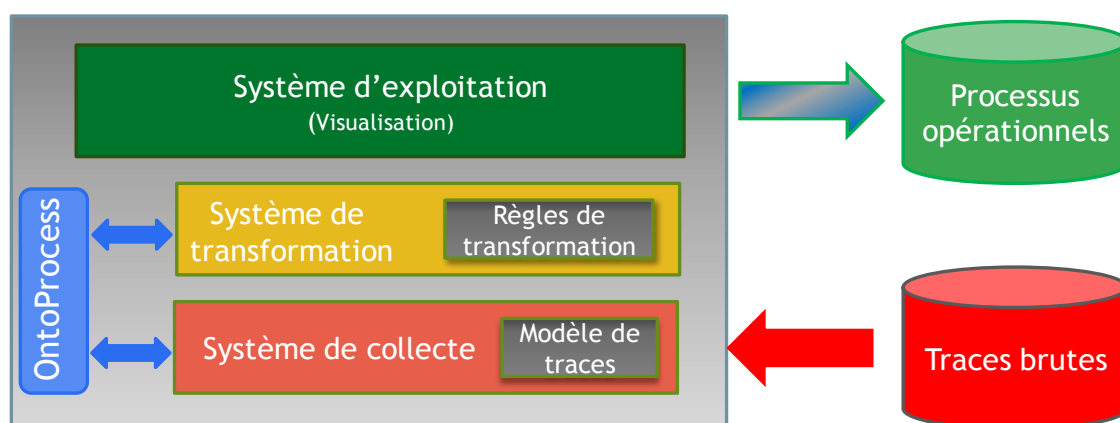


Figure 51. Structure modulaire du Système à Base de Traces

La Figure 51 présente la structure générale d'un SBT. Il comprend 3 modules :

- Collecte des traces,
- Transformation des traces,
- Exploitation des traces.

La collecte des traces débute après une demande de visualisation du processus opérationnel lancée par l'utilisateur depuis l'outil numérique utilisé, dans le cadre de notre travail depuis la suite TDC. Elle est supportée par l'agent extracteur des traces du système intelligent développé. La modélisation des traces est un travail dont on ne perçoit pas directement l'importance et la difficulté. Elle dépend de l'objectif de l'analyse et des connaissances de l'expert qui détermine les aspects de l'activité l'intéressant et lui faisant sens. Le modèle des traces met en évidence les éléments d'intérêt qui vont constituer la trace collectée et permet de les classer ou ranger dans un ordre qui facilitera leur interprétation ou réutilisation par un autre agent. Nos éléments d'intérêt sont les enregistrements historiques des actions informatiques réalisées par les concepteurs sur les données créées au cours du processus de développement du produit via les outils appropriés.

La transformation des traces en processus opérationnel consiste en une analyse des traces à partir de règles de transformation et des ontologies métier présentées au chapitre 3. Elle est supportée par l'agent « constructeur du processus opérationnel » qui génère alors un fichier BPMN. Cet agent a besoin de connaître la syntaxe du fichier des traces, celle du processus opérationnel à générer, les correspondances entre les actions enregistrées dans les traces et les tâches réalisées ainsi que les règles nécessaires pour réaliser les transformations. Une

difficulté réside dans la découverte des relations logico-temporelles entre les tâches et dans les choix d'agrégation (ou composition) des tâches en activités.

Le fichier BPMN généré peut être exploité, afin de visualiser le processus opérationnel à l'aide d'un éditeur graphique de processus acceptant le format BPMN.

L'architecture fonctionnelle de construction et visualisation du processus opérationnel par l'utilisateur peut être décrit ainsi :

- Un utilisateur envoie une demande d'affichage du processus opérationnel via une interface graphique de TDC.
- L'agent ET lance l'export XML des traces dans la suite TDC. Le fichier de traces est stocké dans une base accessible au SMA.
- L'agent ARAO ajoute aux traces les annotations liées au contexte organisationnel.
- L'agent CPO communique avec l'agent ARAO pour obtenir les traces annotées. Il applique l'ensemble des règles de transformations qu'on lui a apprises et construit le processus opérationnel au format BPMN.
- L'agent CPO lance l'édition du processus opérationnel dans l'interface graphique.



Figure 52. Structure fonctionnelle du Système à Base de Traces

La Figure 52 montre les activités de notre système et leur situation dans le SBT. Les sections suivantes détaillent le développement de chacun des modules.

5.3 Le module de collecte des traces collaboratives de conception

Dans cette section, nous présentons comment nous avons spécifié puis implémenté la collecte des traces permettant de construire un processus opérationnel. En effet, au début de notre travail, la suite TDC conservait des traces qui n'étaient pas suffisamment complètes pour permettre une construction performante. De plus, les traces brutes doivent être structurées et annotées pour être utiles au module de transformation. Partant de l'ontologie OntoProcess, nous avons spécifié les données requises dans les traces ainsi que leur structure, pour détailler notre modèle de traces.

5.3.1 Modèle des traces

Pour pallier la complexité de la structure et de l'analyse des traces brutes, nous proposons un module pour la collecte des traces selon un modèle des traces défini en se basant sur l'ontologie OntoProcess.

En effet, les traces des actions impliquent deux points de vue différents : celui de l'acteur de l'action et celui de l'expert qui s'occupera de l'analyse et de l'interprétation des actions.

La modélisation des traces dépend d'une part, de l'objectif de l'analyse et d'autre part, des expériences de l'expert qui détermine les aspects intéressants de l'action pour la construction de la connaissance. Dans ce contexte, la modélisation se déroule de manière itérative où chaque itération produit de nouvelles connaissances qui sont de plus en plus pertinentes, jusqu'à ce que l'on arrive à produire à partir des traces les connaissances que l'on désire analyser.

Ainsi, les traces ne sont pas générées aléatoirement mais résultent de choix et de configurations bien définis. Dans nos travaux, cet aspect itératif s'est concrétisé par deux évolutions de la spécification des traces qui ont abouti à la définition du modèle de traces de la Figure 53.

Le modèle des traces met en évidence les éléments d'intérêt qui vont constituer les traces collectées. Les éléments d'intérêt de nos traces sont les enregistrements laissés sur les actions informatiques réalisées par les acteurs métier lors de l'utilisation d'un module de la suite TDC et portant sur les entités créées, modifiées ou supprimées, au cours d'une étude. Les entités peuvent être les fonctions du produit, sa structure, ses paramètres, ses défaillances potentielles, etc.

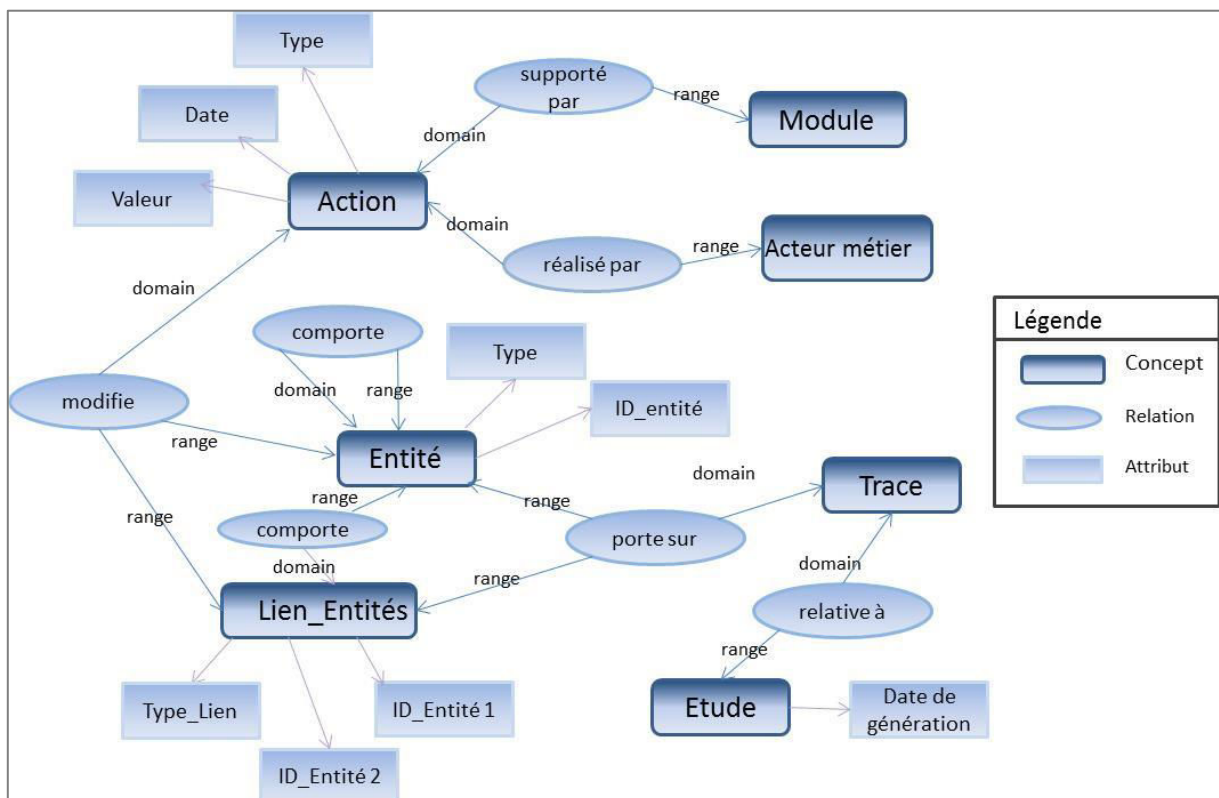


Figure 53. Modèle des traces

La Figure 53 représente le modèle des traces des éléments d'intérêt dans les outils méthodologiques.

Description du modèle des traces

Les *traces* relatives à l'*étude* sont générées par les différents modules de l'outil informatique utilisé. Il est important que ces *modules* soient interconnectés de façon à assurer une cohérence entre les données des traces enregistrées dans l'historique. Le *nom de l'étude* et la *date de génération* sont essentiels pour l'identification de l'historique des *traces*. Cet historique contient des informations sur les *actions* informatiques effectuées au cours de l'*étude* sur les *entités de conception* à travers les différents *modules*.

En effet, une trace porte sur une entité de conception ou sur un lien entre deux entités. Pour chaque entité (ou éventuellement un *lien_entités*), sont tracées les actions informatiques qui l'ont modifiée.

Par exemple, dans le contexte de la suite méthodologique TDC, une action informatique peut être une création d'une nouvelle entité (ou lien), un copier/coller où le concepteur copie et colle une donnée dans un autre endroit de l'étude, un copier/coller à partir des bibliothèques ou de la générique dont disposent les outils TDC, une mise à jour de l'entité ou finalement une action de type suppression. Nous nous intéressons à récupérer la date d'occurrence de l'action, le type de l'action, l'acteur métier qui l'a réalisée, le module dans lequel l'action a été réalisée et la valeur donnée à l'entité modifiée par cette action.

Nous avons élaboré trois tableaux pour présenter les concepts, attributs et relations.

Terme	Concept ID	Parent ID	Définition
Etude	study	--	L'étude réalisée via la suite méthodologique TDC pour un projet de développement
Trace	trace	--	L'enregistrement d'une action réalisée au cours d'une étude
Entité	entity	--	Elément de la base de données sur lequel porte l'action de l'utilisateur de l'outil
Lien_entités	link	--	Lien établi entre deux entités
Action	action	--	Acte informatique élémentaire effectué au cours d'une étude
Acteur métier	author	--	Utilisateur de l'outil numérique, qui participe à une étude
Module	module	--	Module de l'outil qu'un acteur métier utilise

Tableau 12. Les concepts du modèle des traces

Relation	RelationID	ConceptOrigine	ConceptCible	Définition
relative à	relatifTo	Trace	Etude	Les traces sont relatives à une étude
porte sur	concern	Trace	Entité (ou Lien_Entités)	La trace porte sur une entité (ou un lien entre deux entités)
comporte	have	Entité (ou Lien_Entités)	Entité	Une entité ou un lien entité comporte d'autres entités
modifie	modify	Action	Entité (ou Lien_Entités)	L'action a modifié une entité ou un lien entre deux entités
supporté par	supportedBy	Action	Module	Le module avec lequel l'action a été réalisée
réalisé par	realisedBy	Action	Acteur métier	L'action a été réalisée par un acteur métier

Tableau 13. Les relations du modèle des traces

Attribut	Attribut ID	Concept Associé	Attribut Type	Définition
date de génération	dateGen	Trace	Date	Date à laquelle la trace est générée
type_entité	entity_type	Entité	Literal	Chaque entité est typée, elle appartient à une classe de données
type_lien	link_type	Lien_Entité	Literal	Le type de lien est défini en fonction du type des entités liées
première entité liée	entity1	Lien_Entité	Literal	L'ID de la première entité liée via le lien de dépendance
deuxième entité liée	entity2	Lien_Entité	Literal	L'ID de la deuxième entité liée via le lien de dépendance
date_action	action_date	Action	Date	Date d'occurrence de l'action
type_action	action_type	Action	Literal	Une action pourra être une création, mise à jour, etc.
valeur_action	value_action	Action	Literal	C'est la valeur assignée à l'action sur l'entité

Tableau 14. Table de spécifications des attributs du modèle des traces

Il est important d'identifier les *liens* entre les entités de conception tracées. Ces *liens* sont essentiels pour la construction du processus opérationnel. Nous retrouvons des liens de composition/appartenance et des liens de dépendances simples entre les entités.

Les liens entre les « entités »

Nous avons identifié deux types de lien entre les entités :

- Les liens de composition/appartenance, qui permettent de construire un arbre hiérarchique des entités de conception,
- Les liens de dépendance simple qui permettent de tracer, à l'aide de matrices, des dépendances entre entités caractérisant des choix de conception.

Dans cette section, nous expliquons à partir d'exemples la signification de ces liens.

A titre d'exemple, l'Analyse Fonctionnelle Externe (AFE) est une méthode qui consiste à réaliser la modélisation fonctionnelle du produit à partir des exigences formulées par le client. C'est la base de l'établissement du Cahier des Charges Fonctionnel. Le vocabulaire utilisé au cours de l'AFE est standardisé dans des normes telles que [AFNOR NF-X-50-150, 1990], [AFNOR NF-X-50-151], [AFNOR EN-1325-1], etc. Il est constitué des phases de vie, des interacteurs, des fonctions de service, des critères d'appréciation, des niveaux, etc. Ce sont les entités de conception à tracer. Entre ces entités, il existe des liens de dépendance ou de composition.

Nous donnons ci-dessous des exemples sur les différents liens que nous pouvons avoir entre les entités de conception de notre modèle.

Identification des liens de composition/appartenance entre les entités

Dans le cadre de la spécification des traces, nous avons construit la taxonomie des entités de conception. Il s'agit d'une arborescence hiérarchique qui permet de déterminer les différents liens de composition/appartenance entre les entités. Dans chaque branche de l'arborescence, la classification de haut en bas implique qu'une entité à un niveau de la branche englobe des entités qui sont en dessous d'elle.

Exemple 1 - lien de composition/appartenance : Lien phase - fonction de service

Dans cet exemple, il existe un lien de composition/appartenance entre phase de vie et fonctions de service. Ce lien se manifeste par le fait que lorsque la phase de vie est supprimée, toutes les fonctions de services qui appartiennent à cette phase sont supprimées. Une fonction de service est spécifiée pour une phase de vie et ne peut pas appartenir à une autre phase de vie. La suppression d'un lien de composition/appartenance, ici, entre une phase et une fonction de service, implique la suppression, au moins, de l'entité « contenue », ici, de la fonction de service concernée. Dit autrement, la suppression d'une phase entraîne la suppression des fonctions de services et des liens qui lui sont reliés.

Exemple 2- lien de composition/appartenance : Lien fonction de service - critère

Habituellement, un critère est considéré comme un attribut d'une fonction de service. Dans notre modélisation, nous considérons un critère comme une entité qui a un lien avec la fonction de service qu'il caractérise. Ce choix vient du fait que plusieurs critères peuvent être

associés à la même fonction de service et que le nombre des critères qui y sont associés n'est pas prédéfini. Cela nous permet de rester homogènes par rapport à notre modèle de traces, puisque les critères seront représentés séparément des fonctions de service. Les actions de l'acteur sur les critères seront ainsi enregistrées dans les traces.

La construction de l'arbre hiérarchique des entités de conception

Pour chaque module, une taxonomie doit être construite pour permettre par la suite à l'agent « Extracteur de Traces » (ET) et l'agent « Analyseur et Rédacteur d'Annotations Organisationnelles » (ARAO) de construire un fichier de traces avec des données classées dans un ordre prédéterminé et logique. Ce classement permettra au développeur du SBT de vérifier la cohérence des données enregistrées, et facilitera le parcours de l'agent « constructeur de processus opérationnel » à travers ce fichier, lorsqu'il recherchera les actions relatives à une entité ou à un lien qui a été modifié(e).

A titre d'exemple, le module TCD Need supporte essentiellement l'analyse fonctionnelle externe. En reprenant les normes AFNOR de l'AFE, nous vérifions que les entités manipulées par TDC Need sont les mêmes, à savoir : phase de vie, fonction de service, interacteur, critère, niveau, flexibilité, etc. A partir du module TDC Structure qui supporte l'analyse fonctionnelle interne, nous identifions les entités suivantes : solution, composant, fonction élémentaire de contact, fonction élémentaire de flux bouclé, opération.

La Figure 54 représente un extrait de l'arbre hiérarchique des entités de conception de l'outil méthodologique TDC.

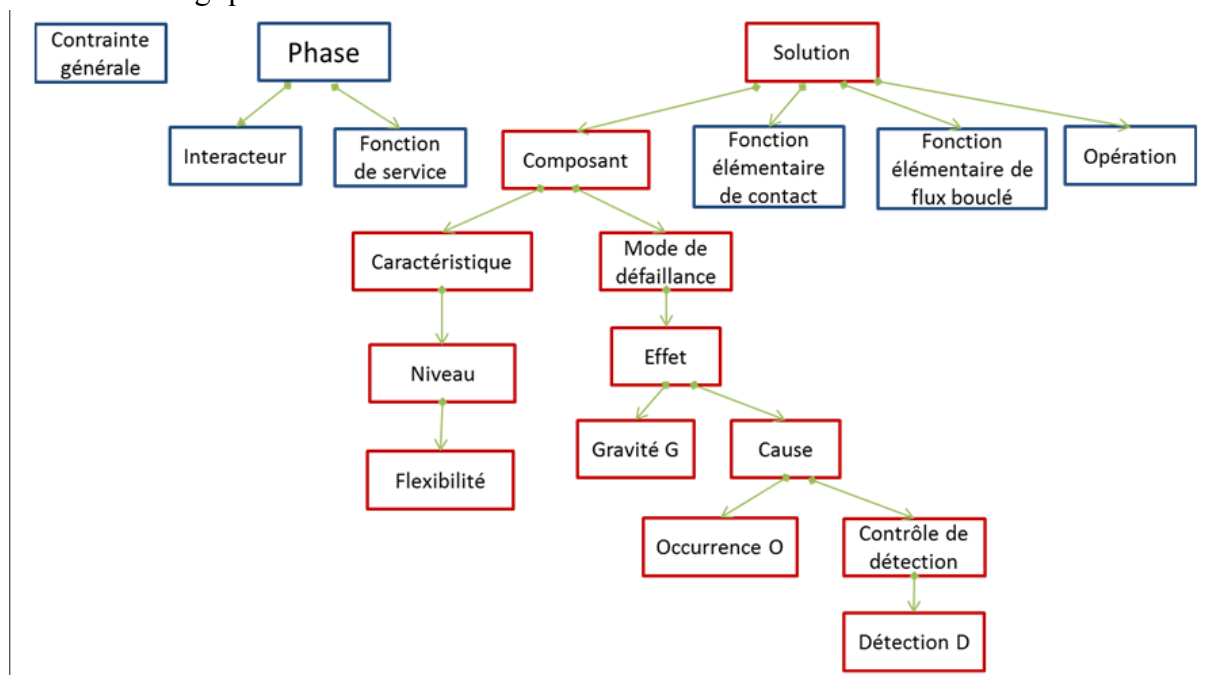


Figure 54. Extrait de l'arbre hiérarchique des entités de conception

Identification des liens de dépendance simple entre les entités

La construction de l'arbre hiérarchique ne suffit pas pour élaborer tous les liens existant entre les entités. Les relations de dépendance simple entre les entités sont représentés dans le modèle par le concept « Lien_entités ». Un lien est une donnée de conception, au même titre

que les entités, on peut lui associer d'autres entités. Au cours d'une étude, il se peut que le concepteur ait besoin de suivre des liens de dépendance entre certaines entités. Le chaînage de ces entités a pour but principal de gérer les impacts des modifications des données.

Exemple 3 - lien de dépendance simple : Lien interacteur- fonction de service

Dans une étude, les entités interacteurs et fonctions de service peuvent être reliées par un lien de dépendance simple. Si une fonction de service est supprimée, les interacteurs qui lui étaient reliés ne sont pas supprimés.

Contrairement aux liens de composition/appartenance, la suppression du lien de dépendance simple n'entraîne pas la suppression de l'entité liée.

Autres exemples de lien de dépendance simple

Nous identifions par exemple les liens composant/fonction de service, composant/fonction élémentaire de flux bouclé, composant/opération.

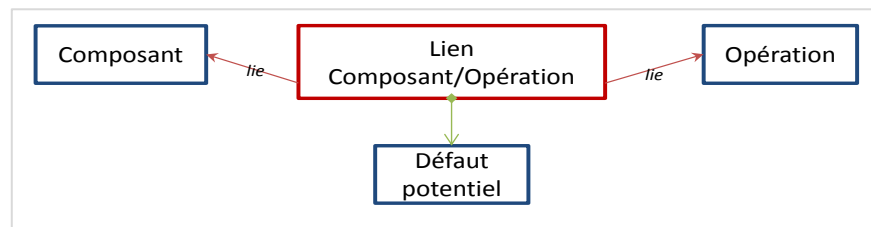


Figure 55. Le type_Lien « Lien Composant/Opération »

La Figure 55 illustre les liens Composant/Opération. Une analyse des modes de défaillances peut être réalisée sur les liens composant/opération. Ainsi, dans les traces, nous retrouverons les défauts potentiels attribués à ces liens.

Après la spécification des traces, nous allons pouvoir les élaborer. Dans la section suivante, nous allons présenter comment nous avons implémenté ce module.

5.3.2 Implémentation de l'extraction des traces XML

Le chapitre 3 et la section précédente ont montré comment les traces sont modélisées. L'étape suivante consiste à implémenter l'élaboration des traces conformément au modèle construit.

Dans la littérature, les traces sont collectées et enregistrées dans des fichiers sous différents formats. Nous citons par exemple la méthode de génération d'un fichier de type tableur avec un évènement par ligne et les propriétés par colonne ou sous forme d'un fichier CSV (Comma Separated Values) dans lequel les évènements sont enregistrés au format texte, séparées par virgule.

Pour nos travaux, nous avons décidé d'élaborer le fichier de traces sous le format XML.

XML est un langage de balisage pour la structuration des données. Il fournit un ensemble de règles qui consistent à utiliser des balises pour ordonner les éléments sous forme hiérarchique. Grâce à l'extensibilité de XML, nous allons élaborer le domaine de données

XML à partir du modèle de traces proposé (Figure 53). Nous allons structurer et poser le vocabulaire et la syntaxe des données qu'il va contenir.

Etant donné que nous traçons les actions sur les entités de l'étude, nous considérons que les éléments au premier niveau de la racine sont les entités.

Le lien existant entre une action et une entité est considéré comme une relation d'appartenance. Pour éviter des redondances de données, nous mettons les entités et non les actions à la racine. La racine ne peut contenir que des éléments de type « entité ».

Un élément entité possède les attributs « ID », « type » et « niveau ». L'« ID » attribue une valeur d'identité unique pour chaque entité. La valeur de l'attribut « type » appartient à la liste de valeurs définie dans la Figure 56. Cet attribut définit le type des entités métiers manipulées et servira donc à identifier l'activité métier correspondante.

Nous considérons ensuite qu'un élément « entité » contient au moins un élément de type « action ». Chaque action possède les attributs « type », « valeur », « auteur », « outil » et « date ». Ils sont ajoutés à l'élément « entité » en fonction de la valeur de l'attribut « date ».

L'élément « entité » peut contenir aussi d'autres éléments de type « entité ». Des restrictions sur l'imbrication des éléments entité sont déterminées en fonction de la valeur de l'attribut « type » relativement à l'arbre hiérarchique des entités de conception (Figure 54) : une entité de type « opération » peut englober des entités de type « synoptique », « machine » mais jamais des entités de type « interacteur ».

```

- <entity id="1931" type="operation" >
  <action type="Trad" value="Laminage" tool="TDC FMEA" author="Etienne Ledoux" version="0-18" date_n="1166565243" date="2006-12-19T22:54:03"/>
  - <entity id="1934" type="NO" >
    <action type="MAJ" value="20" tool="TDC FMEA" author="Etienne Ledoux" version="0-3" date_n="1162541159" date="2006-11-03T09:05:59"/>
    <action type="MAJ" value="" tool="TDC FMEA" author="Etienne" version="0-2" date_n="1160730892" date="2006-10-13T11:14:52"/>
    <action type="Création" value="" tool="TDC FMEA" author="Etienne" version="0-1" date_n="1160726345" date="2006-10-13T09:59:05"/>
  </entity>
  - <entity id="1935" type="SYNOPTIC" >
    <action type="MAJ Bibli" value="" tool="TDC FMEA" author="Etienne" version="0-2" date_n="1160730805" date="2006-10-13T11:13:25"/>
    <action type="Création" value="" tool="TDC FMEA" author="Etienne" version="0-1" date_n="1160726345" date="2006-10-13T09:59:05"/>
  </entity>
  - <entity id="1936" type="MACHINE" >
    <action type="Création" value="" tool="TDC FMEA" author="Etienne" version="0-1" date_n="1160726345" date="2006-10-13T09:59:05"/>
  </entity>
  - <entity id="1937" type="CLASS" >
    <action type="Création" value="" tool="TDC FMEA" author="Etienne" version="0-1" date_n="1160726345" date="2006-10-13T09:59:05"/>
  </entity>
  - <entity id="1938" type="IMAGE_FP" >
    <action type="Création" value="" tool="TDC FMEA" author="Etienne" version="0-1" date_n="1160726345" date="2006-10-13T09:59:05"/>
  </entity>
  - <entity id="15345" type="TYPE_DEFAULT" >
    <action type="MAJ" value="D" tool="TDC FMEA" author="Thierry Beaujon" version="0-2" date_n="1176364344" date="2007-04-12T09:52:24"/>
    <action type="Création" value="" tool="TDC FMEA" author="Thierry Beaujon" version="0-1" date_n="1176364336" date="2007-04-12T09:52:16"/>
    + <entity id="15346" type="MD" >
    + <entity id="15384" type="MD" >
  </entity>
</entity>

```

Figure 56. Extrait du fichier XML de traces des actions portant sur une entité opération et ses entités filles.

Les traces portant sur des liens sont maintenues dans des matrices définies par type de lien (Figure 57). Par conséquent, les liens de type « composant/opération » sont regroupés dans une matrice et les liens de type « fonction/composant » sont regroupés dans une autre.

Un élément de type « lien » est caractérisé par deux attributs qui correspondent aux « entités » reliées par le lien. Leurs valeurs correspondent aux valeurs de l'attribut « ID » des « entités » impliquées.

Une action informatique réalisée sur un lien ne peut être qu'une action de création ou de suppression. Dans le fichier XML, l'action est représentée par un élément « action » imbriqué sous l'élément « lien » correspondant. Les attributs de l'élément « action » sont

identiques à ce qui a été défini précédemment pour une action sur une entité. La seule restriction supplémentaire porte sur les valeurs de l'attribut « type ».

En plus des éléments de type « action », un élément « lien » peut imbriquer des éléments « entité » à condition que la valeur de leurs attributs « type » soit « TYPE_DEFAULT ». Cela permet par exemple, dans une étude, de réaliser des AMDEC sur les liens opération/composant.

Une fois les traces élaborées, la construction du processus opérationnel peut débuter et consiste en une transformation des traces.

```
- <matrix_component_operation hidden="true">
- <link component="864" operation="5886">
  <action type="" value="" tool="" author="Etienne Ledoux" version="0-1" date_n="1162507103" date="2006-11-02T23:38:23"/>
</link>
- <link component="864" operation="6213">
  <action type="" value="" tool="" author="Etienne Ledoux" version="0-1" date_n="1162538496" date="2006-11-03T08:21:36"/>
</link>
- <link component="864" operation="6822">
  <action type="" value="" tool="" author="Animateur ( sans mot de passe )" version="0-1" date_n="1163079332" date="2006-11-09T14:35:32"/>
</link>
- <link component="883" operation="1931">
  <action type="" value="" tool="" author="Etienne" version="0-1" date_n="1160726345" date="2006-10-13T09:59:05"/>
</link>
- <link component="883" operation="1940">
  <action type="" value="" tool="" author="Etienne" version="0-1" date_n="1160726415" date="2006-10-13T10:00:15"/>
</link>
- <link component="883" operation="1949">
  <action type="" value="" tool="" author="Etienne" version="0-1" date_n="1160726437" date="2006-10-13T10:00:37"/>
</link>
- <link component="883" operation="2811">
  <action type="" value="" tool="" author="Etienne" version="0-1" date_n="1161433934" date="2006-10-21T14:32:14"/>
</link>
- <link component="883" operation="3112">
  <action type="" value="" tool="" author="Etienne" version="0-1" date_n="1161442275" date="2006-10-21T16:51:15"/>
</link>
- <link component="883" operation="6505">
  <action type="" value="" tool="" author="Etienne Ledoux" version="0-1" date_n="1162540775" date="2006-11-03T08:59:35"/>
</link>
- <link component="883" operation="6844">
  <action type="" value="" tool="" author="Animateur ( sans mot de passe )" version="0-1" date_n="1163080311" date="2006-11-09T14:51:51"/>
</link>
</matrix_component_operation>
```

Figure 57. Extrait du fichier XML de traces illustrant une matrice de liens composant / opération

Quand les traces ont été élaborées par l'agent Extracteur de Traces dans un fichier XML, l'agent Analyseur et Rédacteur des Annotations Organisationnelles (ARAO) peut les annoter avec le contexte organisationnel. Nous présentons l'implémentation de l'annotation des traces dans la section suivante.

5.3.3 Implémentation de l'annotation des traces

Dans cette section, nous présentons un exemple d'implémentation des règles de transformation à l'aide des requêtes SPARQL. Le module de transformation du système à base de traces que nous proposons est réalisé par l'agent Constructeur de Processus Opérationnel (CPO).

Dans cette section, nous présentons un exemple d'implémentation des annotations organisationnelles des traces à l'aide des requêtes SPARQL. L'annotation est réalisée par l'agent ARAO qui ajoute à chaque action le rôle qui l'a réalisée.

```

SELECT ?Action
WHERE {
    ?Action OntoProcess :estRéaliséPar ?Acteur
    {
        ?ActeurOntoProcess :nomActeur «YIG »
    }
    ///////////////////////////////////////////////////////////////////
CONSTRUCT ( ?r OntoProcess :inferred _ :a
    _ :a a OntoProcess :Role
    _ :a OntoProcess :libellé_role concat ( ? role_générique, ?libellé_métier
        ) As ?role)

WHERE
{ ?acteur OntoProcess :appartientA ?département
  ?département OntoProcess :attachéA ?métier
  ?métier OntoProcess :libellé_métier ?libellé_métier
And
?acteur OntoProcess :réalise ?action
?action OntoPorcess :estSupportéePar ?module
(If ?module = « module_AFE » ?role_générique = "spécificateur"
If ?module = « module_AFI » ?role_générique = "concepteur"
}

```

Figure 58. Requêtes SPARQL

La première requête SPARQL consiste à chercher les actions réalisées par l'acteur métier « YIG » pour les annoter.

La deuxième requête SPARQL ajoute les annotations de rôles sur les actions.

Le rôle est déterminé à partir d'une concaténation du rôle générique `role_générique` déduit à partir du module utilisé par l'acteur et de son métier.

5.4 Le module de transformation

Le module de transformation est le cœur du Système à Base de Traces et constitue une originalité de notre travail de thèse. Nous présentons d'abord dans cette section l'architecture fonctionnelle de ce module où s'alternent des phases de transformations sémantiques et structurelles. Ensuite, nous argumentons notre choix de représentation des processus à l'aide des notations BPMN. Puis nous déterminons les règles de transformation en processus opérationnel. Enfin, nous présentons partiellement l'implémentation de la construction du processus opérationnel par l'agent Constructeur de Processus Opérationnel.

5.4.1 Structure générale du module de transformation

La construction du processus opérationnel à partir des traces passe par des étapes alternées de transformation de la structure et de la sémantique des traces pour aboutir à l'identification des tâches, de leur regroupement dans une activité et de l'enchaînement de ces tâches et activités, à différents niveaux de composition. Cette transformation génère un processus opérationnel interprétable par l'utilisateur, c'est-à-dire décrivant des activités dans son langage métier.

La transformation structurelle vise à construire les éléments de structure (enchaînement) conformément au langage choisi (BPMN).

La transformation sémantique tient compte de la sémantique des données contenues dans les traces et interprétées grâce à l'ontologie métier OntoProcess préalablement construite.

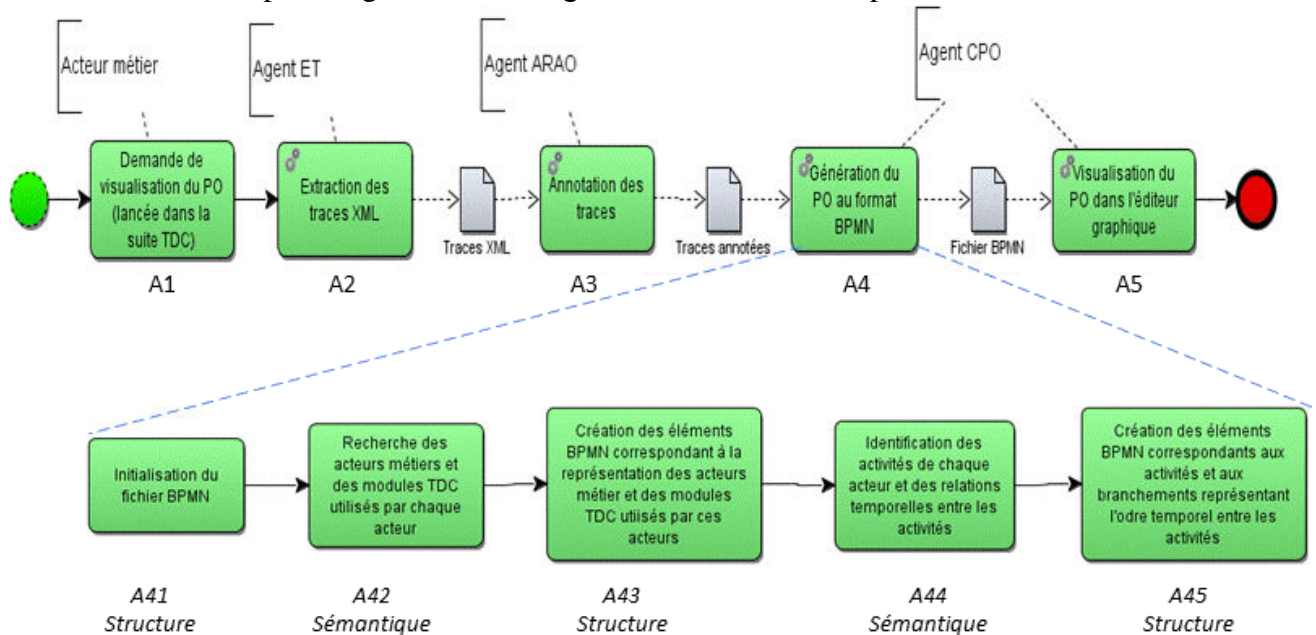


Figure 59. Architecture fonctionnelle du module de transformation des traces en processus

La Figure 59 présente l'architecture fonctionnelle du module de transformation des traces en processus opérationnel. Cette architecture se compose des activités suivantes :

- A41 : C'est l'activité d'initialisation du fichier BPMN. Elle crée les éléments de base communs à tous les fichiers BPMN. Elle est réalisée par la méthode `initiateBPMNFile`. Elle crée l'élément racine « définition », ensuite l'élément « `model:collaboration` » qui servira pour lister les acteurs participant à l'étude, et enfin l'élément « `bpmndi:BPMNDiagram` » qui contiendra la description graphique.
- A42 : C'est une activité de transformation sémantique qui cherche les acteurs qui ont participé au processus opérationnel et les modules qu'ils ont utilisés.
- A43 : L'activité A43 crée les éléments BPMN correspondant aux acteurs et aux outils. A43 crée pour chaque acteur l'élément « `model:participant` » qui sera ajouté à l'élément « `model:collaboration` » et l'élément « `model:process` » représentant l'acteur métier. Elle crée ensuite l'élément « `model:laneset` », l'élément

« model:startEvent » et l'élément « model:endEvent ». Enfin, un élément « model:lane » est créé pour chaque outil.

- A44 : Cette activité identifie les activités des acteurs ainsi que les relations temporelles. C'est le cœur du module de transformation et au niveau scientifique, c'est le verrou à lever. Le principe de la solution repose en premier temps sur la recherche des tâches à partir des regroupements des actions sur les entités de conception dans les traces : les libellés des tâches et des activités supérieures sont retrouvés à partir du libellé « type » des entités et si nécessaire, des entités parentes (voir Figure 22, Tableau 3, Tableau 4).

Les regroupements se font en fonction des exigences déterminées, par exemple s'il est exigé de construire les activités jour par jour ou mois par mois. Des règles de transformation permettent de réaliser ces regroupements et de retrouver les enchaînements logico-temporels entre les activités en analysant les séquencements temporels de ces activités.

- A45 : L'activité A45 crée les éléments BPMN correspondant aux activités de chaque acteur métier et à leur enchaînement, et ce, à différents niveaux de détail. Les éléments BPMN créés par A45 pour un acteur métier sont : un « model:parallelGateway » d'entrée et un « model:parallelGateway » de sortie pour les branchements parallèles, un « model:sequenceFlow » pour représenter les flux de séquences entre les activités, un « model:task » pour une tâche (activité d'un seul niveau) et « model:subProcess » pour les activités de plusieurs niveaux. Ces éléments sont rajoutés à l'élément « model:process » représentant l'acteur métier. Des éléments « model:flowNodeRef » référant les éléments créés (à l'exception des éléments « model:sequenceFlow » qui ne sont pas des nœuds BPMN) sont rajoutés à l'élément « model:lane » représentant l'outil dans lequel se déroulent les activités.

Dans le chapitre 3, nous avons présenté l'ontologie métier OntoProcess.

Dans la section suivante, nous allons présenter les règles de transformation qui ont guidé la construction. Nous exposons d'abord les principes ayant guidé la définition des règles puis nous présentons l'implémentation informatique de ces règles.

5.4.2 Choix d'une représentation des processus en BPMN

Notre choix de présentation des processus opérationnels s'est porté sur la notation BPMN pour deux raisons principales.

La première est que la notation BPMN (Business Process Model Notation) s'appuie sur les formalismes des diagrammes de flux pour la représentation des processus et que cette norme est largement adoptée dans le monde industriel (elle est utilisée dans plus que quatre-vingt outils). Le BPMN dérive des efforts de l'OMG/BPMI pour définir un standard concernant la modélisation des processus d'entreprise. Son but essentiel est de fournir une représentation commune et compréhensible aux différents acteurs de l'entreprise, allant de l'analyste qui crée les ébauches initiales des processus, en passant par le développeur responsable de mettre en place la technologie pour rendre le processus exécutable, jusqu'aux acteurs responsables de gérer et de monitorer le processus.

La deuxième raison est que la notation BPMN est complétée par le langage XPDL (XML Process Definition Language) qui permet de définir un processus métier BPMN à l'aide du

langage XML. En effet, le XPDL est défini à l'aide des schémas XML, c'est le standard qui fournit un format de fichier XML supportant tous les aspects déterminés dans la notation BPMN. Ainsi, le XPDL permet de visualiser un processus métier BPMN à partir d'un fichier XML. En plus, il assure l'interopérabilité sémantique des processus métiers entre les différents outils supportant le BPMN pour des besoins d'affichage, d'exécution ou de gestion du processus, etc. Cependant, c'est le BPMN Diagram Interchange (BPMN DI) qui permet de définir la représentation graphique des processus métier. Selon l'éditeur des processus BPMN choisi, un fichier XML peut être constitué d'une partie sémantique, définie selon le standard XPDL, qui détermine les différents éléments d'un processus et les relations qui existent entre ces éléments et d'une autre partie concernant la description graphique de ces éléments. La partie BPMN DI n'est pas toujours nécessaire dans le fichier pour que l'éditeur BPMN affiche correctement le processus.

5.4.3 Définition des règles de transformation des traces en éléments BPMN

Pour construire le processus opérationnel en BPMN, nous avons défini un ensemble de règles de transformation portant sur la structure du processus en s'appuyant sur les entités de modélisation de BPMN.

Nous avons choisi le concept de collaboration. Une collaboration est formée de plusieurs pools et chaque pool représente un participant (entité métier) de la collaboration. Les collaborations sont destinées à représenter les interactions entre deux ou plusieurs entités métiers. Les interactions entre les participants sont représentées par les flux de messages échangés. Cependant, dans nos travaux, les traces ne contiennent pas d'informations sur les interactions entre les acteurs métiers, la construction automatique des interactions à partir des traces n'est pas donc réalisable. Cependant, une collaboration permet de représenter séparément les activités de chaque acteur métier.

Avant d'appliquer les règles de transformation pour la construction du processus opérationnel à partir des traces, le fichier BPMN contient un élément «model:definitions» à la racine. Puis ce fichier est complété par un élément «model:collaboration» qui contiendra la structure du processus et un élément «bpmndi:BPMNDiagram» qui contiendra la description graphique du processus.

La règle R1 consiste à chercher, dans les traces, les différents acteurs métiers ayant participé à l'étude. Chaque acteur métier est représenté par un «Pool», ainsi la règle R1 crée pour chaque acteur l'élément «model:participant» correspondant qui sera rajouté à «model:collaboration» et un élément «model:process» qui sera rajouté à la racine du fichier. L'élément «model:participant» pointe vers l'élément «model:process». Un «élément «model:laneSet» est rajouté à l'élément «model:process».

La règle R2 consiste à chercher les modules utilisés par chaque acteur et à représenter chaque module par un lane dans le pool correspondant à l'acteur. Pour ce faire, la règle R2 crée un élément «model:lane» et le rajoute au «model:laneSet» du «model:process».

La règle R3.1 consiste à chercher, pour chaque acteur métier, les tâches qu'il a réalisé à partir des actions tracées et des entités sur lesquelles portent les actions. Cette règle est basée sur les correspondances entités/tâches présentées en partie dans Tableau 3 et Tableau 4 de la section 3.3. Les activités du niveau supérieur sont récupérées à partir d'OntoProcess (Figure 24, Figure 25, Figure 26 et Figure 27).

La règle R3.2 consiste à chercher les regroupements temporels des tâches. Elle crée l'élément «model :task » qui sera rajouté à l'élément « model :process » de l'acteur métier correspondant et un élément « model :flowNodeRef » qui fait référence à «model :task » et qui sera rajouté au « model :lane » du module dans lequel la tâche est réalisée.

La règle R3.3 consiste à trouver les regroupements par hiérarchie. Elle crée l'élément « model :subProcess » de l'activité du niveau supérieur qui regroupe les tâches correspondantes.

La règle R4 crée les branchements séquentiels ou parallèles entre les tâches et les activités en fonction des regroupements temporels et hiérarchiques. Elle crée les éléments « model:parallelGateWay » et « model :sequenceFlow » correspondants.

R1 – quel acteur ?	Chercher dans les traces les acteurs métiers (« author ») qui ont participé au projet. Pour chaque acteur métier, construire un « pool ».
R2 – quel module ?	Chercher dans les traces pour chaque acteur les modules utilisés pour réaliser ses activités. Pour chaque module utilisé par l'acteur métier, construire un « lane » à l'intérieur du « pool » correspondant à cet acteur métier.
R3.1 quelle tâche ?	<p>Pour chaque acteur métier, chercher, dans les traces, les actions informatiques qu'il a réalisées et les entités concernées. Faire correspondre à chaque action une tâche à l'aide de l'ontologie OntoProcess.</p> <p>Pour cette tâche, récupérer, à partir des actions correspondantes, les données de « date », « entité », « valeur »... Récupérer dans l'ontologie l'activité de niveau supérieur.</p> <p>Une tâche (activité élémentaire) correspond à l'élément « task » du BPMN.</p>
R3.2 quel regroupement temporel ?	<p>Regrouper, dans des tâches parallèles, les actions correspondantes qui se déroulent dans un intervalle de temps sans interruption par des actions correspondantes à d'autres tâches. Il s'agit d'une itération d'tâche.</p> <p>Pour chaque tâche, récupérer les données de « date », « entité », « valeur ».</p> <p>Pour chaque tâche, construire un élément BPMN « task » et le mettre dans le « lane » et le « pool » correspondants.</p>
R3.3 regroupement par hiérarchie d'activités	<ol style="list-style-type: none"> 1) Rechercher les tâches portant sur des entités différentes et étant composants de la même activité (activité dite supérieure). 2) Si elles se déroulent dans un intervalle de temps sans interruption par une autre tâche, elles peuvent être regroupées en l'activité supérieure. 3) Pour cette activité, récupérer à partir des tâches qui la composent, les données de « date », « entité », « valeur »... <p>Itérer les étapes 1 à 3 pour différents niveaux de composition des activités.</p> <p>Une activité complexe correspond à l'élément « sub-process » du BPMN.</p>
R4 Création des relations logico-temporelles	<p>Après la recherche de toutes les tâches et activités du processus opérationnel, à chaque niveau, les ranger par ordre chronologique afin de retrouver les relations logico-temporelles entre elles. Les relations temporelles que nous cherchons sont de type séquentiel et parallèle.</p> <p>Les branchements parallèles et séquentiels correspondent aux éléments « parallelGateway » et « sequenceFlow » du BPMN.</p>

Tableau 15. Principes d'élaboration des règles de transformation

5.4.4 Implémentation de la construction de processus opérationnel par le CPO

Dans cette section, nous présentons uniquement l'implémentation de l'échange de message avec l'agent CPO pour une demande de visualisation du PO et ensuite la fonction d'initialisation du fichier BPMN réalisée par l'agent.

L'agent CPO reçoit la demande de construction du processus opérationnel (via TDC) et s'occupe de la construction du processus opérationnel après avoir reçu une réponse de l'agent ARAO concernant la demande des traces annotées.

```
XMLMessage m = (XMLMessage) waitNextMessage();
if (m.toString().equals("ConstructOP"))
{
    String BPMNfileName = "c:\\SMADir\\" + myGroup + "\\OPsFolder\\Operational.bpmn";
    String XmlDocument = "c:\\SMADir\\" + myGroup + "\\tracesFolder\\export.xml";

    try {

        GenerateDetailedBPMN.writeBPMNFile(XmlDocument, BPMNfileName);

    } catch (Exception e) { e.printStackTrace();}
}
```

Figure 60. Implémentation de l'Agent CPO

L'Agent CPO appelle la méthode « writeBPMNFile » de « GenerateDetailedBPMN » et indique les emplacements du fichier XML de traces et du fichier BPMN à générer.

La méthode writeBPMNFile crée tous les éléments BPMN correspondants à partir du fichier de traces en suivant les règles définies pour l'obtention du processus opérationnel comme nous l'avons spécifié.

Au départ, la méthode « InstantiateBPMNFile » crée les éléments fondamentaux du fichier BPMN (Figure 61).

L'élément « model:definitions » est l'élément racine du fichier. InstantiateBPMNFile ajoute à « model:definitions » l'élément « model:collaboration » pour indiquer les rôles participant au projet (« model:participant ») et l'élément « bpmndi:BPMNDiagram » qui contiendra la description graphique du processus.

```

private static void InstantiateBPMNFile (Document doc)
{
    Element definitionsElement = new Element("definitions", modelNS);
    definitionsElement.setAttribute("exporter", "Yaoqiang BPMN Editor");
    definitionsElement.setAttribute("expressionLanguage", "http://www.w3.org/1999/XPath");
    definitionsElement.setAttribute("targetNamespace", "http://sourceforge.net/bpmn/definitions/_1402870799854");

    doc.setRootElement(definitionsElement);

    Element collaborationElement = new Element ("collaboration", modelNS);
    collaborationElement.setAttribute ("id","Collaboration");

    definitionsElement.addContent(collaborationElement);

    Element bpmnDiagramElement = new Element("BPMNDiagram", bpmndiNS);
    bpmnDiagramElement.setAttribute ("id", "Yaoqiang_Diagram");
    bpmnDiagramElement.setAttribute ("name", "New Diagram");

    Element bpmnPlaneElement = new Element ("BPMNPlane", bpmndiNS);
    bpmnPlaneElement.setAttribute ("bpmnElement", "Collaboration");
    bpmnPlaneElement.setAttribute ("id", "Yaoqiang-Collaboration");

    bpmnDiagramElement.addContent(bpmnPlaneElement);

    definitionsElement.addContent(bpmnDiagramElement);
}

```

Figure 61. Initialisation du fichier BPMN

5.5 Module de visualisation du processus opérationnel

Ce module pose plus de problèmes techniques que scientifiques. Pour résoudre ces problèmes au mieux, nous avons testé différents éditeurs de processus. Pour comparer les fonctionnalités de ces logiciels, nous avons retenu cinq critères :

- Possibilité d'import/export au format BPMN
- Facilité d'installation et d'interfaçage avec la suite TDC
- Affichage correct des éléments du processus à partir d'un export BPMN sans la description graphique (DI).
- Support de tous les éléments BPMN (en particulier des pools, lanes...), y compris le zoom sémantique (possibilité de représenter des sous-processus)
- Programme libre (open source) ou payant (propriétaire).

Au début 2013, notre choix s'est porté sur Yaoqiang, un éditeur de processus de la communauté logicielle sourceforge qui supporte l'import et l'export des fichiers BPMN, critère très important au niveau du système multi-agents qui va construire le fichier BPMN du processus opérationnel et visualiser le processus dans l'éditeur graphique. Yaoqiang ne

nécessite aucune installation et son lancement est très léger (en quelques secondes). Quand le MPPAS termine la construction du processus opérationnel dans un fichier BPMN, il lance l'éditeur graphique Yaoqiang en lui indiquant l'emplacement du fichier. Cette procédure est transparente à l'utilisateur qui ne verra que le résultat de la visualisation du processus opérationnel dans Yaoqiang.

Critère Outil	Possibilité et format d'import /export	Facilité d'installation	Affichage correct des éléments du processus à partir d'un export BPMN sans le DI	Support de tous les éléments BPMN	Libre/ Payant
JBPM	Supporte Import Fichier conforme BPMN 2.0	Installation dans Eclipse	Non	Non (pas les pools et les lanes)	Libre
Activiti	Supporte Import BPMN Export BPMN, BPM, JPG, GIF, etc.	Installation lourde (fonctionne en plugin dans Eclipse)	Non	Oui	Libre
Bonitasoft	Supporte un Import BPMN (sans description graphique DI)/Export BPMN, BPM, JPG, GIF, etc.	Lent démarrage (+ 2min), import BPMN impossible par ligne de commande (interfaçage non possible avec la suite TDC)	Oui (partiellement)	Oui (partiellement)	Source (test gratuit)
Signavio	Supporte Import/Export (mais pas dans la version d'essai)	Longue procédure d'installation	Non	Oui	Source (payant pour tester l'import/export)
Yaoqiang	Supporte Import BPMN (sans description graphique DI)/Export BPMN, BPM, JPG, GIF, etc.	Aucune installation demandée	Oui	Oui	Libre jusqu'à la fin 2013)
Modelio	Pas d'import/Export BPMN	Installation légère	Non	Oui	Libre
ARIS Express	Pas d'import/Export BPMN	Installation légère	Non	Oui	Source
EA Tool (Enterprise Architecture tool)	Pas d'import /export BPMN	Installation légère	Non	Oui	Source (Sparx System)

Tableau 16. Comparaison des outils de visualisation des processus supportant le BPMN

Les éléments de processus sont automatiquement arrangés par Yaoqiang sans la partie BPMNDiagram dans laquelle est définie l'apparence de chaque élément dans le diagramme.

Mais cela ne reste pas valable lorsque nous décidons de tracer un flux d'activités à travers les différents rôles tenus dans les modules TDC par l'acteur et représentés par les lanes (qui présentent les modules TDC/rôle acteur) de l'acteur métier. Cette présentation nécessite de

rajouter la partie BPMNDiagram au fichier pour que l'affichage soit correct, sinon les éléments seront disposés en vrac.

Nous avons mentionné un autre problème concernant l'affichage des activités multi-niveaux représentées à l'aide des sous-processus. Sans la partie BPMNDiagram, toutes les activités multi-niveaux seront affichées par défaut en mode déployé, ce qui ne nous convient pas puisque notre objectif est d'afficher dans un premier temps un processus avec les activités du plus haut niveau, sans afficher leurs détails. Une activité multi-niveaux n'est déployée que lorsqu'on veut avoir accès à ses détails. Le rajout de la description graphique au fichier BPMN ne suffit pas pour résoudre le problème des activités multi-niveaux puisque la fonctionnalité qui permet de caler automatiquement les activités lorsqu'on passe du mode expansion au mode réduction (beaucoup d'espace perdu entre l'activité réduite et les activités voisines) ou vice versa (l'activité déployée chevauche sur les activités voisines, et par conséquent une partie du diagramme n'est pas lisible) n'est pas encore développée dans l'éditeur.

Cependant, Yaoqiang reste le plus ergonomique des éditeurs que nous avons pu tester : il supporte tous les éléments BPMN, permet une manipulation très intuitive des éléments affichées, et offre un graphique esthétique et professionnel ainsi qu'une liste exhaustive des fonctionnalités pour la personnalisation des éléments.

Cet éditeur était libre jusqu'au début de l'année 2014, date à laquelle la communauté qui l'a développé a rendu son accès payant.

5.6 Conclusion

Dans ce chapitre, nous avons présenté les trois modules du système à base de traces pour la construction automatique du processus opérationnel.

Le module de collecte est dirigé par deux agents du MPPAS : l'agent ET qui structure les traces collectées à partir des bases de données de l'outil TDC suivant le modèle des traces, une partie de l'ontologie OntoProcess, et l'agent ARAO qui rajoute les annotations organisationnelles aux traces collectées.

Le module de transformation utilise des règles de transformation pour construire les différents éléments du processus opérationnel à partir des traces collectées en fonction des exigences déterminées sur la construction du processus opérationnel. Ce module génère un fichier BPMN.

Le module de visualisation utilise un éditeur graphique BPMN pour la visualisation du processus opérationnel.

Le module de transformation et de visualisation sont supportés par l'agent CPO du MPPAS.

Dans le chapitre suivant, nous présentons les interfaces de MPPAS que nous avons conçues pour l'intégration de ses fonctionnalités dans la suite TDC.

Chapitre 6. Spécification d'interfaces et validation sur études de cas

Le système multi-agents MPPAS a été conçu pour être intégré dans la suite TDC de l'éditeur logiciel TDC Software. Dans ce chapitre, nous proposons la spécification d'interfaces utilisateurs de MPPAS implantées dans les modules de la suite TDC puis nous présentons des expérimentations de la construction du processus opérationnel sur quatre cas test et trois études de cas issues de données industrielles concernant une étude d'analyse fonctionnelle, une étude AMDEC et une étude de conception préliminaire. Pour des raisons de confidentialité, ces études ne peuvent pas être détaillées dans ce mémoire.

6.1 Interfaces utilisateurs et modules logiciels

Nous présentons dans cette section quatre interfaces : l'interface de consultation du référentiel des démarches métier et de construction d'une démarche métier, l'interface d'enrichissement d'une démarche métier, l'interface de visualisation du processus opérationnel et l'interface d'assistance à la démarche métier.

6.1.1 Interface de construction d'une démarche métier et de consultation du référentiel de démarches métier

Les démarches métier pratiquées par l'entreprise sont sauvegardées dans un référentiel de telle sorte que l'expert méthodologique puisse les consulter à tout moment, quand il veut enrichir, définir, ou créer une démarche métier pour une nouvelle étude.

Nous avons proposé pour ce faire une interface intégrée dans les modules de la suite TDC (voir Figure 62) et gérée par l'agent AEM (assistant expert méthodologique).

Pour une demande de consultation, l'agent AEM utilise l'ontologie OntoProcess pour structurer les requêtes de l'expert méthodologique et questionner le référentiel. Il retourne une liste des démarches qu'il a repérées dans le référentiel.

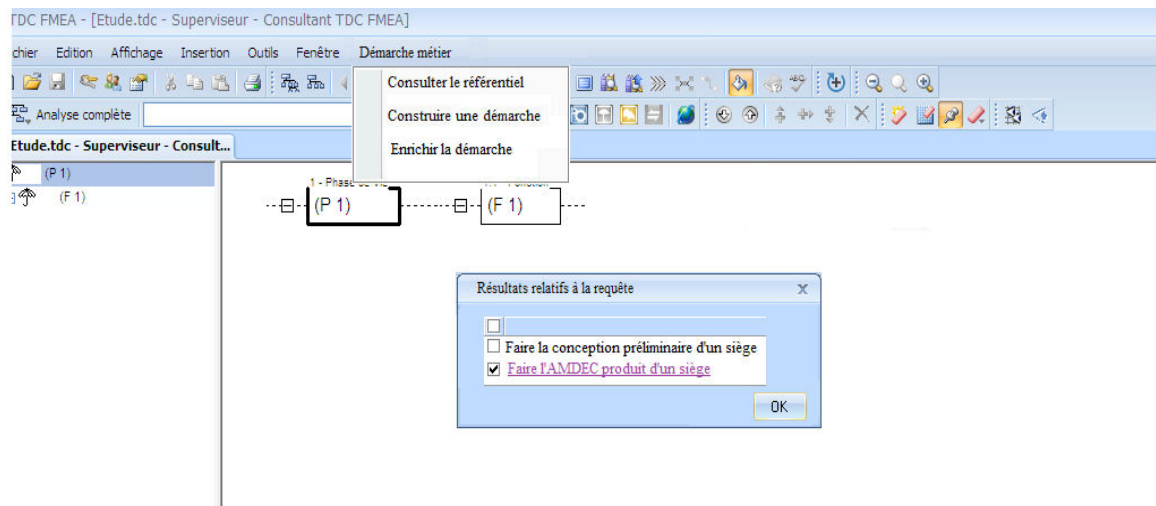


Figure 62. Interface « Consulter le référentiel/Construire la démarche métier »

Le choix de l’option «construire la démarche » dans le menu renvoie l’expert méthodologique sur l’éditeur BPMN dans lequel il va pouvoir modéliser la démarche métier. Ensuite, l’agent AEM se charge d’enregistrer la démarche créée dans le référentiel.

6.1.2 Interface enrichir les démarches métier

Au cours du projet, la démarche métier a pour rôle de guider et d’assister les acteurs métiers dans leurs activités afin d’améliorer l’efficacité du projet.

A la fin du projet, le retour d’expérience qu’offre le système sur le processus opérationnel permet de l’analyser d’une manière simple et de décider éventuellement d’enrichir la démarche métier correspondante afin de la réutiliser dans des projets similaires (voir Figure 63).

L’expert méthodologique va évaluer les résultats récoltés (expériences capitalisées) par le système : il compare la démarche métier au processus opérationnel et analyse les écarts.

Suite à cette évaluation, l’expert méthodologique peut décider de faire évoluer la démarche métier, il crée alors une nouvelle version de la même démarche métier. Après validation, toutes les connaissances capitalisées à partir du projet seront sauvegardées.

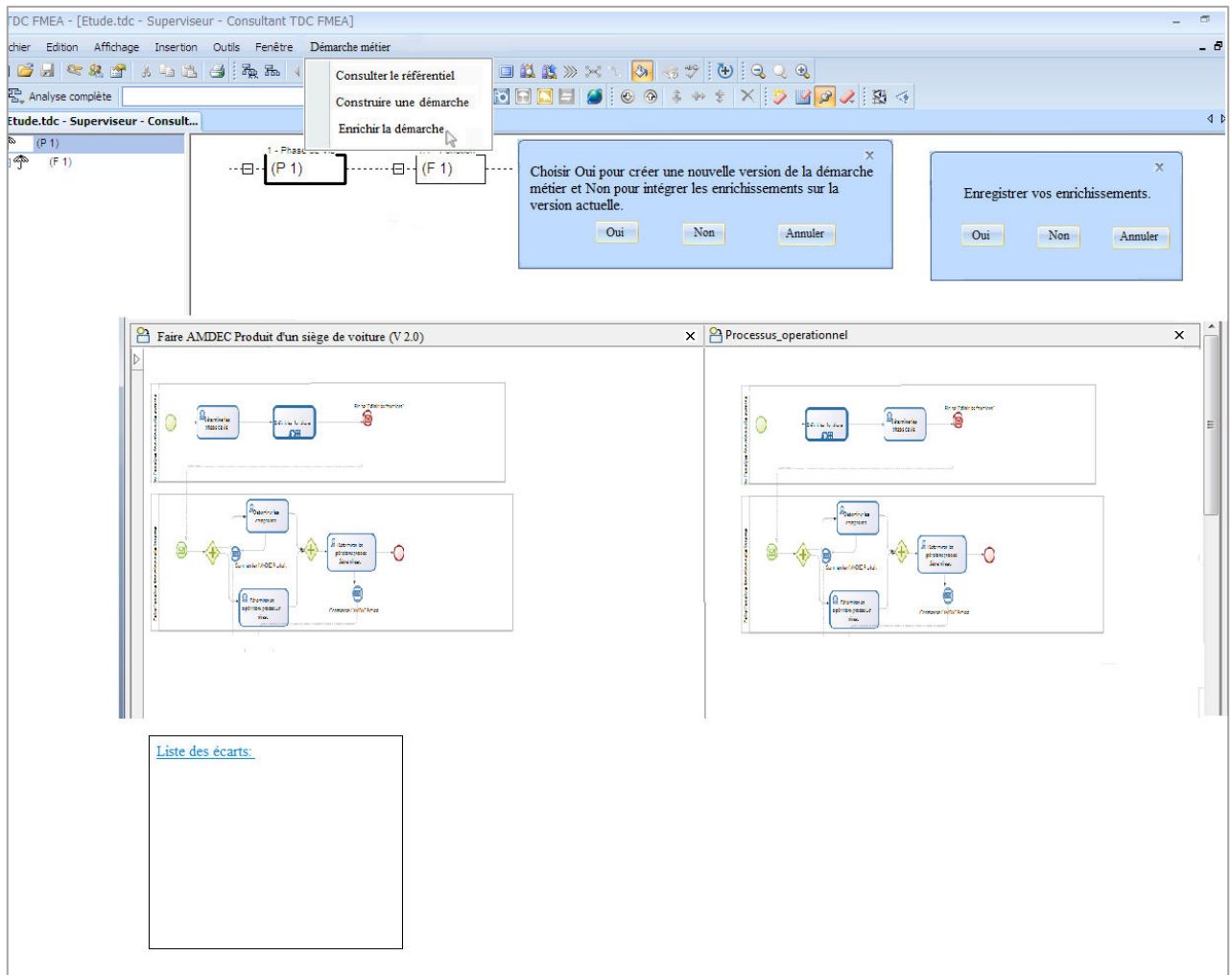


Figure 63. Interface « Enrichir une démarche métier »

6.1.3 Interface consulter le processus opérationnel

La consultation (ou visualisation) du processus opérationnel est un service basique que le système MPPAS assure. Le processus opérationnel construit constitue une donnée de base pour la surveillance du déroulement d'une étude et dans l'amélioration et l'enrichissement des démarches métiers. Suite à une demande de consultation, le processus opérationnel est construit par l'agent CPO (Constructeur du processus opérationnel) à partir des traces. L'interface de visualisation est représentée dans la Figure 63.

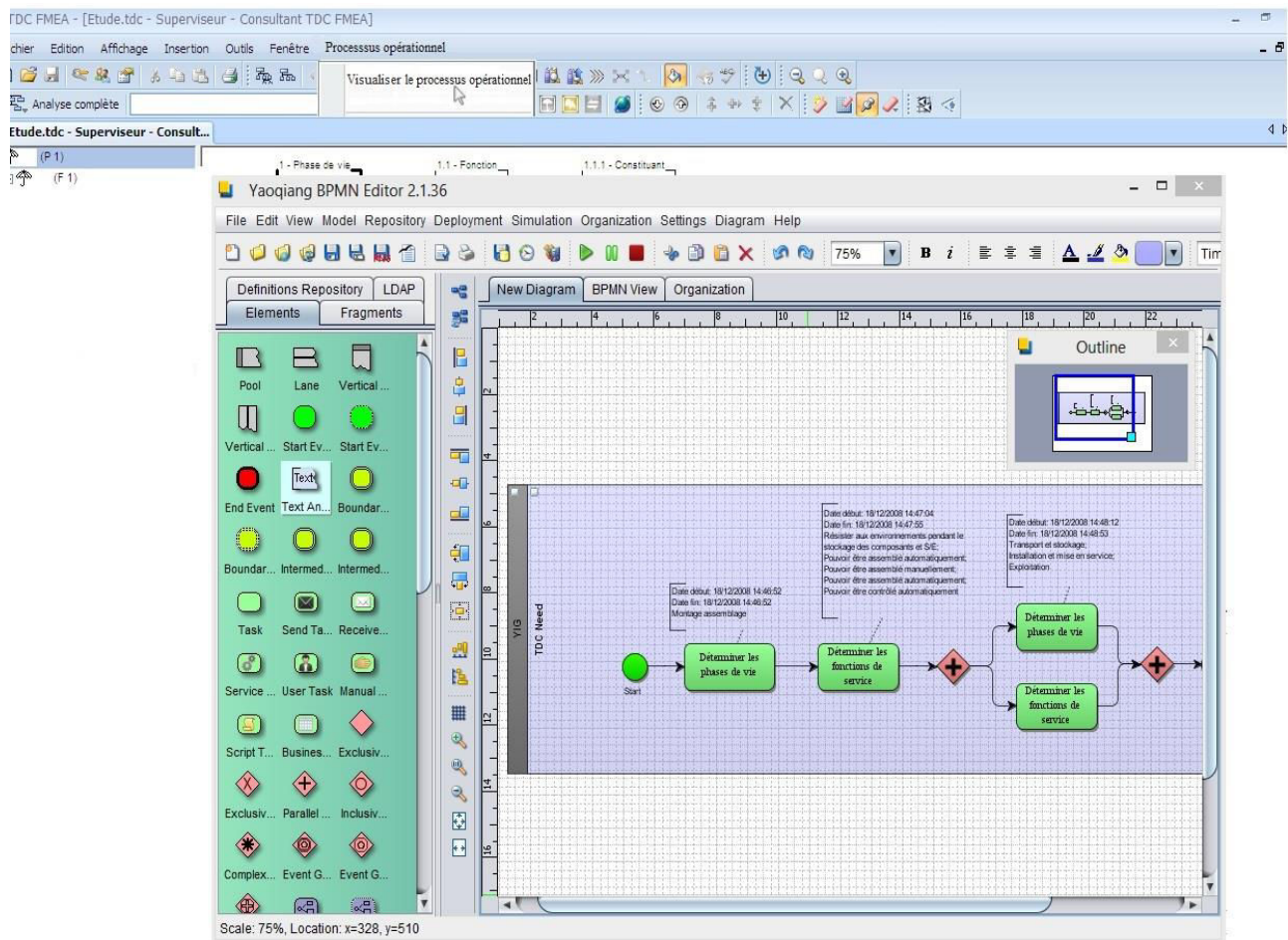


Figure 64. Interface « visualiser le processus opérationnel »

6.1.4 Interface demander de l'assistance dans le processus opérationnel

L'interface de demande d'assistance permet à l'acteur métier d'être guidé au cours d'une étude. Cette interface (Figure 65) est gérée par l'agent APM (Assistant Processus métier) qui est en mesure d'avertir l'acteur métier lorsqu'il réalise des activités ou enchaînement d'activités différents de ce qui est spécifié dans la démarche métier et de lui fournir une aide proactive permettant après la réalisation d'une tâche de proposer à l'acteur métier la réalisation d'une nouvelle tâche suivant la démarche métier.

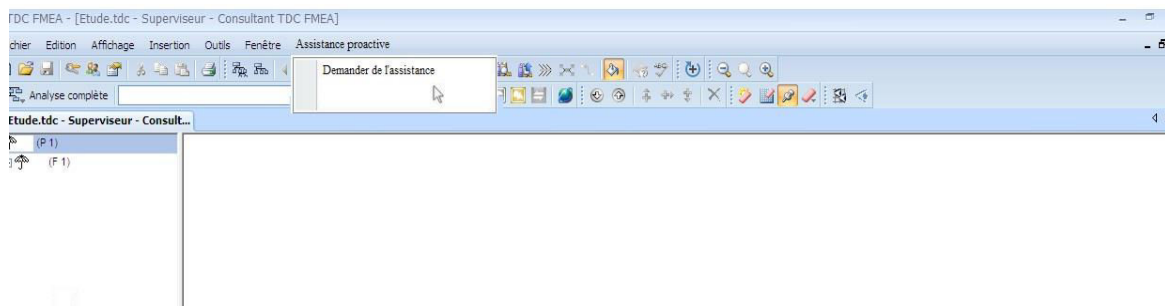


Figure 65. Interface « demander de l'assistance »

6.2 Expérimentations

Nous allons maintenant présenter des expérimentations issues de projets industriels dont les données ont été collectées et enregistrées par la suite TDC dans des fichiers de traces. Ces expérimentations que nous avons menées portent sur les quatre cas test spécifiés à la section 5.1 et sur trois études de cas industriels.

Ces études de cas vont nous permettre d'une part, de vérifier que les fonctionnalités spécifiées du SBT sont correctement implémentées et d'autre part, de mettre en évidence des points d'amélioration.

6.2.1 Résultats sur les quatre cas test

Dans la section 5.1, nous avons décrit quatre cas test relativement simples permettant de vérifier le bon fonctionnement de la construction d'un processus opérationnel.

Le cas test 1 représente un cas où chaque activité a été réalisée en une seule itération.

Dans le cas test 2, nous observons plusieurs occurrences d'une même activité.

Les cas test 1 et 2 illustrent les exigences sur la représentation des activités à un seul niveau de détail réalisées au cours d'une étude :

- Le processus opérationnel reconstruit est une collaboration entre les acteurs métiers ayant participé à l'étude.
- On a choisi de reconstruire séparément les activités de chaque acteur métier. Les activités sont placées dans des lanes représentant les modules TDC dans lesquels les activités sont produites. En effet, à partir du module TDC, on identifie le rôle de l'acteur métier.
- Les enchaînements logico-temporels entre les activités d'un acteur métier sont construits à travers les différents modules TDC.

Les cas test 3 et 4 fournissent des exemples de représentation sur des activités à plusieurs niveaux de détails. Nous avons vérifié que le système MPPAS génère un fichier BPMN correct mais l'affichage des sous-processus dans l'éditeur Yaoqiang pose problème, comme nous l'avons signalé précédemment (voir discussion section 5.5).

Nous présentons maintenant des expérimentations réalisées sur trois études de cas différentes, issus de projets industriels réels.

6.2.2 Etude de cas 1

Cette étude de cas est réalisée sur une étude d'analyse fonctionnelle. Nous illustrons, à partir des données issues de cette étude, toutes les étapes : collecte des traces brutes, génération des traces structurées et ensuite construction du processus opérationnel.

Collecte des traces

Les données relatives aux entités de conception manipulées au cours d'une étude sont collectées dans différents tableaux qui ont été générés par la suite TDC. Chaque tableau a une structure qui lui est propre : par exemple, des données sur les fonctions de service sont collectées dans un tableau (Figure 66), des données sur les phases dans un deuxième tableau, des données sur les versions des entités et l'historique des actions dans un troisième (Figure 67).

Tous les objets Access		curlID	Id_TREE	FCT_Str	FCT_ID	FCT_ID_Lien
Rechercher...		571	1	Résister aux environnements pendant le stockage des composants et S/E	571	-1
Arborescence_0_Item		607	2	Pouvoir être assemblé automatiquement	607	-1
Arborescence_Dump		677	3	Pouvoir être assemblé manuellement	677	-1
Arborescence_FCT_Col		766	4	Pouvoir être contrôlé manuellement	766	-1
Arborescence_FCT_E_Col		836	5	Pouvoir être contrôlé automatiquement	836	-1
Arborescence_FCT_E_C_Item		1317	6	Pouvoir être emballé	1317	-1
Arborescence_FCT_E_F_Col		1368	7	Avoir des composants facilement transportables	1368	-1
Arborescence_FCT_E_F_Item		1438	8	Pouvoir être déplacé	1438	-1
Arborescence_FCT_Item		1458	9	Résister aux chocs	1458	-1
Arborescence_FLUX_BOUC_Col		1478	10	Résister aux vibrations du transport	1478	-1
Arborescence_FLUX_BOUC_Item		1492	11	Résister aux ambiances extérieures	1492	-1
Arborescence_INTERACT_Col		1538	12	Supporter le gerbage	1538	-1
Arborescence_INTERACT_Item		1583	13	Ne pas blesser l'installateur	1583	-1
Arborescence_OPERATION_Col		1623	14	Permettre de se raccorder à la charge	1623	-1
Arborescence_OPERATION_Item		1637	15	Permettre de se raccorder à la source électrique	1637	-1
Arborescence_PHASE_Col		1746	16	Permettre de se fixer au support	1746	-1
		1815	17	Permettre de fixer les accessoires	1815	-1
		1876	18	Garantir la sécurité de l'exploitant	1876	-1
		1929	19	Permettre à l'exploitant de connaître la position de l'interrupteur	1929	-1

Figure 66. Extrait du tableau des fonctions de service

CurID	Value	Comment	AppliName	Author	Date	Type	Code	
570	Montage assemblage	Copier-Coller	TDC Need	YIG	18/12/2008 14:46:52	11	PHASE	Arborescence_PHASE_Item
571	Résister aux environnements pendant le stockage de	Copier-Coller	TDC Need	YIG	18/12/2008 14:47:04	10	FCT	Arborescence_FCT_Item
607	Pouvoir être assembler automatiquement	Copier-Coller	TDC Need	YIG	18/12/2008 14:47:33	10	FCT	Arborescence_FCT_Item
677	Pouvoir être assembler manuellement	Copier-Coller	TDC Need	YIG	18/12/2008 14:47:55	10	FCT	Arborescence_FCT_Item
766	Pouvoir être contrôler manuellement	Copier-Coller	TDC Need	YIG	18/12/2008 14:47:55	10	FCT	Arborescence_FCT_Item
836	Pouvoir être contrôler automatiquement	Copier-Coller	TDC Need	YIG	18/12/2008 14:47:55	10	FCT	Arborescence_FCT_Item
1317	Pouvoir être emballé	Copier-Coller	TDC Need	YIG	18/12/2008 14:47:55	10	FCT	Arborescence_FCT_Item
1368	Faciliter la manutention des composants et S/E avant	Copier-Coller	TDC Need	YIG	18/12/2008 14:47:55	10	FCT	Arborescence_FCT_Item
1437	Transport et stockage	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:12	11	PHASE	Arborescence_PHASE_Item
1438	Pouvoir être déplacé	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:12	10	FCT	Arborescence_FCT_Item
1458	Résister aux chocs	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:12	10	FCT	Arborescence_FCT_Item
1478	Résister aux vibrations du transport	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:12	10	FCT	Arborescence_FCT_Item
1492	Résister aux ambiances extérieures	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:12	10	FCT	Arborescence_FCT_Item
1538	Supporter le gerbage	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:12	10	FCT	Arborescence_FCT_Item
1582	Installation et mise en service	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:30	11	PHASE	Arborescence_PHASE_Item
1583	Ne pas blesser l'installateur	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:30	10	FCT	Arborescence_FCT_Item
1623	Pouvoir se raccorder à la charge	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:30	10	FCT	Arborescence_FCT_Item
1637	Permettre de se raccorder à la source électrique	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:30	10	FCT	Arborescence_FCT_Item
1746	Permettre de se fixer au support	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:30	10	FCT	Arborescence_FCT_Item
1815	Permettre de fixer les accessoires	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:30	10	FCT	Arborescence_FCT_Item
1875	Exploitation	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:53	11	PHASE	Arborescence_PHASE_Item
1876	Garantir la sécurité de l'exploitant	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:53	10	FCT	Arborescence_FCT_Item
1929	Permettre à l'exploitant de connaître la position de l'	Copier-Coller	TDC Need	YIG	18/12/2008 14:48:53	10	FCT	Arborescence_FCT_Item

Figure 67. Extrait du tableau de l'historique des actions

Les données collectées dans les tableaux sont difficiles à analyser. En effet, ces données brutes concernent tous les détails informatiques d'une étude, et ne sont pas conçues dans le but de construire le processus opérationnel. Cela s'avère difficile à réaliser à partir des données brutes car :

- toutes les données enregistrées dans les tableaux ne sont pas utiles ;
- nous ignorons la structure des différents tableaux ;
- il peut y avoir une perte de temps dans le parcours des tableaux.

Il est donc nécessaire d'extraire des traces structurées à partir de ces données brutes. Cela est réalisé par l'agent ET (extracteur des traces) selon le modèle des traces que nous avons défini dans la section 5.3.1.

La Figure 68 présente une extraction des traces structurées à partir des données brutes enregistrées dans les tableaux des Figure 66 et Figure 67.

```

<entity level="1" type="phase" id="570">
  >> Phase 570
  <action type="MAJ" date="2008-12-18T14:46:52" version="0-2" author="YIG" tool="TDC Need" value="Montage assemblage"/> >> Action réalisée sur la phase 570
  <entity level="1" type="function" id="571">
    >> Fonction de service 571 de la Phase 570
    <action type="MAJ" date="2009-01-12T09:34:45" version="0-3" author="YIG" tool="TDC Need" value="Résister aux environnements pendant le stockage des composants et S/E"/>
    <action type="Copier-Coller J:\users\yig\p86\TDC\p86_SIROC.tdc" date="2008-12-18T14:47:04" version="0-1" author="YIG" tool="TDC Need" value="Résister aux environnements pendant le stockage des composants"/>
  </entity>
  <entity level="1" type="function" id="607">
    >> Fonction de service 607 de la Phase 570
    <action type="MAJ" date="2009-01-12T09:34:55" version="0-4" author="YIG" tool="TDC Need" value="Pouvoir être assemblé automatiquement"/> >> Action réalisée sur la fonction 607
    <action type="MAJ" date="2009-01-08T16:14:35" version="0-2" author="YIG" tool="TDC Need" value="Pouvoir être assemblé automatiquement"/>
    <action type="Copier-Coller J:\users\yig\p86\TDC\p86_SIROC.tdc" date="2008-12-18T14:47:33" date_n="1229608012" version="0-1" author="YIG" tool="TDC Need" value="Pouvoir être assembler automatiquement"/>
  </entity>
  <entity level="1" type="function" id="677">
    <action type="MAJ" date="2009-01-12T09:35:01" version="0-4" author="YIG" tool="TDC Need" value="Pouvoir être assemblé manuellement"/>
    <action type="MAJ" date="2009-01-08T16:14:40" version="0-2" author="YIG" tool="TDC Need" value="Pouvoir être assemblé manuellement"/>
    <action type="Copier-Coller J:\users\yig\p86\TDC\p86_SIROC.tdc" date="2008-12-18T14:47:55" date_n="1229608012" version="0-1" author="YIG" tool="TDC Need" value="Pouvoir être assembler manuellement"/>
  </entity>
</entity>

```

Figure 68. Extrait des traces XML structurées

Construction du processus opérationnel

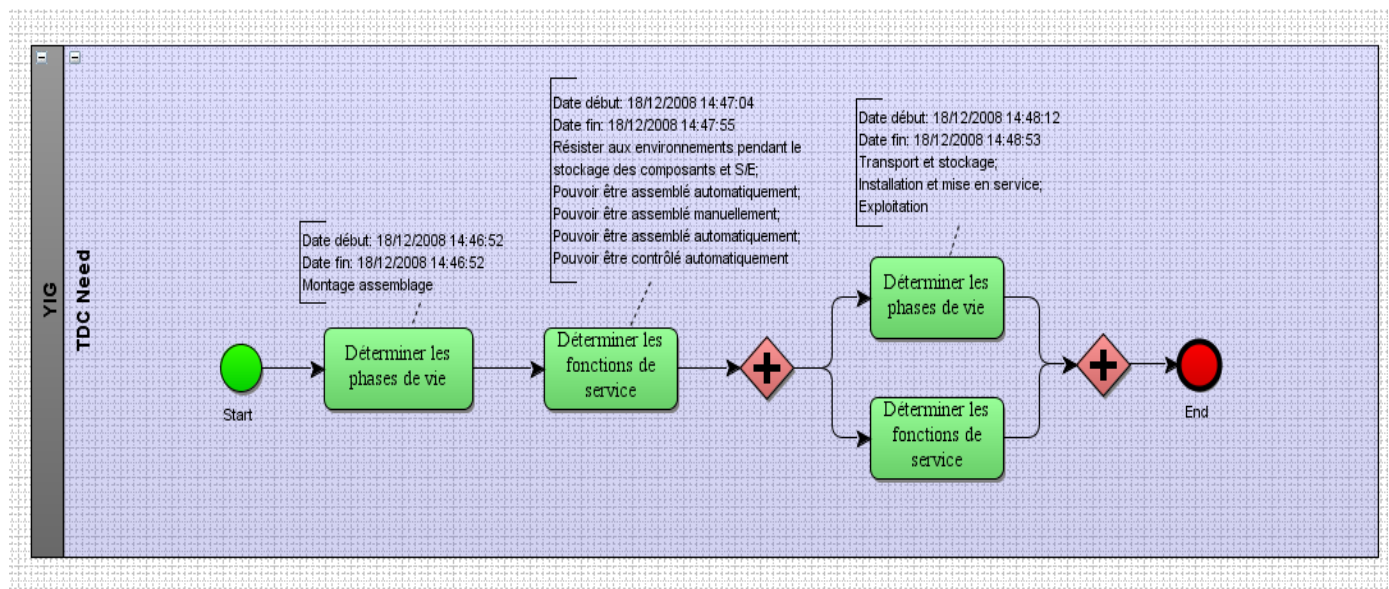


Figure 69. Visualisation du processus opérationnel construit pour l'étude de cas 1

Dans cette étude de cas, nous avons vérifié les exigences suivantes :

- Bonne construction du pool (pour l'acteur) et du lane (pour l'outil utilisé).
- Bonne construction des flux de séquence et des branchements parallèles. Lorsque l'acteur métier copie plusieurs entités provenant d'une étude anté-

rière, les traces sont enregistrées à la même date. Dans ce cas, nous vérifions que les activités correspondantes sont alors représentées en parallèle.

- Bonne construction des itérations des activités à partir des regroupements des actions.

Amélioration possible :

Il est possible de donner plus de précision sur les activités, et cela à partir des entités créées au cours de l'activité. Par exemple, la deuxième activité dans le processus opérationnel de la Figure 69 « déterminer les fonctions de service » pourrait être remplacée par « déterminer les fonctions de service mécaniques »

6.2.3 Etude de cas 2

La Figure 70 correspond à un projet industriel concernant une étude AMDEC produit, impliquant deux acteurs métier différents et utilisant TDC FMEA et TDC Need.

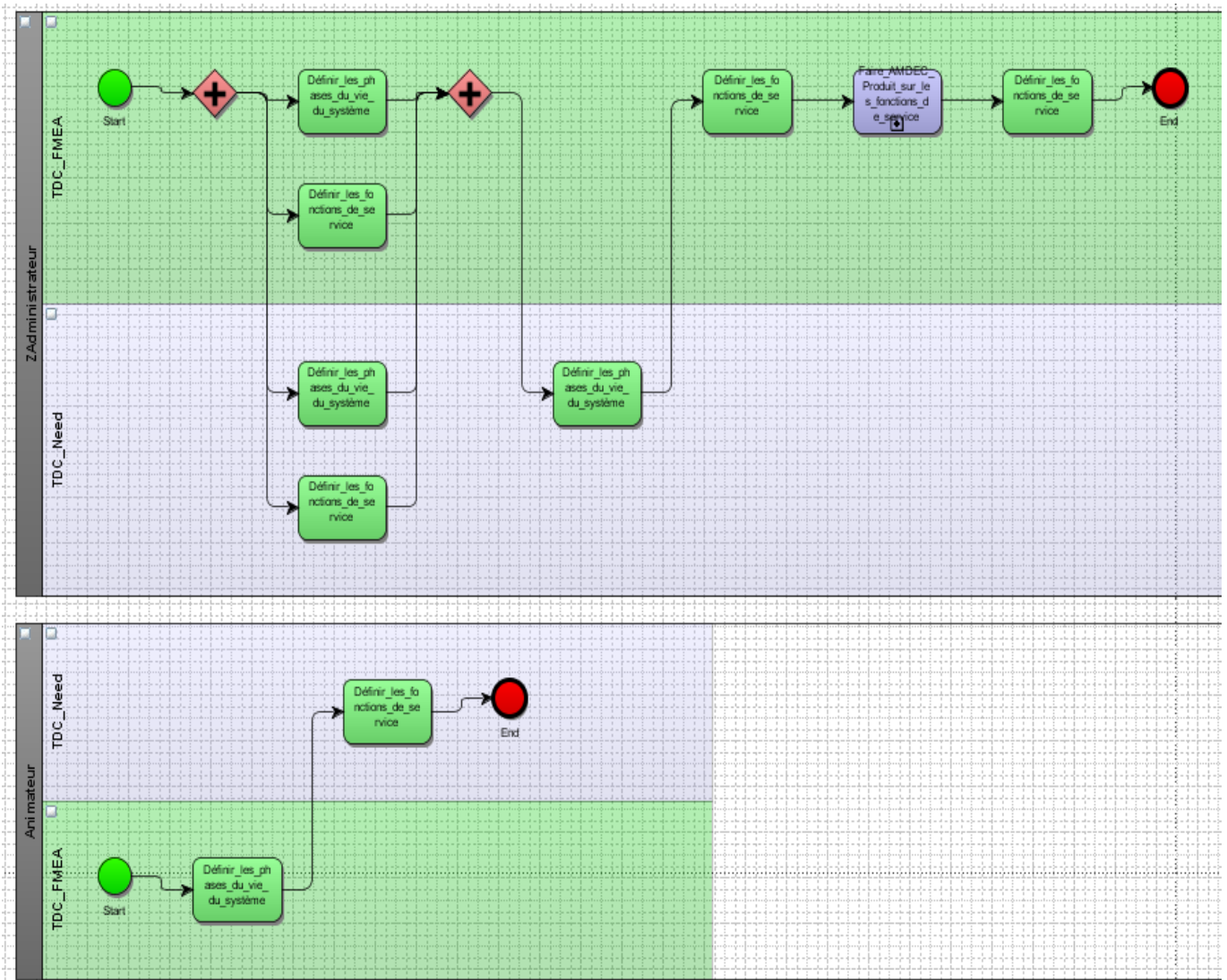


Figure 70. Visualisation du processus opérationnel construit pour l'étude de cas 2

La Figure 71 présente la visualisation détaillée du sous-processus Faire l'AMDEC produit sur les fonctions de service, qui correspond à l'avant-dernière activité dans le processus visualisée dans la Figure 70.

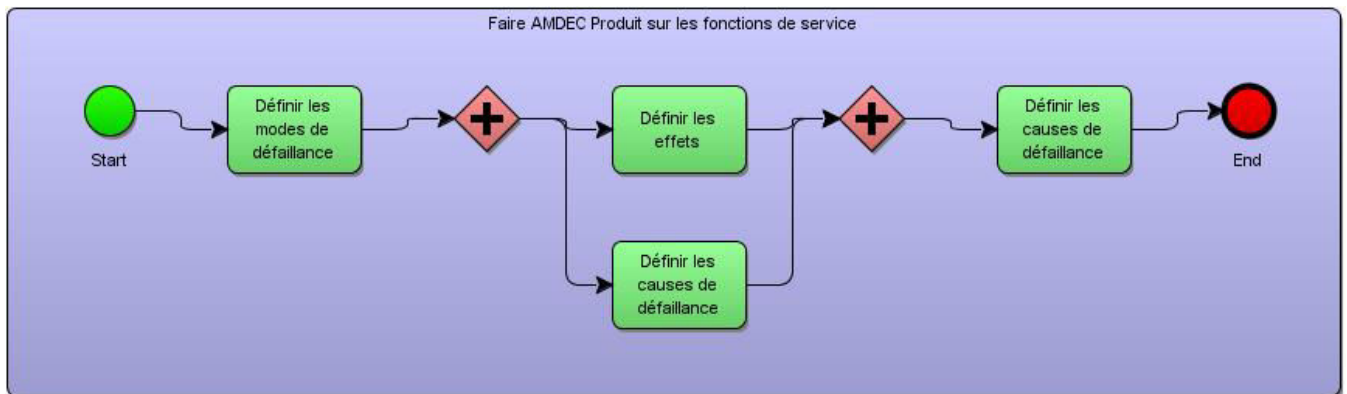


Figure 71. Visualisation détaillée du sous-processus

La fenêtre « Outline » (Figure 72) donne une vision macroscopique du processus et permet de savoir où se situe l'activité à consulter par rapport au processus complet.

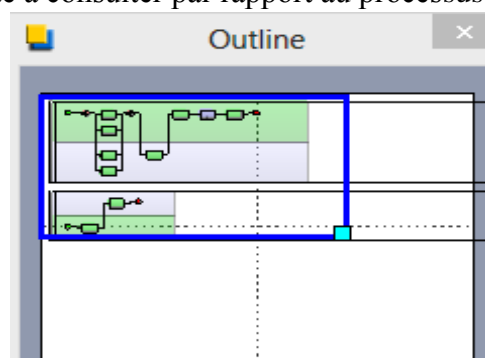


Figure 72. Fenêtre d'affichage global

Par rapport aux exigences définies sur l'affichage du processus opérationnel, nous avons validé :

Au niveau syntaxique :

- Bonne identification des pools/lanes. Une couleur a été attribuée au lane en fonction du module qu'il représente.
- Construction correcte du « StartEvent » et du « EndEvent » annonçant le commencement des activités de l'acteur métier. Pour chaque pool, le StartEvent est placé dans le lane correspondant au module dans lequel l'acteur métier débute son activité et le EndEvent est placé dans le lane correspondant au module où l'acteur métier finit son activité.
- Bon regroupement des actions informatiques en activités dans les lanes en fonction des rôles de l'acteur métier.
- Flux de processus à travers les divers rôles de l'acteur.
- Bonne construction des sous-processus.

Amélioration possible :

Actuellement, les messages de communication entre les collaborateurs (acteurs métier) ne sont pas définis dans OntoProcess, et par conséquent le processus opérationnel construit ne contient pas ce type d'élément BPMN. Un enrichissement d'OntoProcess serait possible pour supporter la représentation des échanges (par messages) entre les collaborateurs.

Cependant, l'expert méthodologique peut enrichir manuellement la représentation d'une démarche métier en utilisant les différents éléments de modélisation BPMN.

6.2.4 Etude de cas 3

La Figure 73 correspond à un processus opérationnel construit avec une collaboration entre trois acteurs pouvant manipuler plusieurs modules de la suite TDC, tenant ainsi différents rôles dans le processus de conception.

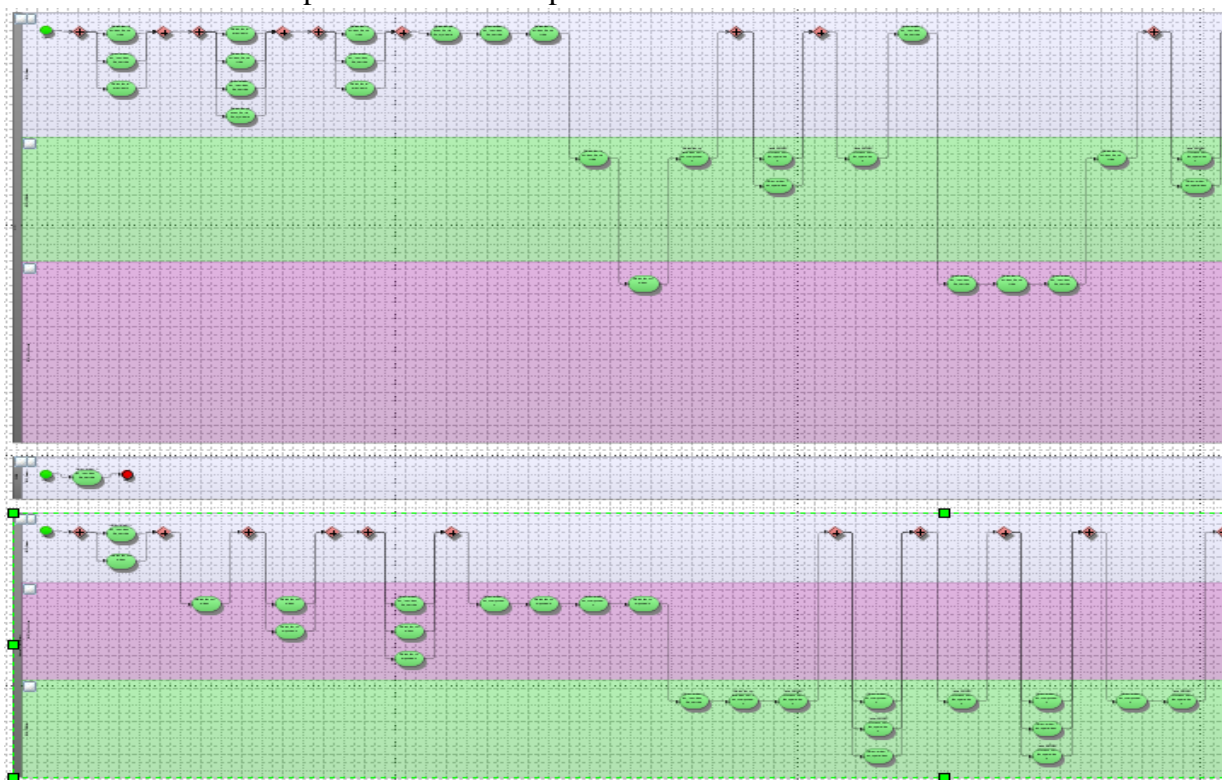


Figure 73. Visualisation du processus opérationnel construit pour l'étude de cas 3

Ce test a permis de vérifier les fonctionnalités suivantes :

- Bonne représentation d'une séquence d'activités d'un acteur métier (représenté par un pool) tenant différents rôles lors de l'utilisation des modules de la suite TDC (représentés par des lanes).
- Processus interprétable par les experts.
- Choix possible d'une plage temporelle d'affichage, par saisie des dates de début et de fin de visualisation.

Il met en évidence des points d'amélioration :

- Possibilité de visualiser les informations concernant les entités modifiées lors d'une activité. Cette fonctionnalité a été spécifiée (voir Figure 22) mais n'a pas été implémentée.
- Positionnement des gateways à améliorer. Actuellement, ces éléments BPMN sont redondants lors de l'enchaînement de deux branches parallèles successives. Ils sont positionnés dans le premier lane lors d'une séquence se déroulant sur plusieurs lanes.

Nous avons vérifié, par des expérimentations sur des cas test et sur des études de cas, que les règles de transformation que nous avons définies dans la section 5.4.3 et que nous avons implémentée dans l'agent Constructeur de Processus Opérationnel (CPO) ont permis, en grande partie, de construire les processus opérationnels selon les exigences identifiées en section 5.1.

6.3 Conclusion

Dans ce chapitre, nous avons présenté quatre interfaces utilisateurs : l'interface de consultation du référentiel des démarches métier et de construction d'une démarche métier, l'interface d'enrichissement d'une démarche métier, l'interface de visualisation du processus opérationnel et l'interface d'assistance à la démarche métier.

Ensuite, nous avons présenté des expérimentations issues de projets industriels. Ces expérimentations portent sur les quatre cas test spécifiés à la section 5.1 et sur trois études de cas industriels. Les études de cas nous ont permis de vérifier la satisfaction des exigences sur la construction du processus opérationnel et de mettre en évidence des points d'amélioration.

Chapitre 7. Conclusions et perspectives

De nombreuses activités de conception sont aujourd'hui instrumentées avec des logiciels permettant de modéliser différentes vues du produit et de ses systèmes associés. Ces données sont gérées directement par les logiciels supportant la conception ou par des systèmes d'information de type PLM ou GED.

En complément aux modèles du produit, il est important pour les entreprises de conserver la mémoire des processus opérationnels mis en œuvre. Ces processus collaboratifs reflètent une compétence collective de l'organisation, pour mener à bien un projet nouveau.

Il est bien connu qu'une nouvelle étude, menée au sein d'une entreprise, est souvent semblable à une étude précédente et par conséquent, peut être structurée selon un processus de référence commun au type d'étude correspondant. Dans notre travail, nous l'avons appelé démarche métier. Il s'agit d'une bonne pratique de réalisation d'un type d'études.

La difficulté majeure réside dans la formalisation de cette démarche métier. Les approches traditionnelles de capitalisation des connaissances s'appuyant sur des verbalisations d'experts ont montré leur limite. Souvent réalisées en dehors de l'activité réelle, les experts omettent des détails qui peuvent être d'importance.

De plus en plus fréquemment en conception, les acteurs métiers utilisent un outil numérique qui est le support à leurs activités collaboratives et qui permet d'enregistrer le résultat de leurs actions élémentaires. Notre thèse repose sur l'idée qu'il est possible de construire le processus opérationnel collaboratif mis en œuvre lors des activités des acteurs métier à partir des traces enregistrées d'utilisation de l'outil numérique.

Nous faisons les hypothèses que des connaissances cruciales sont incorporées dans le processus collaboratif réellement mis en œuvre et que la nature et l'enchaînement des activités de conception ont une relative invariance pour un certain type de projets ou d'études.

Les types de projet ou d'étude peuvent dépendre des pratiques de l'entreprise (formalisées sous forme de processus institutionnalisés ou de référence) ou du lien du type d'étude avec une méthode de conception normalisée comme l'Analyse Fonctionnelle ou l'AMDEC. Ces hypothèses sont particulièrement vraies dans le contexte industriel de notre travail de recherche.

Les contributions présentées dans ce mémoire sont les suivantes :

- Un état de l'art sur les méthodes de gestion des connaissances et en particulier, sur les systèmes à base de traces et les systèmes multi-agents, au potentiel important dans le cadre de notre thèse,
- Un double cycle de capitalisation pour des activités instrumentées,
- Une approche globale de gestion des démarches métier,

- Une ontologie OntoProcess modélisant des aspects organisationnels génériques (séparant clairement des concepts liés aux traces et d'autres liés aux démarches métier) et des extensions métiers spécifiques à l'outil utilisé,
- Un système multi-agents supportant l'approche globale de gestion des démarches métiers et s'appuyant sur l'ontologie OntoProcess,
- Un système à base de traces permettant de construire un processus opérationnel à partir des traces enregistrées lors d'une étude.

Le système à base de traces a été implémenté et testé sur deux jeux de données qui proviennent de traces d'études industrielles réalisées par des clients de TDC et qui nous ont été fournis.

L'ensemble de ces contributions est actuellement à des niveaux différents de conception, implémentation, et test. Le tableau suivant résume le niveau de maturité obtenu à la fin de la thèse et montre le chemin qu'il reste à parcourir pour obtenir un système industrialisé au sein de la suite logicielle TDC Software.

Module	Conçu	Implémenté dans un prototype	Intégré dans ou interfacé avec TDC	Testé
Système à Base de Traces :				
- Extracteur de traces	X	X	X	X
- Analyseur et rédacteur d'annotations organisationnelles	X	X		X
- Constructeur de processus opérationnel	X	X	X (partiellement)	X
Comparateur entre processus opérationnels et démarche métier	X			
Assistant expert méthodologique	X			
Assistant processus métier (aide proactive)	X			

Figure 74. Avancement du développement du Système Multi-agents

Des problèmes scientifiques ont été traités sur les règles de transformation et sur la spécification du système multi-agents. Des difficultés techniques ont été rencontrées sur la visualisation des processus.

Partant du travail qui reste à réaliser et d'approfondissements possibles, nous pouvons dégager plusieurs perspectives de recherche.

Un travail de réflexion important des experts métiers est nécessaire pour identifier les types d'études et leur attacher une démarche métier. Cette réflexion est basée sur la recherche d'invariants entre les objectifs et situations des études ainsi que sur les processus considérés comme performants. Nous avons placé cette réflexion hors du périmètre de ce travail de recherche mais elle pourra faire l'objet d'un travail ultérieur. Deux approches complémentaires peuvent être imaginées. L'une consiste à questionner les experts, à s'appuyer sur les types de produits conçus, sur les méthodes de conception utilisées, afin

d'identifier les types d'études. La seconde suppose que l'entreprise dispose déjà d'une base de données de processus opérationnels. Les techniques de data mining actuelles, ainsi que des algorithmes de mesure de similarité entre processus, peuvent être utilisées pour rechercher des invariants entre processus et proposer des classifications de processus. Une interprétation de cette classification peut alors confirmer les résultats de la première approche ou faire ressortir de nouveaux types d'étude.

Dans l'approche globale que nous avons proposée, l'enrichissement d'une démarche métier est basé soit sur la génération des écarts entre le processus opérationnel construit par l'agent CPDOM et la démarche métier, soit sur une comparaison manuelle que l'expert méthodologique réalise. Une méthode de calcul de mesure de similarité a été proposée dans un master recherche (López Vásquez, 2013) adossé à ce travail de thèse et testée sur un cas d'étude issu de la littérature. Cependant, elle n'a pas pu être implémentée dans le cadre du SMA et testée sur un cas industriel. En utilisant l'ontologie OntoProcess comme lexique pour les activités, le calcul de la mesure de similarité sera facilité dans nos travaux. Nous devons surtout approfondir la mesure de similarité structurelle qui joue un rôle important. Un autre aspect rarement pris en compte dans la littérature concerne le niveau de généralité pour la comparaison des processus. L'ontologie OntoProcess facilitera le calcul sur ce type de similarité.

Il peut être envisagé de valider que les relations déterminées par l'expert dans la démarche métier sont bien respectées lors de la réalisation du projet. Cela consiste à déterminer et à effectuer un ensemble de tests de comparaison entre les activités identifiées dans la démarche métier et celles qui sont déduites à partir des actions.

Une autre perspective de recherche concerne la modélisation d'une expérience de conception (une étude) comme un cas. Il serait intéressant de comparer l'intérêt de notre approche avec une approche de type raisonnement à partir de cas. Les problématiques posées seraient alors l'identification d'études antérieures correspondant à une nouvelle étude (problème cible), la réutilisation ou l'adaptation d'une démarche métier (solution), la maintenance de la base de cas (des études)... D'autre part, des travaux pourraient concerner le passage d'un ensemble d'expériences (processus opérationnel construit) à la découverte de leçons nouvelles, c'est-à-dire des règles d'enchaînement d'activités découvertes à partir de l'analyse des processus construits. Ces leçons pourraient alors être utilisées pour proposer à l'expert méthodologique des enrichissements de la démarche métier.

D'autre part, on pourrait déterminer un ensemble de relations qu'on souhaite construire et effectuer pour chaque relation les tests nécessaires pour proposer ou non à l'expert d'ajouter la relation dans la démarche métier.

D'autres voies d'amélioration concernent de nouvelles fonctionnalités qui peuvent être assurées par le système proposé.

Quand la démarche métier est déterminée, il peut être intéressant de l'utiliser pour *générer un planning de projet* adapté aux objectifs du projet. Il suffit de définir les jalons et allouer les activités aux acteurs métier ayant les compétences nécessaires pour les accomplir, tout en respectant les relations existant entre les activités conformément à la démarche métier choisie. A cet égard, nous avons spécifié une interface (Figure 75) qui nous semble adaptée

pour cette fonctionnalité. Le système va donc retourner automatiquement les éléments de la démarche à planifier pour le projet.

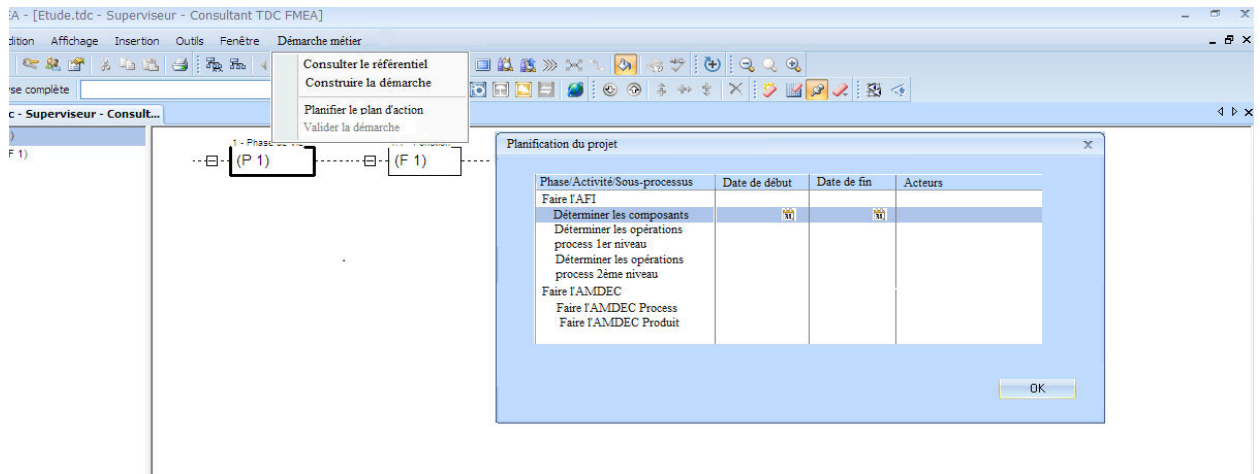


Figure 75. Interface « Générer un planning de projet »

Pour que le chef de projet soit capable d'évaluer et *suivre l'avancement du processus* qu'il a planifié, il faut lui fournir des informations sur chaque activité. Nous pouvons spécifier une interface adaptée (Figure 76). Pour cela, la construction du processus opérationnel au cours du projet pourra permettre d'évaluer l'avancement de l'étude et d'estimer un indicateur d'avancement. Pour chaque activité du projet, nous évaluons si elle a commencé en avance, à temps, en retard, ou si elle n'a pas encore commencé. De même, le système évalue l'état de fin des activités devant normalement être clôturées. Pour les activités en cours, le système évalue aussi leur état d'avancement (par exemple : un quart/à moitié/ trois quart effectué).

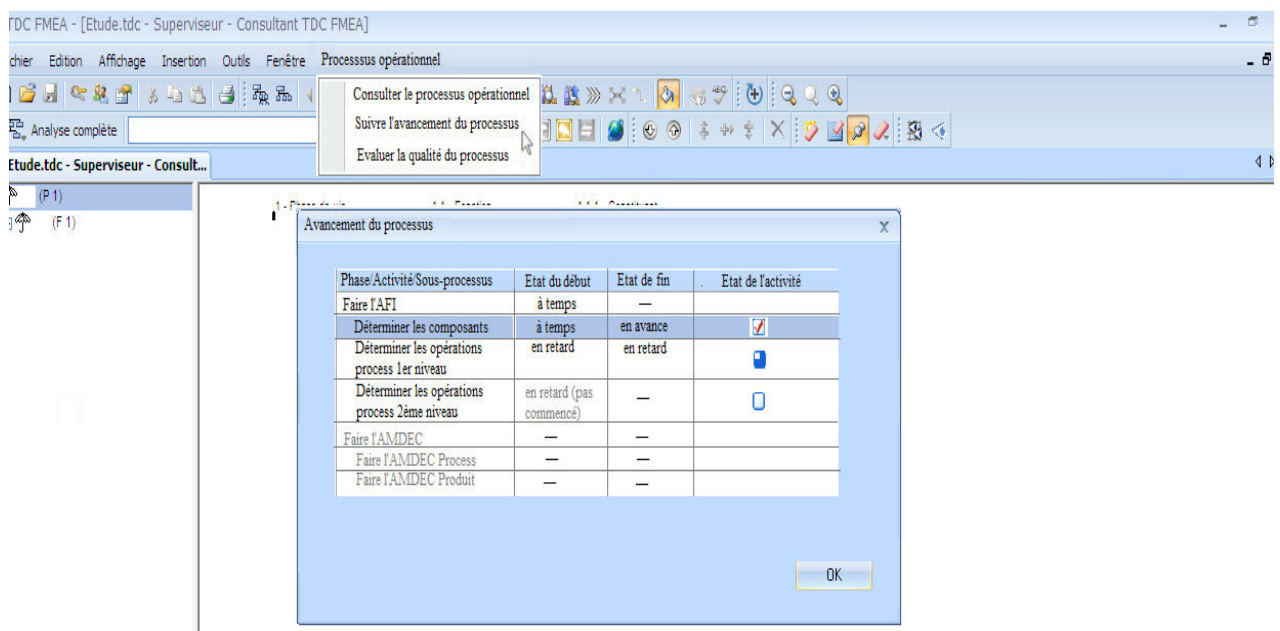


Figure 76. Interface « suivre l'avancement du processus »

Chapitre 8. Références bibliographiques

- Abecker, A., A. B., & Van Elst, E. (2003). Agent technology for distributed organizational memories. *Proceedings of the 5th International Conference On Enterprise Information Systems, 2*, 3–10.
- Arana, J., Hassas, S., & Prié, Y. (2004). MAZETTE: Multi agent MUsETTE for sharing and reusing ontologies. In *Proceedings of the OTM 2004 Workshop on Ontologies, Semantics and E-Learning (WOSE'2004)* (Vol. 3292). Larnaca, Cyprus, Greece: LNCS , Springer Verlag.
- Axelrod, R. (1997). The Complexity of Cooperation – Agent-Based Models of Competition and Collaboration. In P. U. Press (Ed.), . Princeton, New Jersey.
- Bachimont, B. (2000). Engagement sémantique et engagement ontologique: conception et réalisation d'ontologies en ingénierie des connaissances. In J. Charlet, M. Zacklad, K. G., & B. D. (Eds.), *Ingénierie des connaissances. Evolutions récentes et nouveaux défis*. Paris: Eyrolles.
- Barthès, J. P., Dieng, R., & Kassel, G. (1999). Dossier Mémoire d'Entreprise. *Bulletin de l'AFIA*, 36.
- Barthès, J. P., & Tacla, C. (2002). Agent-supported portals and knowledge management in complex R&D projects. *Computer in Industry -Special Issue: CSCW in Design*, 48(1), 3 –16.
- Bekhti, S. (2003). *DypKM: un processus dynamique de définition et de réutilisation de mémoires de projets*. Thèse de Doctorat, Université de Technologie de Troyes. Retrieved from <http://hal.archives-ouvertes.fr/tel-00511045/>
- Bekhti, S., & Matta, N. (2003). A formal approach to model and reuse the project memory. In *Proceedings of I-KNOW'3, Industry meets Science*. Graz, Austria.
- Belkadi, F. (2006). *Contribution au pilotage des compétences dans les activités de conception: de la modélisation des situations à la caractérisation des compétences*. Thèse de Doctorat, Université de Franche-Comté, N° 1176. Retrieved from <http://tel.archives-ouvertes.fr/tel-00217404/>
- Belkadi, F., Bonjour, E., & Dulmet, M. (2006). Modelling Framework of a Traceability System to Improve Knowledge Sharing and Collaborative Design. *Lecture Notes in Computer Science, LNCS, Springer-Verlag, Berlin*, 3865, 355–364.
- Bergmann, R., Kolodner, J., & Plaza, E. (2005). Representation in case-based reasoning. *The Knowledge Engineering Review*, 00, 1–4. doi:10.1017/S0000000000000000
- Bersini, H., & Wellesz, I. (2002). L'orienté objet. *Eyrolles, ISBN : 2-212-11108-8*.

- Bigand, M., Mekhilef, M., & Bourey J.P. (2006). Traçabilité des justifications de décisions en conception de produit. In B. Yannou & B. E. (Eds.), *Evaluation et décision dans les processus de conception* (Hermès, co.).
- Blanco, E. (1998). *L'émergence du produit dans la conception distribuée, vers de nouveaux modes de rationalisation dans la conception de systèmes mécaniques*. Thèse de doctorat, INP de Grenoble.
- Bock, C., Zha, X., Suh, H., & Lee, J.-H. (2010). Ontological product modeling for collaborative design. *Advanced Engineering Informatics*, 24(4), 510–524. doi:10.1016/j.aei.2010.06.011
- Bonifacio, M., Bouquet, P., & Traverso, P. (2002). Enabling distributed knowledge management. managerial and technological implications. *Novatica and Informatik/Informatique*, 3(1), 23–29.
- Bonjour, E. (2008). Du retour d'expérience au développement des compétences. In J. Renaud, E. Bonjour, B. Chebel-Morello, B. Fuchs, & N. Matta (Eds.), *Retour et capitalisation d'expériences* (AFNOR., pp. 155–198).
- Bonjour, E., Dulmet, M., & Lhote, F. (2002). An internal modeling of competency, based on a systemic approach, with socio-technical systems management in view. In *IEEE International Conference on Systems, Man and Cybernetics, SMC'2002, Hammamet, Tunisie*.
- Boujut, J.-F., & Laureillard, P. (2002). A co-operation framework for product–process integration in engineering design. *Design Studies*, 23(6), 497–513. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0142694X01000448>
- Bousbia, N. (2011). *Analyse des traces de navigation des apprenants dans un environnement de formation dans une perspective de détection automatique des styles d'apprentissage*. Université Paris 6.
- Bousbia, N., Rebaï, I., Labat, J.-M., & Balla, A. (2010). Learners' navigation behavior identification based on trace analysis. *User Modeling and User-Adapted Interaction*, 20(5), 455–494. doi:10.1007/s11257-010-9081-5
- Brazier, F., Moshkina, L., & Wijngaards, N. (2001). Knowledge level model of an individual designer as an agent in collaborative distributed design. *Artificial Intelligence in Engineering*, 15, 137–152.
- Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., & Mylopoulos, J. (2001). A Knowledge Level Software Engineering Methodology for Agent Oriented Programming. In *Proceedings of the 5th International Conference on Autonomous Agents*. (pp. 648–655). Montreal (Canada): ACM Press.
- Budzik, J., Hammond, K., & Birnbaum, L. (n.d.). Information access in context. *Knowledge Based Systems*, 14(1-2), 37–53.
- Chabin, M.-A. (2004). Document trace et document source. La technologie numérique change-t-elle la notion de document? *Information-Interaction-Intelligence*, 4(1), 141–158. Retrieved from http://archivesic.ccsd.cnrs.fr/docs/00/06/24/02/PDF/sic_00001020.pdf

- Champin, P. A., Prié, Y., & Mille, A. (2003). MUNETTE: Modelling USEs and Tasks for Tracing Experience. In *Proc. From structured cases to unstructured problem solving episodes* (pp. 279–286). Trondheim (NO), NTNU: WS 5 of ICCBR'03.
- Charlet, J. (2003). *L'ingénierie des connaissances, développements, résultats et perspectives pour la gestion des connaissances médicales*.
- Chella, A., Cossentino, M., Sabatucci, L., & Seidita, V. (2004). From PASSI to Agile PASSI: Tailoring a Design Process to Meet New Needs. In *IEEE/WIC/ACM International Joint Conference on Intelligent Agent Technology (IAT-04)*. Beijing China.
- Chinosi, M., & Trombetta, A. (2012). BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 34(1), 124–134. doi:10.1016/j.csi.2011.06.002
- Choquet, C., & Iksal, S. (2007). Modeling Tracks for the Model Driven Re-engineering of a TEL System. *Journal of Interactive Learning Research*, 18(2), 161–184.
- Cirilo, E., Nunes, I., Kulesza, U., & Lucena, C. (2012). Automating the product derivation process of multi-agent systems product lines. *The Journal of Systems and Softwar*, 85, 258–276.
- Clauzel, D., Sehaba, K., & Prié, Y. (2011). Enhancing synchronous collaboration by using interactive visualisation of modelled traces. *Simulation Modelling Practice and Theory*, 19(1), 84–97. doi:10.1016/j.simpat.2010.06.021
- Corkill, D. Lander, S. (1998). Diversity in Agent Organizations. *Object Magazine*, 8(4), 41–47.
- Cossentino, M. (2005). Agent-oriented methodologies. In P. Giorgini (Ed.), *Agent-Oriented Methodologies* (Idea Group.). Hershey, PAS, USA.
- Cram, D., Jouvin, D., & Mille, A. (2007). Visualisation interactive de traces et réflexivité: application à l'EIAH collaboratif synchrone eMédiathèque. *Revue Sticef*, 14, 1–22. Retrieved from http://sticef.univ-lemans.fr/num/vol2007/06-cram/sticef_2007_cram_06p.pdf
- Cutkosky, M. R., & Englemore, R. S. (1993). PACT: An experiment in integrating concurrent engineering systems. *IEEE Computer*, 26(1), 28–37.
- D.1.1. (2007). Business Process Ontology Framework. *Super Deliverable*.
- Darses, F., Détienne, F., & Visser, W. (2001). Assister la conception: perspectives pour la psychologie cognitive ergonomique. In *EPIQUE 2001, Journées d'étude en Psychologie ergonomique, Nantes, France, 29-30 Octobre 2001*. Retrieved from <http://sfpsy.org/spe-grape/Actes-epique-2001-article-darses.pdf>
- Dignum, V. (2004). *A Model for Organizational Interaction: based on Agents, founded in Logic*. PhD dissertation, Universiteit Utrecht.
- Dignum, V., Dignum, F., & Meyer, J. J. (2004). An Agent-Mediated Approach to the Support of Knowledge Sharing in Organizations. *Knowledge Engineering Review, Cambridge University Press*, 19(2), 147–174.

- Dongen, B., van, Dijkman, R., & Mendling, J. (2008). Measuring Similarity between Business Process Models. In Z. Bellahsene & M. Léonard (Eds.), *Advanced Information Systems Engineering* (Springer B., pp. 450–464).
- Ebrahimipour, V., Rezaie, K., & Shokravi, S. (2010). An ontology approach to support FMEA studies. *Expert Systems with Applications*, 37(1), 671–677. doi:10.1016/j.eswa.2009.06.033
- Ehrig, M., Koschmider, A., & Oberweis, A. (2007). Measuring similarity between semantic business process models. In *Proceedings of the Fourth Asia-Pacific Conference on Conceptual Modelling - Vol. 67, (Darlinghurst, Australia)* (pp. 71–80). Retrieved from <http://dl.acm.org/citation.cfm?id=1274465>
- El kadiri, S., Pernelle, P., Delattre, M., & Bouras, A. (2008). Collaborative process control: Observation of tracks generated by PLM system. In *APMS 2008 - Innovations in Networks*. Espoo, Finlande.
- Eliseo, M. A., Oliveira, J. M. P. De, & Pellegrino, S. R. M. (2011). Domain Ontology with Temporal Descriptions for Architectural Buildings as a Support for Learning History of Architecture. *IEEE Multi-Disciplinary Engineering Education Magazine*, 6(2), 24–31.
- Enembreck, F., & Barthès, J.-P. (2003). IMPROVING COMPUTER SUPPORTED COOPERATIVE DESIGN WITH PERSONAL ASSISTANT AGENTS. *Journal of Integrated Design and Process Science, IOS Press*, 7(2), 3–19.
- Ermine, J., Chaillot, M., Bigeon, P., Charreton, B., & Malavieille, D. (1996). MKSM : Méthode pour la gestion des connaissances. *Ingénierie Des Systèmes D'information, AFCET-Hermès*, 4(4), 541–575.
- Esteva, M., Rodriguez-Aguilar, J. A., Sierra, C., Garcia, P., & Arcos, J. L. (2001). On the formal specification of electronic institutions. *Agent Mediated Electronic Commerce*, 126–147.
- Eynard, B. (1999, January 1). *Modélisation du produit et des activités de conception : contribution à la conduite et à la traçabilité du processus d'ingénierie*. Bordeaux 1. Retrieved from <http://www.theses.fr/1999BOR10571>
- Ferber, J. (1999). *Multi-Agent System: An Introduction to Distributed Artificial Intelligence*. Harlow: Addison Wesley Longman, ISBN: 0-201 (Vol. 1). Harlow, England: Harlow: Addison Wesley Longman.
- Fernandez, M., Gomez-Perez, A., & Juristo, N. (1997). From Ontological Arts Towards Ontological Engineering. *Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering*, 33-40.
- Filipowska, A., Hepp, M., Kaczmarek, M., & Markovic, I. (2009). Organisational Ontology Framework for Semantic Business Process Management. *LNBP*, 21, 1–12.
- Gandon, F. (2002). *Distributed Artificial Intelligence and Knowledge Management: ontologies and multi-agent systems for a corporate semantic web*. Thèse de doctorat, Université de Nice - Sophia Antipolis.
- Gandon, F., Berthelot, L., & Dieng-Kuntz, R. (2002). A Multi-Agent Platform for a Corporate Semantic Web. In C. Castelfranchi & W. L. Johnson (Eds.), *AAMAS 2002, 6th International Conference on*

Autonomous Agents, 5th International Conference on Multi-Agents Systems, 9th International Workshop on Agent Theories Architectures and Languages (pp. 1025–1032). Bologna.

- Gandon, F., & Dieng-Kuntz, R. (2005). Ontologie pour un système multi-agents dédié à une mémoire d'entreprise. In R. Teulier, J. Charlet, & P. Tchounikine (Eds.), *Ingénierie des connaissances* (pp. 141–163). Paris: L'Harmattan.
- Georgeon, O. (2008). *Analyse de traces d'activité pour la modélisation cognitive*. Thèse de doctorat, Université de Lyon 2, psychologie.
- Georgeon, O., Henning, M., & Bellet, T. (2007). Creating cognitive models from activity analysis: A knowledge engineering approach to car driver modeling. In *International Conference on Cognitive Modeling* (pp. 43–48). Ann Arbor, Michigan: Taylor & Francis.
- Georgeon, O., Mille, A., & Bellet, T. (2012). Supporting activity modelling from activity traces. *Expert Systems*, 29(3), 261–275. Retrieved from <http://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2011.00584.x/full>
- Giunchiglia, F., Mylopoulos, J., & Perini, A. (2002). The Tropos Software Development Methodology: Processes, Models and Diagrams. In *Proceedings of Agent-Oriented Software Engineering (AOSE-2002)* (pp. 63–74). Bologna, Italy.
- Gomez-Perez, A., Fernandez, M., & De Vicente, A. (1996). Towards a Method to Conceptualize Domain Ontologies Workshop on ontological engineering. *ECAI'96*, 41–51.
- Greco, D., & Brown, D. (1996). Learning by single function agents during spring design. In J. Gero & Sudweeks (Eds.), *Artificial Intelligence in design 96 (AID 96)* (pp. 409–428). Dordrecht: Kluwer Academic.
- Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 2(5), 199–220.
- Grundstein, M. (2002). *Le Management des Connaissances dans l'Entreprise. Problématique, Axe de progrès, Orientations*.
- Grundstein, M., & Barthès, J.-P. (1996). An Industrial View of the Process of Capitalizing Knowledge. In J. F. Schreinemakers (Ed.), *Knowledge Management: Organization, Competence and Methodology, Proc. of ISMICK'96* (Vol. 1, pp. 258–264). Rotterdam, the Netherlands: Verlag, Advances in Knowledge Management.
- Gu, T., Chen, S., Tao, X., & Lu, J. (2010). An unsupervised approach to activity recognition and segmentation based on object-use fingerprints. *Data & Knowledge Engineering*, 69(6), 533–544. doi:10.1016/j.datak.2010.01.004
- Guizzardi, R. (2006). *Agent-Oriented Constructivist Knowledge Management*. Thèse de doctorat, University of Twente, the Netherlands.
- Guizzardi, R., Aroyo, L., & Wagner, G. (2004). Agent-oriented Knowledge Management in Learning Environments: A Peer-to-Peer Helpdesk Case Study. *Agent-Mediated Knowledge Management, LNCS, Springer, 2926*, 57–72. doi:10.1007/978-3-540-24612-1_4

- Guizzardi, R., & Perini, A. (2005). Analyzing Requirements of Knowledge Management Systems with the Support of Agent Organizations. In *Journal of the Brazilian Computer Society (JCBS). Special Issue on Agents Organizations*, 11(1), 51–62.
- Hadj-Hamou, K., Aldanondo, M., Lamothe, J., & Caillaud, E. (2002). Configurateur et Modeleur CAO : deux outils complémentaires pour l'aide à la conception de produits à forte diversité. *International Journal of Design and Innovation Research*, 3(1-2), 81–95.
- Hadzic, M., Pompit, W., Dillon, T., & Chang, E. (2009). Ontology-Based Multi-Agent Systems. *Studies in Computational Intelligence, Springer*, 219, 1–273.
- Harani, Y. (1997). *Une approche multi modèles pour la capitalisation des connaissances dans le domaine de la conception*. Thèse de Doctorat en génie industriel, INPG, Grenoble.
- Hatchuel, A. (2002). Weil a, B.(2002). La théorie CK: Fondements et usages d'une théorie unifiée de la conception. In *Colloque «Sciences de la conception»* (pp. 1–24). Retrieved from <http://www.scinnovation.net/wp-content/uploads/soft/100921/6-100921131206.pdf>
- Heer, T., & Wörzberger, R. (2011). Support for modeling and monitoring of engineering design processes. *Computers & Chemical Engineering*, 35(4), 709–723. doi:10.1016/j.compchemeng.2010.12.004
- Helmy, T., Amamiya, S., Mine, T., & Amamiya, M. (2002). An Agent-Oriented Personalized Web Searching System. In *AOIS '02 International Bi-Conference Workshop on Agent-Oriented Information System (AOIS-2002 at AAMAS*02)*. Italy.
- Hilbert, D. M., & Redmiles, D. F. (2000). Extracting usability information from user interface events. *ACM Computing Surveys*, 32(4), 384–421. doi:10.1145/371578.371593
- Iglesias, C. A., Garijo, M., & Gonzalez, J. C. (1999). 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 02. *Intelligent Agents V: Agents Theories, Architectures and Languages, LNCS, Springer Verlag*, 1555, 317–330. doi:10.1007/3-540-49057-4_21
- Jennings, N. R., & Wooldridge, M. J. (1998). *Agent Technology: Foundation, Applications, and Markets*. Springer, New York (pp. 1–325). New York, USA: Springer.
- Kamsu Foguem, B., Coudert, T., Béler, C., & Geneste, L. (2008). Knowledge formalization in experience feedback processes: An ontology-based approach. *Computers in Industry*, 59(7), 694–710. doi:10.1016/j.compind.2007.12.014
- Karray, M. H. (2012). *Contribution à la spécification et à l'élaboration d'une plateforme de maintenance orientée connaissances*. Thèse de doctorat, Université de Franche-Comté.
- Karray, M.-H., Chebel-Morello, B., & Zerhouni, N. (2014). PETRA: Process Evolution using a TRAcE-based system on a maintenance platform. *Knowledge-Based Systems*. doi:10.1016/j.knosys.2014.03.010
- Kassel, G. (2002). OntoSpec : une méthode de spécification semi-informelle d'ontologies. In *Actes des Journées Francophones d'Ingénierie des Connaissances: IC'2002* (pp. 75–87). Rouen.

- Kwan, M., & Balasubramanian, P. (2003). KnowledgeScope: managing knowledge in context. *Decision Support Systems*, 35(4), 467–486. doi:10.1016/S0167-9236(02)00126-4
- Lahoud, I., Monticolo, D., & Gomes, S. (2010). OCEAN: A Semantic Web Service to Extract knowledge in EGroupware. In *KARE*. Malaysia.
- Lai, M.-C., Huang, H.-C., & Wang, W.-K. (2011). Designing a knowledge-based system for benchmarking: A DEA approach. *Knowledge-Based Systems*, 24.
- Lancieri, L. (2009). De l'analyse de traces à l'exploitation des phénomènes d'intelligence collective. In M. Jean-Charles & M. Alain (Eds.), *Analyse de traces et personnalisation des environnements informatiques pour l'apprentissage humain* (Traité IC2.). Eds Hermes.
- Le Boterf, G. (2000). *Construire les compétences individuelles et collectives* (éd. d'orga.). Paris.
- Lee, M. R., & Chen, T. T. (2012a). Revealing research themes and trends in knowledge management: From 1995 to 2010. *Knowledge-Based Systems*, 28, 47–58. doi:10.1016/j.knosys.2011.11.016
- Lee, M. R., & Chen, T. T. (2012b). Revealing research themes and trends in knowledge management: From 1995 to 2010. *Knowledge-Based Systems*, 28, 47–58. doi:10.1016/j.knosys.2011.11.016
- Lees, B., Branki, C., & Aird, I. (2001). A framework for distributed agent-based engineering design support. *Automation in Construction*, 10(5), 631–637.
- Lhote, F., Chazelet, P., & Dulmet, M. (1999). The extension of principles of cybernetics, towards engineering and manufacturing. *Annual Reviews in Control*, 23, 139–148.
- Liao, S. (2003). Knowledge management technologies and applications—literature review from 1995 to 2002. *Expert Systems with Applications*, 25(2), 155–164. doi:10.1016/S0957-4174(03)00043-5
- Lieberman, H. (1995). Letizia: An agent that assists web browsing. In C. S. Mellish (Ed.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)* (pp. 924–929). Montreal, Quebec, Canada: Morgan Kaufmann.
- Lieberman, H., Van Dyke, N., & Vivacqua, A. (1999). Let's browse: A collaborative web browsing agent. In *Proceedings of the 1999 International Conference on Intelligent User Interfaces (IUI-99)* (Mark Maybu., pp. 65–70). New York, USA: ACM Press.
- Liu, Y., & Harding, J. (2008). Editorial for the special issue of knowledge discovery and management in engineering design and manufacturing. *Journal of Intelligent Manufacturing*, 20(5), 499–500. doi:10.1007/s10845-008-0209-y
- López Vásquez, B. B. (2013). *Determining similarity between business process models*. Université de Lorraine, Master thesis, master Design Global.
- Maes, P. (1994). Agents that Reduce Work and Information Overload. *Communications of the ACM*.
- Malone, T. W., & Crowston, K. (1994). The interdisciplinary study of coordination. *ACM Computing Surveys*, 26(1), 87–119.

- Malone, T. W., & Smith, S. S. (1988). Modeling the performance of organizational structures. *Operations Research*, 36(3), 421–436.
- Malvache, P., & Prieur, P. (1993). Mastering Corporate Experience with the REX Method, Management of Industrial and Corporate Memory. In *International Symposium on the Management of Industrial and Corporate Knowledge (ISMICK'93)* (pp. 33–41). Compiègne.
- Mantelet, F. (2006). *Prise en Compte de la Perception Emotionnelle du Consommateur dans le Processus de Conception de Produits*. Thèse de doctorat, Paris: Laboratoire Conception de Produits et Innovation, ENSAM.
- Matta, N., Corby, O., & Ribiere, M. (1999). *Méthodes de capitalisation de mémoire de projet*. Retrieved from <http://hal.inria.fr/inria-00072839/>
- Matta, N., Ribiere, M., & Corby, O. (1999). *Définition d'un modèle de mémoire de projet*. Retrieved from <http://hal.archives-ouvertes.fr/inria-00072945/>
- McAlinden, L., Florida-James, B., Chao, K.-M., Norman, P., Hills, W., & Smith, P. (1998). Information and knowledge sharing for distributed design agents. In J. Gero & F. Sudweeks (Eds.), *Artificial intelligence in design 98 (AID98)* (pp. 537–556). Netherlands: Dordrecht:Kluwer Academic Publishers.
- Menand, S. (2002). *Modélisation pour la réutilisation du processus de conception multi acteurs de produits industriels: application à la conception fonctionnelle des systèmes*. INPG, Grenoble. Retrieved from <http://pagesperso.g-scop.grenoble-inp.fr/~tollenam/thesis/Memoire-Menand.pdf>
- Micaëlli, J. P., & Forest, J. (2003). *Artificialisme, Introduction à une théorie de la conception*. collection Epistémologie, INSA de Lyon.
- Mikos, W. L., Ferreira, J. C. E., Botura, P. E. a., & Freitas, L. S. (2011). A system for distributed sharing and reuse of design and manufacturing knowledge in the PFMEA domain using a description logics-based ontology. *Journal of Manufacturing Systems*, 30(3), 133–143. doi:10.1016/j.jmsy.2011.06.001
- Mille, A. (2006). From case-based reasoning to traces-based reasoning. *Annual Reviews in Control*, 30(2), 223–232. doi:10.1016/j.arcontrol.2006.09.003
- Monticolo, D. (2008). *Une approche organisationnelle pour la conception d'un système de destion de connaissances fondé sur le paradigme agent*. Thèse de doctorat, Université de Technologie de Belfort Montbéliard et Université de Franche Comté.
- Monticolo, D., Mihaita, S., Darwich, H., & Hilaire, V. (2014). An agent-based system to build project memories during engineering projects. *Knowledge-Based Systems*, 33. doi:10.1016/j.knosys.2013.12.022
- Mougenot, C. (2008). *Modélisation de la Phase d'Exploration du Processus de Conception de Produits: Pour une Créativité Augmentée*. Thèse de doctorat, Paris, Arts et Metiers ParisTech: Laboratoire Conception de Produits et Innovation.

- Nedjl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., & Nilsso. (2002). Edutella: A P2P networking infrastructure based on RDF. In *Proceedings of WWW-2002*,. New York, USA: ACM Press.
- Nonaka, I., & Takeuchi, H. (1995). *The knowledge-Creating Company : How Japanese Companies Create the Dynamics of Innovation* (Oxford Uni.). Oxford.
- Noy, N. F., McGuinness, D. L., & others. (2001). Ontology development 101: A guide to creating your first ontology. *Development*. Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880.
- Pahl, G., & Beitz, W. (1996). *Engineering Design - A Systematic Approach*. 1996. Springer-Verlag, London. Retrieved from <http://www.springer.com/engineering/mechanical+engineering/book/978-1-84628-318-5>
- Paraiso, E. C., & Barthès, J. P. (2006). An intelligent speech interface for personal assistants in R&D projects. *Expert Systems with Applications*, 31(4), 673–683.
- Parunak, H., & Odell, J. (2002). Representing Social Structures in UML. *Agent-Oriented Software Engineering II, LNCS 2222, Springer*, 1–16.
- Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004). WordNet:Similarity: measuring the relatedness of concepts. *Demonstration Papers at HLT-NAACL 2004, (Stroudsburg, PA, USA: Association for Computational Linguistics)*, 38–41.
- Perrin, J. (1999). *Pilotage et évaluation des processus de conception*. L'Harmattan.
- Plaza, E. (2005). Cooperative reuse for compositional cases in multi-agent systems. In *Case-Based Reasoning, Research and Development, 6th International Conference, on Case-Based Reasoning, ICCBR 2005, Chicago, IL, USA, August 23-26, 2005, Proceedings* (Vol. 01, pp. 1–16).
- Plaza, E., & Arcos, J.-L. (2001). Context-aware personal information agents. In M. Klusch & F. Zambonelli (Eds.), *Cooperative Information Agents V, Proceedings of the 5th International Workshop CIA 2001, 2182* (pp. 44–55). Heidelberg, Germany: Lecture Notes in Artificial Intelligence, Springer Verlag.
- Potes Ruiz, P., Kamsu Foguem, B., & Grabot, B. (2014). Generating knowledge in maintenance from Experience Feedback. *Knowledge-Based Systems*. doi:10.1016/j.knosys.2014.02.002
- Prax, J.-Y. (2000). *Le guide du Knowledge management – Concepts et pratiques de la gestion des connaissances*. (Dinod, Ed.). Ed. Dunod.
- Preece, A. D., Hui, K., Gray, W. A., Marti, P., Bench-Capon, T. J. M., Jones, D. M., & Cui, Z. (2000). The KRAFT architecture for knowledge fusion and transformation. *Knowledge Based Systems*, 13(3), 113–120.
- Qinglin, G., & Zhang, M. (2009). A novel approach for multi-agent-based Intelligent Manufacturing System. *Information Sciences*, 179(18), 3079–3090. doi:10.1016/j.ins.2009.05.009
- Raghu, T. S., & Vinze, A. (2007). A business process context for Knowledge Management. *Decision Support Systems*, 43(3), 1062–1079. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0167923605000965>

- Rao, H. R., & Chaudhury, A. (1995). Modeling team processes: issues and specific example. *Information Systems Research*, 6(3).
- Rao, L., Mansingh, G., & Osei-Bryson, K. M. (2012). Building ontology based knowledge maps to assist business process re-engineering. *Decision Support Systems*, 52, 577–589.
- Redman, T. C. (1992). *Data Quality: Management and Technology*. (Bantam, Ed.). USA. Retrieved from <http://www.abebooks.fr/Data-Quality-Management-Technology-Thomas-Redman/3390307129/bd>
- Reffay, C., Chanier, T., Noras, M., & Betbeder, M.-L. (2008). Contribution à la structuration de corpus d'apprentissage pour un meilleur partage en recherche. *Revue Sticef*, 15.
- Rhodes, R. J., & Maes, P. (2000). Just-in-time information retrieval agents. *IBM Systems Journal*, 39(3-4), 685–704.
- Roldán, M. L., Gonnet, S., & Leone, H. (2010). TracED: A tool for capturing and tracing engineering design processes. *Advances in Engineering Software*, 41(9), 1087–1109. doi:10.1016/j.advengsoft.2010.06.006
- Rosenman, M. A., Smith, G., Maher, M. L., Ding, L., & Marchant, D. (2007). Multidisciplinary collaborative design in virtual environments. *International Journal in Automation in Construction*, 16, 37–44.
- Russell, S., & Norvug, P. (2009). *Artificial Intelligence: A Modern Approach*. Prentice Hall.
- Sánchez, D., Solé-Ribalta, A., Batet, M., & Serratos, F. (2012). Enabling semantic similarity estimation across multiple ontologies: An evaluation in the biomedical domain. *The Journal of Biomedical Informatics*, 45, 141–155.
- Sanderson, P. M., & Fisher, C. (1994). Exploratory sequential data analysis: foundations. *Human-Computer Interaction*, 9(3), 251–317.
- Saucier, A. (1997). *Un modèle multi-vues du produit pour le développement et l'utilisation de systèmes d'aide à la conception en ingénierie mécanique*. E.N.S. de Cachan, France.
- Scaravetti, D. (2004). *Formalisation préalable d'un problème de conception, pour l'aide à la décision en conception préliminaire*. Thèse de Doctorat, École Nationale Supérieure d'arts et Métiers.
- Schreiber, A. (1992). *Pragmatics of the knowledge level*. Université d'Amsterdam, Pays-Bas. Retrieved from <http://www.few.vu.nl/~guus/papers/Schreiber92c.pdf>
- Serrafero, P., Gomes, S., Bonnivard, D., & Jézéquel, L. (2006). De la mémoire projet à la compétence métier : vers la synthèse de connaissances métier en ingénierie robuste des produits/process. In *6th International conference on Integrated Design and Manufacturing in Mechanical Engineering (IDMME'06)*. Grenoble.
- Settouti, L., Prié, Y., Mille, A., & Marty, J.-C. (2006a). Système à base de traces pour l'apprentissage humain. In *Colloque international des Technologies de l'Information et de la Communication dans l'Enseignement Supérieur et l'Entreprise, TICE*.

- Settouti, L., Prié, Y., Mille, A., & Marty, J.-C. (2006b). Système à base de traces pour l'apprentissage humain. In *Colloque international des Technologies de l'Information et de la Communication dans l'Enseignement Supérieur et l'Entreprise, TICE*.
- Shen, W. (2003). Editorial of the special issue on knowledge sharing in collaborative design environments. *Computers in Industry*, 52(1), 1–3. doi:10.1016/S0166-3615(03)00064-2
- Shen, W., Yong, J., Yang, Y., Barthes, J. P., & Luo (editors), J. (2008). Computer Supported Cooperative Work in Design IV. *LNCS*, 5236.
- Shumin, W., Hamada, G., Yue, Z., & Weiming, S. (2006). Personal assistant agents for collaborative design environments. *Computers in Industry*, 57(8-9), 732–739.
- Sless, L., Hazon, N., Kraus, S., & Wooldridge, M. (2014). Forming coalitions and facilitating relationships for completing tasks in social networks. In *AAMAS '14 Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems* (pp. 261–268). Paris, France.
- Staab, S., Schnurr, H. P., Studer, R., & Sure, Y. (2000). Knowledge Processes and Ontologies. *IEEE Intelligent Systems*, 16(n°1 Special Issue on Knowledge Management), 26–34.
- Studer, R., Benjamins, V., & Fensel, D. (1998). Knowledge engineering: principles and methods. *Data & Knowledge Engineering*, 25, 161–197. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0169023X97000566>
- Sun, H., Fan, W., Shen, W., & Xiao, T. (2012). Ontology-based interoperation model of collaborative product development. *Journal of Network and Computer Applications*, 35(1), 132–144. doi:10.1016/j.jnca.2011.02.012
- Tacla, C. (2003). *De l'utilité des systèmes multi-agents pour l'acquisition des connaissances au fil de l'eau*. Thèse de doctorat, Université de Technologie de Compiègne.
- Ten Bosch, O., Van der Wolf, P., & Van der Hoeven, A. (1995). Design Flow Management: More than Convenient Tool Invocation. *Electronic Design Automation Frameworks IFIP — The International Federation for Information Processing, Springer*, 4, 149–158.
- Tichkiewitch, S. (1996). Specifications on integrated design methodology using a multi-view product model. In *3rd Biennial Joint Conference on Engineering System Design and Analysis Conference, ASME'96, Montpellier* (pp. 101–108).
- Tka, M., & Ghannouchi, S. A. (2012). Comparison of Business Process Models as Part of BPR Projects. *Procedia Technol.*, 5, 427–436.
- Tsai, H.-H. (2013). Knowledge management vs. data mining: Research trend, forecast and citation approach. *Expert Systems with Applications*, 40(8), 3160–3173. doi:10.1016/j.eswa.2012.12.029
- Tsui, E., Garner, B., & Staab, S. (2000). The role of artificial intelligence in knowledge management. *Knowledge-Based Systems*, 13, 235–239.
- Van der Aalst, W. M. P., Ter hofstede, A. H. M., Kiepuszewski, B., & Barros, A. P. (2003). Workflow Patterns. *Distributed and Parallel Databases*, 14(1), 5–51. doi:10.1023/A:1022883727209

- Van Elst, L., Dignum, V., & Abecker, A. (2004). Towards Agent-Mediated Knowledge Management. *Agent-Mediated Knowledge Management, LNCS, Springer, 2926*, 1–30. doi:10.1007/978-3-540-24612-1_1
- Vargas, C. (1995). *Modélisation du processus de conception en ingénierie des systèmes mécaniques. Application à la conception d'une culasse automobile*. Thèse de doctorat, E.N.S. Cachan, France.
- Wang, J. X., & Tang, M. X. (2005). Towards Agent-Mediated Knowledge Management. In *Computer-aided Conceptual Design (CAID & CD 2005)*. Delft, The Netherlands.
- Weber, R., Aha, D. ., & Becerra-Fernandez, I. (2001). Intelligent lessons learned systems. *Expert Systems with Applications, 20*(1), 17–34. doi:10.1016/S0957-4174(00)00046-4
- Wooldridge, M. J. (2000). The Gaia Methodology for Agent-Oriented Analysis and Design. *Journal of Autonomous Agents and Multi-Agent Systems, 3*(3), 285–312.
- Wooldridge, M. J. (2002). *An Introduction to Multiagent Systems*. New York, USA: John Wiley.
- Wooldridge, M. (1999). Intelligent Agents. In G. Weiss (Ed.), *Multiagent Systems – A Modern Approach to Distributed Artificial Intelligence* (pp. 27–78). Cambridge, Massachusetts: MIT Press.
- Yannou, B., & Bonjour (sous la direction de), É. (2006). *Evaluation et décision dans le processus de conception. 2006* (éd. Hermès.). Lavoisier. Retrieved from <http://www.lavoisier.fr/livre/economie/evaluation-et-decision-dans-le-processus-de-conception/yannou/descriptif-9782746213944>
- Yu, B., Venkatraman, M., & Singh, M. P. (2003). An adaptive social network for information access: Theoretical and experimental results. *Journal of the Applied Artificial Intelligence, 17*(1), 21–38.
- Yu, E. (1995). *Modelling Strategic Relationships for Process Reengineering*. PhD Thesis, University of Toronto, Department of Computer Science.
- Zacklad, M., & Grundstein, M. (2001). *Management des connaissances : modèles d'entreprise et applications. Coll. IC2. Hermès (Hermès).* collection IC2.
- Zambonelli, F., Jennings, N. R., & Wooldridge, M. J. (2001). Organisational Rules as an Abstraction for the Analysis and Design of Multi-Agent Systems. *International Journal of Software Engineering and Knowledge Engineering, 11*(3), 303–328.
- Zambonelli, F., Jennings, N. R., & Wooldridge, M. J. (2003). Developing multiagent systems: The Gaia methodology. *ACM Trans. Software Eng. Meth., 12*(3), 417–470.

Sommaire des figures présentées dans ce mémoire

Figure 1.	Plan du document	9
Figure 2.	Exemple de modèle processus (Harani, 1997)	20
Figure 3.	Éléments du langage BPMN (Chinosi & Trombetta, 2012)	21
Figure 4.	Cycle traçabilité – exploitation en conception	23
Figure 5.	Modèle KnowledgeScope d'une mémoire organisationnelle (Kwan & Balasubramanian, 2003)	24
Figure 6.	Ontologie organisationnelle du processus	26
Figure 7.	Cartes de connaissances pour les processus de conception (L. Rao et al., 2012)	27
Figure 8.	Le contexte processus métier dans la gestion de connaissances (Raghu & Vinze, 2007)	28
Figure 9.	Système à base de traces pour une plateforme de maintenance (M. H. Karray, 2012)	31
Figure 10.	Modèle de traces selon (Champin et al., 2003)	32
Figure 11.	Modèle des traces d'activité ABSTRACT (OL Georgeon, Mille, & Bellet, 2012)	34
Figure 12.	L'environnement MUSETTE	42
Figure 13.	Le système KARE vu à partir de ses objectifs en fonction des agents	43
Figure 14.	Notre contribution, à l'intersection de différentes thématiques ou techniques d'intelligence artificielle	46
Figure 15.	Cycle de gestion des connaissances selon (Grundstein, 2002)	48
Figure 16.	Double cycle de capitalisation des connaissances pour des activités instrumentées	50
Figure 17.	Approche globale de gestion des démarches métier	53
Figure 18.	Modèle conceptuel du point de vue des métiers	56
Figure 19.	Modèle conceptuel du point de vue étude	57
Figure 20.	Modèle conceptuel de la réalisation d'une étude	59
Figure 21.	Vue globale de l'ontologie OntoProcess	60
Figure 22.	Exemple d'instances de traces et de processus à partir de l'ontologie OntoProcess	62
Figure 23.	Exemple d'instances trace et rôle métier à partir de l'ontologie OntoProcess (suite de Figure 22)	63
Figure 24.	Extension de l'Ontologie OntoProcess pour les activités d'Analyse Fonctionnelle Externe	64
Figure 25.	Extension de l'ontologie OntoProcess pour les activités d'Analyse Fonctionnelle Interne	65
Figure 26.	Extension de l'Ontologie OntoProcess pour les activités d'AMDEC Produit	65
Figure 27.	Extension de l'Ontologie OntoProcess pour les activités d'AMDEC Process	66
Figure 28.	Classes de l'Ontologie OntoProcess représentées sous Protégé	68
Figure 29.	Propriétés et attributs utilisés dans OntoProcess, représentée sous Protégé	68
Figure 30.	Le processus utilisé dans Gaia (Wooldridge, 2000)	72
Figure 31.	La méthodologie PASSI (Cossentino, 2005)	75
Figure 32.	Modélisation d'une organisation à l'aide du formalisme RIOCK	78
Figure 33.	Liste des fonctionnalités que doit assurer le système multi-agent	79
Figure 34.	Opérations réalisées par l'agent ET	81
Figure 35.	Opérations réalisées par l'agent ARAO	82
Figure 36.	Opérations réalisées par l'agent CPO	83
Figure 37.	Opérations réalisées par l'agent CPODM	84
Figure 38.	Opérations réalisées par l'agent « Assistant Expert Méthodologique »	85
Figure 39.	Opérations réalisées par l'agent « Assistant Processus Métier »	86
Figure 40.	Organisation « Gestion des traces »	89
Figure 41.	Diagramme de séquence de l'organisation « gestion des traces »	90
Figure 42.	Organisation « Gérer les Processus Opérationnels & les Démarches Métier »	91
Figure 43.	Diagramme de séquence de l'organisation	92
Figure 44.	Organisation « Assister les utilisateurs et aide à la décision »	93
Figure 45.	Diagramme de séquence de l'organisation « Assister les utilisateurs et aide à la décision »	94
Figure 46.	Scénario de construction et de visualisation d'un processus opérationnel	97
Figure 47.	Exemple de processus opérationnel lié au cas test 1	98

Figure 48.	Exemple de processus opérationnel lié au cas test 2	99
Figure 49.	Exemple de processus opérationnel lié au cas test 3	100
Figure 50.	Exemple de processus opérationnel lié au cas test 4	101
Figure 51.	Structure modulaire du Système à Base de Traces	102
Figure 52.	Structure fonctionnelle du Système à Base de Traces	103
Figure 53.	Modèle des traces	104
Figure 54.	Extrait de l'arbre hiérarchique des entités de conception	108
Figure 55.	Le type_Lien « Lien Composant/Opération »	109
Figure 56.	Extrait du fichier XML de traces des actions portant sur une entité opération et ses entités filles.	110
Figure 57.	Extrait du fichier XML de traces illustrant une matrice de liens composant / opération	111
Figure 58.	Requêtes SPARQL	112
Figure 59.	Architecture fonctionnelle du module de transformation des traces en processus	113
Figure 60.	Implémentation de l'Agent CPO	117
Figure 61.	Initialisation du fichier BPMN	118
Figure 62.	Interface « Consulter le référentiel/Construire la démarche métier »	122
Figure 63.	Interface « Enrichir une démarche métier »	123
Figure 64.	Interface « visualiser le processus opérationnel »	124
Figure 65.	Interface « demander de l'assistance »	124
Figure 66.	Extrait du tableau des fonctions de service	126
Figure 67.	Extrait du tableau de l'historique des actions	126
Figure 68.	Extrait des traces XML structurées	127
Figure 69.	Visualisation du processus opérationnel construit pour l'étude de cas 1	127
Figure 70.	Visualisation du processus opérationnel construit pour l'étude de cas 2	128
Figure 71.	Visualisation détaillée du sous-processus	129
Figure 72.	Fenêtre d'affichage global	129
Figure 73.	Visualisation du processus opérationnel construit pour l'étude de cas 3	130
Figure 74.	Avancement du développement du Système Multi-agents	133
Figure 75.	Interface « Générer un planning de projet »	135
Figure 76.	Interface « suivre l'avancement du processus »	135

Résumé de la thèse

Il est bien connu qu'une nouvelle étude, menée au sein d'une entreprise, est souvent semblable à une étude précédente et par conséquent, peut être structurée selon un processus de référence commun au type d'étude correspondant. Dans notre travail, nous l'avons appelé démarche métier. Il s'agit d'une bonne pratique de réalisation d'un type d'études.

La difficulté majeure réside dans la formalisation de cette démarche métier. Les approches traditionnelles de capitalisation des connaissances s'appuyant sur des verbalisations d'experts ont montré leur limite. Souvent réalisées en dehors de l'activité réelle, les experts omettent des détails qui peuvent être d'importance.

Notre thèse repose sur l'idée qu'il est possible de construire le processus opérationnel mis en œuvre lors des activités collaboratives de conception, à partir des traces enregistrées lors de l'utilisation de l'outil numérique par les acteurs métier. Le processus opérationnel ainsi construit pourra aider les acteurs métiers et les experts à prendre du recul sur le travail réel et à formaliser et enrichir les démarches métier de l'entreprise.

Notre travail s'est déroulé au sein du laboratoire ERPI (Equipe de Recherche sur les Processus Innovatifs) de l'Université de Lorraine et en collaboration avec la société TDC Software dans le cadre d'une thèse CIFRE.

Les contributions que nous proposons sont les suivantes :

- Un double cycle de capitalisation pour des activités instrumentées,
- Une approche globale de gestion des démarches métier,
- Une ontologie *OntoProcess* modélisant des aspects organisationnels génériques (séparant clairement des concepts liés aux traces et d'autres liés aux démarches métier) et des extensions métiers spécifiques à l'outil utilisé,
- Un système multi-agents supportant l'approche globale de gestion des démarches métiers et s'appuyant sur l'ontologie *OntoProcess*,
- Un système à base de traces permettant de construire un processus opérationnel à partir des traces enregistrées lors d'une étude.

Abstract

It is well known in the enterprises that each new project to be carried out is usually similar to a certain previous projects. Those projects can be structured according to a common reference process depending on their type. In the dissertation, we call this reference process of the enterprise's good practices the "expertise business process".

The main difficulty lies in the formalization of the expertise business process. The traditional knowledge capitalization approaches based on the experts' debriefings showed their limits: the experts often leave out details which may be of relevance because the debriefings are habitually realized externally to the activities.

Our thesis relies on the idea that it is possible to construct the operational process, implemented during the collaborative activities in a product development study, from the traces recorded by the used IT tools. The constructed operational process allows the business actors and experts to step back on their work and formalize the new deducted experience to enhance the expertise business processes of the firm.

Our work had taken place in the ERPI (Equipe de Recherche sur les Processus Innovatifs) laboratory of the "Université de Lorraine" under a partnership with TDC Software society and through a CIFRE Convention.

This dissertation offers five key contributions:

- A double cycle to capitalize over the instrumented activities.
- A global approach for the management of expertise business processes.
- An ontology "OntoProcess" to conceive the generic organizational aspects, separating distinctly the concepts related to traces from those related to the business process, and providing extensions in function of the used tools.
- A multi-agents system based on the ontology "OntoProcess" to support the presented global approach of the expertise business processes management.
- A trace based system that allows the construction of the operational process from the traces registered over the study.