

Conception et Réalisation d'une Plateforme de Gouvernance des Systèmes d'Information à base des Workflows inter- organisations, du Web sémantique et des Systèmes Multi Agents.

Meriyem Chergui

► **To cite this version:**

Meriyem Chergui. Conception et Réalisation d'une Plateforme de Gouvernance des Systèmes d'Information à base des Workflows inter- organisations, du Web sémantique et des Systèmes Multi Agents.. Informatique [cs]. ENSEM, 2017. Français. tel-01756441

HAL Id: tel-01756441

<https://hal.archives-ouvertes.fr/tel-01756441>

Submitted on 2 Apr 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.



Université Hassan II –Casablanca
Ecole Nationale Supérieure d'Électricité
Et de mécanique
CED : Sciences de l'ingénieur



THESE

En vue de l'obtention du grade de :

Docteur de l'Université Hassan II Casablanca
Ecole Nationale Supérieure d'Electricité & de Mécanique
Centre d'Etudes Doctorales Sciences de l'Ingénieur
(Arrêté n° 1371- 07 du 23 Septembre 2008)

Formation Doctorale : Génie Informatique

Présentée Par : CHERGUI MERIYEM

Conception et Réalisation d'une Plateforme de Gouvernance des Systèmes d'Information à base des Workflow Inter- organisations, du Web sémantique et des Systèmes Multi- Agents

Soutenue le 11/07/2017 à 10h devant le Jury :

Pr. Hamid EL OUARDI	ENSEM, Casablanca	Président
Pr. Abdellah LAKHOULI	FST, Settat	Rapporteur
Pr. Abdelkarim AMMOUMOU	EST, Casablanca	Rapporteur
Pr. Hicham BEHJA	ENSEM, Casablanca	Rapporteur
Pr. Siham BENHADOU	ENSEM, Casablanca	Examineur
Pr. Hamid NAHLA	ENSEM, Casablanca	Examineur
Pr. Hicham MEDROMI	ENSEM, Casablanca	Directeur de thèse
Pr. Adil SAYOUTI	Ecole Royale Navale	Co-directeur de thèse

Remerciements

Les travaux présentés dans cette thèse ont été effectués au sein de l'Equipe Architecture des Systèmes du Laboratoire de Recherche en Ingénierie de l'Ecole Nationale Supérieure d'Electricité et Mécanique de l'Université Hassan II - Casablanca.

Je remercie mon directeur de thèse Monsieur Hicham MEDROMI, je le prie d'accepter à cette occasion le témoignage de ma reconnaissance pour ses conseils prodigués tout au long de mes recherches, et sa participation au jury de cette thèse.

J'adresse toute ma gratitude à Monsieur Adil SAYOUTI, Co-directeur de thèse, pour ses efforts pour la réussite de ce projet ainsi que la richesse de ses conseils tout au long de mon parcours.

Je tiens également à remercier le Professeur Hamid NAHLA pour sa contribution enrichissante à se travail.

Mes sincères remerciements vont à Messieurs les membres du jury pour m'avoir honorée en acceptant d'évaluer ce travail.

Un remerciement spécial à mon équipe FMPC pour leurs conseils et leur soutien tout au long de cette thèse. Plus spécialement Monsieur le Doyen, Pr.Farid CHEHAB, Madame la Vice doyenne de la recherche Pr.Nadia TAHIRI JOUTI et Monsieur le Vice doyen des affaires pédagogiques Pr Mohammed BENNANI OTHMANI.

Je ne peux clôturer mes remerciements sans me retourner vers les personnes qui me sont le plus cher et qui ont un rôle essentiel dans ma réussite :

Merci à mon cher père pour son inspiration et sa confiance en moi,

Merci à ma chère mère pour son amour et ses prières,

Merci à mon cher mari pour son soutien quotidien, ses encouragements aux moments les plus difficiles et sa compréhension.

Merci à mes deux chers frères,

Merci à toute ma famille,

Acronymes

ACL	: Agent Communication Language.
AGR	: Agent Group Role.
AMS	: Agent Management System.
AUML	: Agent Unified Modelisation Langage (Langage de Modélisation Unifié d'Agent).
BDI	: Belief Desire Intention.
BSC	: Balanced Scorecard.
CMMI	: Capability Maturity Model Integration.
COBIT	: Control Objectives for Business Information and related Technology.
COSO	: Committee of Sponsoring Organizations de la Treadway Commission.
CPU	: Central Processing Unit.
DF	: Directory Facilitator.
ERP	: Enterprise Ressources Planning.
FIPA	: Foundation for Intelligent Physical Agents (Organisme de normalisation pour les agents et systèmes multi-agents).
GSI	: Gouvernance des systèmes d'information.
GTI	: Gouvernance des Technologies de l'information.
HTTP	: Protocole de transfert hypertexte.
IA	: Intelligence artificielle.
IAD	: Intelligence artificielle distribuée.
IBM	: International Business Machines.
IHM	: Interface Homme Machine.
ISACA	: Information Systems Audit and Control Association.
ISO	: International system organisation (Organisation internationale de normalisation).
IT GRC	: Information Technology Governance Risk Conformity.
ITGI	: IT Governance Institute.

ITIL : IT Infrastructure Library.

JADE : Java Agent Development Framework.

KIF : Knowledge Interchange Format.

KQML : Knowledge Query and Manipulation Language.

MADKIT : Multi-Agents Development Kit.

OIL : Ontology Inference Layer ou Ontology Interchange Language.

OILEd : Ontology Inference Layer Editor.

OWL : Web Ontology Language.

OWL-S : Web Ontology Language for Web Services.

PMBOK : Project Management Body of Knowledge.

PMI : Project Management Institute.

RDF : Resource Description Framework.

RDFS : Resource Description Framework Schema.

SHOE : Simple HTML Ontology Extensions.

SI : Système d'information.

TI : Technologie de l'information.

UML : Unified Modelisation Langage.

W3C : World Wide Web Consortium.

WF : Workflow.

WIO : Workflow inter-organisations.

XML : Extensible Markup Language.

XPDL : XML Process Definition language.

YAWL : Yet Another Workflow Language.

Résumé

Les technologies de l'information et de la communication sont devenues de nos jours une nécessité incontournable de l'entreprise. L'un des défis majeurs des organisations et de leurs managers est la mise en place d'un système d'information (SI) répondant efficacement aux attentes métiers tout en maîtrisant ses impacts sur la productivité et la performance. Cependant, l'évolution économique et technologique du marché actuel, la conformité aux lois et aux réglementations ainsi que les risques potentiels de l'environnement, imposent une remise en question itérative du SI. D'où la nécessité d'une omniprésence de la gouvernance des systèmes d'informations (GSI).

L'objectif de cette thèse est la conception et la réalisation d'une architecture de gouvernance des technologies de l'information répondant aux critères suivants :

- Adaptabilité à tout type de SI,
- Intelligence et autonomie pour comprendre les besoins métiers continuellement en changement,
- Distribution pour impliquer les parties prenantes et les composantes hétérogènes du SI,
- Evolutivité pour cumuler le savoir faire de l'entreprise en matière de GSI.

Les solutions de GSI existantes présentent plusieurs limitations à savoir :

- Spécialisation dans un métier particulier de l'entreprise,
- Exigence d'une gestion intégrée du SI,
- Implémentations rigides de l'un des référentiels de la gouvernance des technologies de l'information (TI)
- Nécessité des pré-requis de GSI de la part des utilisateurs pour pouvoir les exploiter.

Pour y remédier, nous présentons une architecture distribuée, basée sur les workflows inter-organisations (WIO) et les systèmes multi-agents et dotée d'un moteur sémantique qui traduit les objectifs métiers réels en requête interprétable par l'ensemble des référentiels de la gouvernance SI.

Cette solution représente la couche stratégique de la plateforme générique de l'équipe architecture des systèmes (EAS) nommée « EAS IT GRC » ayant comme fonctionnalité principale la détermination du meilleur référentiel de Gouvernance Risque et Conformité SI.

Pour la réalisation de cette solution nous avons procédé comme suit :

- La conception d'un WIO de type tâche spécifique à la GSI,
- La modélisation d'un système expert de Médiation intégré au WIO,
- La mise en place d'un processus de modélisation d'ontologie de gouvernance SI à bases des référentiels et bonnes pratiques,
- La création de « IT Governance Ontology » servant de base au moteur sémantique de la plateforme.

Mots clés : Gouvernance des Système d'information, Gouvernance Risque Conformité SI, Workflow, Inter-Organisation, Système multi-agents, Ontologie, Web sémantique, Système expert, référentiels.

ABSTRACT

Information and communication technologies have become an indispensable necessity of the company today. One of the major challenges facing organizations and their managers is the implementation of an Information System (IS) that effectively meets business expectations while controlling its impact on productivity and performance. However, the economic and technological evolution of the current market, compliance with laws and regulations as well as the potential risks of the environment, require an iterative questioning of the IS. Hence the need of the information *systems governance (ISG)*.

The objective of this thesis is to set up a new IT governance platform that meets the following criteria:

- Adaptability to any type of IS,
- Intelligence and autonomy to understand continuously changing business needs,
- Distribution to involve stakeholders and heterogeneous components of the IS,
- Scalability to combine the know-how of the company with regard to ISG.

Existing GSI solutions have several limitations, namely:

- Specialization in a particular business of the company,
- Requirement for integrated IS management (ERP module),
- Rigid implementations of one of the repositories of IT governance,
- Necessity of the ISG pre-requisites on the part of the users in order to use it.

To solve this problem, we present a distributed architecture, based on the Inter-Organization Workflows (IOW) and the multi-agent systems and with a semantic engine that translates the real business objectives into a query that can be interpreted by all the repositories of the Governance.

This solution represents the strategic layer of the EAS IT GRC platform of the 'System Architecture' team, whose main function is to determine the best repository of Risk Governance and SI Compliance.

For the realization of this solution we proceeded as follows:

- Design of an inter-organizational workflow of the loose type specific to the GSI,
- Integrated mediation expert system modeling in the IOW,
- Implementation of a modeling process of IS governance ontology based on references and best practices.
- creation of "ITGovernanceOntology" as the basis for the semantic engine of the platform

Key words: Information system governance, System Governance Risk and Compliance, Inter-Organization workflow, Multi-agent system, ontology, semantic web, Expert system, repositories.

Table des matières

Remerciements	i
Acronymes	ii
Résumé	iii
Abstract.....	iv
Liste des Figures.....	v
Liste des Tableaux.....	iv
Introduction Générale.....	1
Partie I : Etat de l’art	
Chapitre 1 : Gouvernance des Systèmes d’Information	
I.1. Introduction.....	6
I.2. Gouvernance des Systèmes d’information.....	6
I.3. Référentiels de Gouvernance des systèmes d’information	9
I.3.1. Committee of Sponsoring Organizations of the Treadway Commission	10
I.3.2. IT Balanced Scorecard	11
I.3.3. Famille ISO 27000	12
I.3.3.1. ISO 27001	12
I.3.3.2. ISO 27002	13
I.3.4. IT Infrastructure Library - ITIL	14
I.3.5. Control Objectives for Business Information and related Technology- COBIT	14
I.4. Complémentarité des référentiels de Gouvernance des Systèmes d’Information	16
I.4.1. Relations entre référentiels	16
I.4.1.1. COBIT-BSC.....	17
I.4.1.2. COBIT-ITIL.....	17
I.4.1.3. COBIT /ISO 27002/ ISO 27001 / ISO 9001	18
I.4.1.4. COBIT /PMBOK/ CMMI.....	19
I.4.2. Aspect orienté processus.....	20
I.5. Implémentation de Gouvernance SI dans l’entreprise	21

I.5.1. Vision.....	22
I.5.2. Positionnement.....	23
I.5.3. Outils et Structures.....	23
I.5.4. Evaluation et comparaison.....	23
I.5.5. Référentiels et bonnes pratiques	24
I.5.6. Evolution et innovation.....	24
I.6. Étude empirique: spécificités des SI au Maroc pour une meilleure gouvernance de l'IT.....	24
I.7. Plateforme de Gouvernance des Systèmes d'Information.....	29
I.7.1. Fonctionnalités et Atouts des solutions.....	29
I.7.2. Etude Comparative.....	31
I.8. Problématique	32
I.9. Conclusion	34
Chapitre 2 : Workflows Inter-organisations et systèmes multi-agents	
II.1. Workflows et Workflows Inter-organisationnels	36
II.1.1. Introduction	36
II.1.2. Terminologie	36
II.1.3. Architecture et langages	40
II.1.3.1. Architecture de référence.....	40
II.1.3.2. Langages de modélisation des WF	42
II.1.4. Conception des WIO	45
II.1.5. Coordination dans les systèmes WIO.....	47
II.1.6. Scénarios de Workflow Inter-organisationnel.....	49
II.1.7. Synthèse.....	50
II.2. Agent et Système Multi-Agents	50
II.2.1. Terminologie autour des systèmes multi-agents	51
II.2.1.1. Agent	51
II.2.1.2. Agent Versus Objet	52
II.2.1.3. Systèmes multi-agents	53
II.2.2. Problématiques et apport des systèmes multi-agents	54
II.2.3. Catégories d'agents.....	55
II.2.3.1. Agents réactif.....	56
II.2.3.2. Agent cognitif.....	56
II.2.3.3. Agent hybride	57
II.2.4. Architecture et Méta-modèle d'agents intelligents.....	58

II.2.4.1. Belief Desire Intention - BDI	58
II.2.4.2. Agent Groupe Rôle - AGR	59
II.2.5. Interaction entre Agents et Environnement	61
II.2.5.1. Communication	61
II.2.5.2. Coopération	64
II.2.5.3. Coordination	66
II.2.6. Plateforme de développement des Systèmes multi-Agents	67
II.2.6.1. La plate-forme ZEUS	67
II.2.6.2. La plateforme Jade.....	68
II.2.6.3. La plate-forme MadKit.....	69
II.2.6.4. La plate-forme AgentBuilder.....	70
II.2.6.5. Choix de la plateforme multi-agents	71
II.3. Conclusion.....	72
 Partie II: Solution proposée	
Chapitre 3 : Architecture proposée du Workflow de gouvernance SI	
III.1. Introduction	75
III.2. Plateforme EAS IT- GRC	75
III.2.1. Présentation	75
III.2.2. Description	76
III.2.3. Contribution de la solution proposée dans la plateforme EAS IT-GRC	79
III.3. Fonctionnalités et Avantages	81
III.4. Modélisation du WIO	81
III.4.1. Niveau Organisationnel.....	81
III.4.2. Niveau fonctionnel	83
III.4.3. Niveau Comportemental	83
III.5. Architecture proposée	84
III.5.1. EAS IT Stratégie –Version 1.....	84
III.5.2. EAS IT Stratégie –Version 2.....	85
III.5.3. EAS IT Stratégie –Version 3.....	89
III.5.3.1. Présentation	89
III.5.3.2. Description	89
III.6. Conclusion.....	93
 Chapitre 4: Nouvelle ontologie de gouvernance SI	
IV.1. Introduction.....	95

IV.2. Web Sémantique et Ontologies.....	95
IV.2.1. Ontologie.....	96
IV.2.2. Types d'ontologies.....	96
IV.2.3. Langages de représentation et outils de construction.....	97
IV.2.3.1. Langages de représentation des ontologies.....	97
IV.2.3.2. Outils de construction des ontologies.....	99
IV.2.3.3. Méthodes de conception d'ontologies.....	99
IV.2.4. Synthèse.....	99
IV.3. Modélisation « ITGovernanceOntology ».....	100
IV.3.1. Introduction.....	100
IV.3.2. Spécifications.....	101
IV.3.3. Acquisition de Connaissances.....	101
IV.3.4. Conceptualisation.....	104
IV.3.5. Implémentation.....	106
IV.3.6. Evaluation et synthèse.....	107
IV.4. Conclusion.....	109
Chapitre 5: Système Expert de médiation sémantique	
V.1. Introduction.....	111
V.2. Contexte.....	111
V.3. Systèmes de Médiation.....	112
V.3.1. Agents Médiateurs.....	114
V.4. Algorithmes de matchmaking sémantique.....	114
V.5. System Expert.....	115
V.5.1. Terminologie.....	115
V.5.2. Types des Systèmes Expert.....	117
V.5.3. Limite des Systèmes Expert.....	117
V.5.4. Systèmes Expert et Systèmes multi-agents.....	117
V.6. Architecture de médiation proposée.....	118
V.6.1. Entité de persistance.....	119
V.6.2. Entité de traitement sémantique.....	119
V.6.3. Entité de médiation.....	120
V.7. Application au contexte de la Gouvernance IT.....	120
V.7.1. Introduction.....	120
V.7.2. Modélisation de la requête.....	121

V.7.3. Algorithme de Comparaison	121
V.7.4. Priorisation	122
V.7.5. Exemple.....	123
V.8. Conclusion.....	123

Partie III: Conception et réalisation

Chapitre 6: Conception et réalisation de la plateforme

VI.1. Introduction	126
VI.2. Contexte	126
VI.3. Modélisation Orienté Objet.....	127
VI.4. Modélisation Orienté Agent.....	128
VI.5. Spécifications fonctionnelles et techniques	129
VI.5.1. Description des cas d'utilisation	129
VI.5.2. Description des Scénarii.....	131
VI.6. Analyse et Conception	133
VI.6.1. Diagrammes de séquences	133
VI.6.2. Diagrammes de classes	136
VI.7. Réalisation.....	138
VI.7.1. Introduction	138
VI.7.2. Environnement de Développement.....	139
VI.7.3. Phases de développement.....	142
VI.7.4. Description des interfaces	143
VI.7.5. Test et Intégration	146
VI.7.6. Evaluation de la plateforme	148
VI.7.6.1. Evaluation quantitative	148
VI.7.6.2. Evaluation qualitative	149
VI.7.7. Conclusion	150
Bilan	152
Originalité.....	153
Limites.....	153
Perspectives.....	154
Bibliographie.....	155
Annexe I : Correspondance Objectifs métiers, Objectifs Informatiques et Processus IT	165
Objectifs Informatiques et Processus IT	165
Objectifs métiers Objectifs Informatiques	166

Annexe II : Questionnaire de la Gouvernance des systèmes d'information.....	168
Annexe III : Mise en œuvre de « ITGovernanceOntolgy » sur Protégé	174
Annexe IV : Publications effectuées dans le cadre de la thèse	178

Liste des Figures

Figure 1: Normes de la famille ISO27	12
Figure 2: Liaisons entre composants COBIT	15
Figure 3: COBIT et ITIL	17
Figure 4: Couverture des processus COBIT par ITIL	18
Figure 5: Secteur d'activité des entreprises de l'étude	25
Figure 6: Gestion des processus et force du SI	26
Figure 7: Force du SI par rapport à sa pertinence et sa couverture des processus métiers	26
Figure 8: Faiblesses du SI par rapport à sa pertinence et sa couverture des processus métiers	27
Figure 9: Analyse factorielle de la gestion des processus métiers	28
Figure 10: Relation entre notions autour des workflows	38
Figure 11: Architecture de référence du Workflow	40
Figure 12: Méta-modèle organisationnel	45
Figure 13: Exécution d'activité au niveau informationnel	47
Figure 14: Interaction d'un agent avec son environnement	52
Figure 15: Système multi-agents	54
Figure 16: Architecture d'agent réactif	56
Figure 17: Architecture d'agent cognitif	57
Figure 18: Architecture d'un agent hybride	58
Figure 19: Architecture Belief-Desire-Intention	59
Figure 20: Diagramme UML du Méta-Modèle AGR	60
Figure 21: Représentation en couches de KQML	62
Figure 22: Performatives KQML	63
Figure 23: Format de message ACL	64
Figure 24: Plateforme Zeus	67
Figure 25: Plateforme Jade	69
Figure 26: Plateforme Madkit	70
Figure 27: Plateforme Agent Builder	71
Figure 28: Architecture GlobalE EAS IT-GRC	78
Figure 29: Modèle de Communication du Workflow de Gouvernance IT	80
Figure 30: Modèle Organisationnel du WIO	82
Figure 31: Liaisons fonctionnelles des composants COBIT au sein du WIO	83
Figure 32: Diagramme de séquence WIO	83
Figure 33: Architecture proposée version 1	84
Figure 34: Architecture proposée version 2	86
Figure 35: Architecture agent médiateur	87
Figure 36: Couche Framework de l'architecture proposée version 3	90
Figure 37: Architecture proposée version 3	92
Figure 38: Relation entre les types d'ontologies	97
Figure 39: Langages de représentation d'ontologie W3C	98
Figure 40: Relation entre les composants de COBIT	103
Figure 41: Le diagramme de classes des concepts COBIT	105
Figure 42: Vue d'IT Governance Ontology sur OWLViz	106

Figure 43: Aperçu du code OWL d'IT Governance Ontology	107
Figure 44: Classes inférées par Fact ++ d'IT Governance Ontology	107
Figure 45: Résultats requêtes ITGovernanceOntology sur Solr.....	108
Figure 46: Architecture d'un système expert.....	116
Figure 47: Architecture du système expert de médiation sémantique	119
Figure 48: Diagramme cas d'utilisation utilisateurs	130
Figure 49: Diagramme cas d'utilisation Agents	131
Figure 50: Diagramme de séquence fonctionnel.....	133
Figure 51: Diagramme de séquence globale Agents.....	134
Figure 52: Diagramme de séquence SMA SI Manager et Agent Workflow SI.....	135
Figure 53: Diagramme de séquence SMA Matchmaking	135
Figure 54: Diagramme de séquence Framework Manager	136
Figure 55: Diagramme de classe couche SI	136
Figure 56: Diagramme de classe couche Framework	137
Figure 57: Diagramme de classe couche Médiation	137
Figure 58: Exemple de classe Agent.....	138
Figure 59: Structure du projet dans l'IDE.....	141
Figure 60: Environnement d'analyse sémantique.....	141
Figure 61: Environnement de développement de l'ontologie.....	142
Figure 62: Authentification	143
Figure 63: Gabarit de l'application et le menu principal	144
Figure 64: Configuration statique de l'environnement.....	144
Figure 65: Création de la requête stratégique.....	145
Figure 66: Résultats de la demande.....	146
Figure 67: Temps de réponse de la couche de médiation	149

Liste des Tableaux

Tableau 1.1:	Définitions de la Gouvernance des Systèmes d'Information	7
Tableau 1.2	Référentiels de Gouvernance par domaine	8
Tableau 1.3	Ébauche du BSC appliqué à la GTI	11
Tableau 1.4	Correspondance COBIT-BSC	15
Tableau 1.5	Processus COBIT de sécurité de l'information.....	18
Tableau 1.6	Facteurs de dépendance et importance en GTI	22
Tableau 1.7	Projection des fonctionnalités GTI sur les acteurs de l'entreprise	29
Tableau 1.8	Solutions IT GRC	30
Tableau 1.9	Comparaisons solutions IT GRC	31
Tableau 2.1	Modèles et langages de Workflows	42
Tableau 2.2	Approche d'interopérabilité et de coordination WIO	49
Tableau 2.3	Formes de coordination WIO	50
Tableau 2.4	Différences entre un Objet et un Agent	53
Tableau 2.5	Types d'interaction SMA	65
Tableau 2.6	Comparatifs des plateformes Multi-agents	72
Tableau 3.1	Agents et rôle du WIO de GTI	82
Tableau 3.2	Types des agents et justification du choix	87
Tableau 3.3	Mode de communication des agents	88
Tableau 4.1	Résultat de l'indexeur MAUI appliqué à COBIT 4.1.....	102
Tableau 5.1	Comparaison des deux modes de médiation	113
Tableau 5.2	Comparaison des agents médiateurs	114
Tableau 5.3	Résultats du matchmaking	123
Tableau 5.4	Comparaison des résultats du matchmaking et recherche sémantique	123
Tableau 6.1	Offres adéquates à la requête	147
Tableau 6.2	Objectifs informatiques priorisés	147

INTRODUCTION GENERALE

Introduction Générale

Les systèmes d'information (SI) ont connu une évolution exponentielle aussi bien en termes de complexité architecturale qu'en termes de taille de données traitées. Au début des années 80, le système d'information était une application informatique qui se focalisait sur la méthode de développement et les techniques utilisées. Par la suite, l'intérêt s'est porté sur la théorisation de la production avec reformulation des problématiques d'ordre cumulatif, telles que les références, les techniques de travail et l'assemblage des connaissances. A partir des années 90, des études de cas autour des SI ont eu lieu concernant l'organisation et la gestion des SI, la collaboration, et l'utilisation des nouvelles technologies.

Avec l'apparition des Systèmes d'Information Complexe, de nouvelles problématiques sont apparues telles que l'hétérogénéité des composants, la gestion des flux d'information et la rationalisation des décisions SI du Top management.

A cet effet, les entreprises ont adopté des solutions SI telles que les progiciels de gestion intégrés (PGI) en anglais « Enterprise Resources Planning » (ERP), l'infogérance ou la refonte personnalisée du SI. Les résultats n'étaient pas toujours positifs, entre des dépassements de coût et de délai et des décisions stratégiques non abouties au sujet des technologies de l'information. D'où l'apparition de la Gouvernance, Risque et Conformité des systèmes d'information (IT GRC).

La Gouvernance des SI (GSI) est un cadre organisationnel, ayant pour objectif l'alignement du SI à la stratégie de l'entreprise pour en faire un support et non pas seulement un applicatif de gestion. Issue de la gouvernance de l'entreprise, elle vise le pilotage du SI en production et s'assure de son adéquation avec les besoins des parties prenantes internes et externes. La Gouvernance SI a fait preuve d'efficacité en matière de retour sur l'investissement. Des entreprises ont minimisé leurs dépassements de budget IT, et ont pu bâtir des systèmes de pilotage et de prise de décision participatifs et transversaux.

En effet, il s'agit d'une discipline de mesure et de contrôle non seulement applicable à des projets de mise en place de nouveaux SI, mais permettant de cadrer les besoins et optimiser les services des systèmes existants. La gouvernance SI permet également à l'entreprise de répondre à des besoins métiers potentiels par des processus informatiques fiables et performants.

Plusieurs référentiels ont été mis en place à partir de la collecte des bonnes pratiques GSI et de leur généralisation tels que COBIT, ITIL, CMMI, ISO 27001, etc. Ces référentiels ont pour but de cadrer la discipline par le biais d'un ensemble de directives qui servent de repères aux processus de l'entreprise. Des mesures sont également définies pour les auditeurs afin d'évaluer la maturité des SI. Cependant, ses référentiels sont sous forme de documentation nécessitant une implémentation personnalisée de la part de l'entreprise. C'est la raison pour laquelle des plateformes de Gouvernance des SI doivent être utilisées pour mener à bien ce défi managérial.

C'est dans cette perspective que s'inscrit cette thèse de doctorat qui propose une plateforme intelligente de gouvernance des SI et qui représente la couche stratégique d'une plateforme générique de la Gouvernance Risque et Conformité des SI réalisée par l'équipe EAS. Cette solution implémente les bonnes pratiques GSI pour une entreprise de type et de taille divers.

Les atouts de cette solution détournent également les limitations que présentent les plateformes de Gouvernance des Systèmes d'Informations existantes.

Ce rapport de thèse comprend trois parties :

La première partie intitulée état de l'art, est composée de deux chapitres. Après l'introduction générale, nous présentons dans le premier chapitre, l'état de l'art de la gouvernance des systèmes d'information et la problématique de ce travail de recherche. Dans le deuxième chapitre, nous nous intéressons aux Workflows Inter-Organisations et aux Systèmes Multi-Agents.

La deuxième partie, consacrée à la solution proposée se compose de trois chapitres. Dans le troisième chapitre, nous détaillons l'architecture du Workflow de gouvernance SI. Dans le quatrième chapitre, nous proposons une ontologie de gouvernance SI, qui est le noyau du moteur sémantique de notre solution. Dans le cinquième chapitre nous exposons le système expert de médiation de notre solution.

La troisième partie est dédiée à la conception et la réalisation de notre plateforme de Gouvernance SI, ainsi qu'aux tests et à l'intégration à la plateforme générique.

Nous finirons ce mémoire de thèse par une conclusion générale, en faisant le bilan du travail réalisé, son originalité, ses limites et ses perspectives.

Partie I:
ETAT DE L'ART

CHAPITRE 1 :
GOUVERNANCE DES SYSTEMES
D'INFORMATION

I. Gouvernance des Systèmes d'Information

I.1. Introduction

La structure des technologies de l'information (TI) peut avoir un impact sur la performance des entreprises (Garrity, 1963). Ces technologies ont non seulement changé les méthodes traditionnelles par lesquelles les gens obtiennent des informations, mais aussi les schémas de gestion de la production et la structure de l'organisation des entreprises dans l'espace et dans le temps. Les TI jouent un rôle important dans les transactions inter-organisationnelles et les relations. Ainsi, elles sont devenues un atout précieux et une ressource de la stratégie de l'entreprise contemporaine (Dahlberg, 2007).

Les technologies de l'information sont devenues essentielles pour le support, la durabilité et la croissance de l'entreprise. L'utilisation excessive de la technologie a créé une dépendance critique de l'informatique à la gouvernance des TI (GTI) (Haes, 2008). Certaines études ont montré que les entreprises qui ont des modèles de bonnes GTI, génèrent plus de valeur TI que leurs concurrentes (Curry, 2004). Une GTI efficace est considérée comme un moyen essentiel pour assurer le retour sur investissements informatiques tout en améliorant la performance organisationnelle.

Dans le présent chapitre, nous donnons un aperçu général de la Gouvernance des systèmes d'information. Ensuite, nous étudions son implémentation dans l'entreprise, avant de mener une étude comparative des plateformes de gouvernances SI existantes. La dernière partie du chapitre sera consacrée à la problématique générale de cette thèse.

I.2. Gouvernance des Systèmes d'information

Aujourd'hui, la gouvernance est un concept qui peut être utilisé dans de nombreux contextes (Brown, 1997). Il y a différents types de gouvernance: la « Corporate » gouvernance, la gouvernance d'entreprise et la gouvernance informatique. Ces trois types de gouvernance sont corrélés et nous devrions les traiter comme « une Gouvernance d'ensemble » avec des dépendances entre elles et un ordre à suivre.

La Gouvernance des Technologies de l'information (GTI) ne peut exister isolément, mais doit être un sous-ensemble de la gouvernance d'entreprise et est aussi communément appelé le sous-ensemble de la gouvernance d'entreprise (Lunardi G. L., 2014). Nous concluons donc que la GTI est le niveau le plus bas des trois types de gouvernance, le plus spécifique et le plus ciblé.

Depuis que les termes technologie de l'information (TI) et système d'information (SI) sont fortement reliés, le manque de clarté sur le concept de GTI n'est pas surprenant étant donné que le SI est une discipline relativement nouvelle qui a émergé dans une variété de domaines d'activités, mais certainement qui n'est pas limité aux sciences exactes ou aux sciences sociales. De nombreuses

études continuent à se concentrer sur la définition de la GTI (Peterson, 2004). Nous citons à titre d'exemple :

Année	Définition et Référence
1994	Ensemble de décisions du locus de la responsabilité des fonctions informatiques. (<i>Brown and Magill, 1994</i>).
1996	Degré de l'autorité pour prendre des décisions définies et partage entre la direction et les gestionnaires de processus à la fois informatique et des organisations professionnelles en appliquant dans l'établissement des priorités et allouant des ressources informatiques. (<i>Lufman, 1996</i>)
1999	Modèle d'autorité pour les activités informatiques clés (<i>Sambamurthy and Zmud, 1999</i>)
2000	Capacité organisationnelle par le conseil, la haute direction et la DSI pour contrôler la formulation et la mise en œuvre de la stratégie informatique pour assurer la fusion de l'entreprise et l'informatique. (<i>Grembergen, 2000</i>)
2002	Description des processus globaux d'une entreprise pour le partage des droits de décision à propos de l'information et le suivi de la performance des investissements informatiques. (<i>Weil and Vitale, 2002</i>)
2003	Constitution de structures liées aux TI ou architectures (et les modèles d'autorité associés), mis en œuvre pour accomplir avec succès (impératif) les activités en réponse à un environnement d'entreprise et les impératifs stratégiques. (<i>Schwarz and Hirschheim, 2003</i>)
2004	Responsabilité du conseil d'administration et la direction générale. C'est une partie intégrante de la gouvernance d'entreprise qui se compose de la direction et de l'organisation des structures et des processus qui garantissent que l'organisation des TI soutient et étend les stratégies et les objectifs de l'organisation. (ITGI, 2004)
2004	Spécification des droits de décision et du niveau de responsabilité pour encourager un comportement souhaitable dans l'utilisation des TI. (<i>Weil and Ross, 2004</i>)
2005	Processus par lequel les décisions sont prises autour des investissements informatiques: Comment les décisions sont prises ?, qui les prend ?, qui est tenu responsable de leurs mise en place ?, et comment les résultats des décisions sont mesurés et surveillés ? (<i>Craig et al., 2005</i>)
2006	Alignement stratégique des TI avec les entreprises de telle sorte que la valeur maximale de l'entreprise est réalisée par le développement et le maintien d'un contrôle efficace de l'information et de la responsabilité, la gestion du rendement et la gestion des risques. (<i>Webb et al., 2006</i>)
2006	Préparation de la mise en œuvre de l'informatique liée aux décisions concernant les objectifs, les processus, les personnes et la technologie au niveau tactique ou stratégique. (<i>Simonsson and Ekstedt, 2006</i>)
2010	Processus qui assure l'utilisation efficace et efficiente de l'information pour permettre à une organisation d'atteindre ses objectifs. (<i>Gerrard, 2010</i>)

Tableau 1.1: Définitions de la Gouvernance des Systèmes d'Information

Etant donné que la GTI est une discipline née au sein de l'entreprise, il s'avère nécessaire de définir son périmètre opérationnel comme étant un outil qui apporte :

- Une meilleure prise de décision concernant l'ensemble du système d'information afin d'accroître son efficacité ;
- Une clarification des rôles des différents acteurs afin de créer des synergies ;

- Une meilleure définition des responsabilités des acteurs afin de faire prendre conscience des droits et des devoirs de chacun ;
- une meilleure connaissance des processus clés liés au système d'information afin de faire partager la compréhension de la complexité de leur mise en œuvre.

En effet, un SI a donc comme mission principale, le traitement des technologies de l'information de l'activité de l'entreprise et la conformité à son métier et à ses besoins.

Pour vérifier cet alignement du SI sur les exigences métiers et sur la stratégie de l'entreprise, il faut avoir recours à la Gouvernance SI. La Gouvernance SI décrit comment un SI est dirigé et contrôlé via la définition des démarches et des bons principes à mettre en œuvre pour la recherche de la performance et la réduction des coûts et des risques. Elle permet de répondre à des questions pertinentes qu'on se pose lors de l'évaluation des SI et de leurs mises en place ou mises à jour à savoir :

- Quelles relations doivent exister entre Direction générale (DG) et Direction de SI (DSI) ?
- Comment se répartissent les rôles entre les parties utilisant le SI ?
- Quels sont les processus clés de la DSI ?
- Comment s'assurer d'une utilisation efficace du SI ?
- Comment accroître la pérennité du SI ?
- Quelle cohérence entre l'architecture technique et fonctionnelle de l'entreprise ?

Il existe des preuves de l'effet positif de la bonne mise en œuvre de la GTI dans les organisations, par exemple:

- Avec l'aide de la GTI, les organisations peuvent augmenter leur retour sur investissements informatiques jusqu'à 40% (Weil, 2004) .
- Les entreprises qui obtiennent de bons résultats dans la GTI peuvent obtenir des rendements sur investissement informatique plus élevé que leurs concurrents; compte tenu de la même stratégie business, ceux avec une performance moyenne en GTI peuvent faire 20% de profit en plus.

En dépit de toutes ces preuves au sujet de l'influence des GTI dans le succès et le rendement des organisations, il y a aussi des preuves que l'informatique continue d'être mal gérée et régie:

- Les changements structurels de grande envergure suite à une mise en œuvre d'ERP peuvent être désastreux, comme le montre (Bingi, 1999) qui stipule qu'il y a plusieurs tentatives infructueuses ERP, et les entreprises ont perdu non seulement le capitale investi dans le progiciel et des millions versés à des consultants extérieurs, mais aussi une grande partie de leurs activités. Dans plusieurs implémentations ERP, le dépassement des coûts et du calendrier ainsi que le manque d'améliorations est élevé. (Exemple : le cas de la faillite FoxMeyer). En effet, 70% des implémentations ERP ne parviennent pas à atteindre leurs objectifs d'entreprise (Bernroider, 2008).

- Le retard de mise en œuvre d'une nouvelle version d'un produit a été frustrant parce qu'un grand nombre de ses projets ont pris plus de temps que ce qui a été prévu, suite à des contraintes technico-fonctionnelles ou de gestion.
- Les organisations doivent cesser de dépenser plus de 80% de leur budget informatique en entretien, correctifs, mises à niveau et autres dépenses courantes, et moins de 20% par rapport au développement de nouvelles applications et capacités.
- 72% des projets informatiques sont en retard, manquent de fonctionnalités, ou ne sont jamais livrés; Pour les 28% des projets qui réussissent, 45% dépassent le budget et 68% ont pris plus de temps que prévu.
- La mise en évidence du fait que de nombreuses entreprises dépensent environ 50% de tous leurs investissements de capitaux sur l'informatique (Lunardi G. L., 2009).

Avec des preuves de l'échec de nombreuses implémentations informatiques, la proposition de nouvelles approches pour la GTI est impérative. Actuellement, les solutions les plus adoptées sont les référentiels et les bonnes pratiques existantes sur le marché. En effet, il existe plusieurs cadres et meilleures pratiques pour soutenir les organisations dans la mise en œuvre de la GTI. Nous présentons les référentiels les plus répandus.

I.3. Référentiels de Gouvernance des systèmes d'information

Dans la littérature de la gouvernance des TI, nombreux cadres de référence sont présentés, chacun avec ses atouts, ses avantages et ses limites. Ainsi, chaque cadre de référence offre un niveau de détail spécifique à son domaine. L'un des soucis de la gouvernance des TI est de les faire coexister pour de meilleurs résultats, vu qu'ils présentent une complémentarité. Nous présentons dans l'ordre les principaux référentiels en matière de pilotage global, de pilotage des services, des projets et de la sécurité des TI.

Il existe plusieurs grandes familles de référentiels par domaine :

- Pour ce qui est de la gouvernance d'entreprise et la gouvernance SI stratégique : COSO, Balanced Scorecard (BSC) et COBIT ;
- Pour ce qui est du management de la sécurité et la gestion des risques: les Normes de la famille ISO27000 ;
- Pour ce qui est du management des services IT : ITIL ;

Il existe de nos jours un nombre important de référentiels spécialisés dans des domaines de pointes comme le montre le tableau ci-après :

REFERENTIEL	DOMAINE
CMMI	Développement de projet informatique
PMBOK	Gestion de projet
ISO 9001	Qualité et relation client
TOGAF	Architecture de l'entreprise
Prince 2	Gestion de Projet
eScam	Gestion de la relation client

Tableau 1.2: Référentiels de Gouvernance par domaine

Dans ce qui suit, nous présentons brièvement les référentiels les plus utilisés.

I.3.1. Committee of Sponsoring Organizations of the Treadway Commission

COSO (COmmittee of Sponsoring Organizations de la Treadway Commission) publié en 1992 comme la base du contrôle interne pour aider les entreprises à évaluer et à améliorer leurs systèmes. Le contrôle interne est un processus décrit comme la responsabilité mise en place afin d'atteindre les objectifs regroupés dans les domaines suivants:

- l'efficacité de l'exploitation;
- la fiabilité de l'information financière;
- le respect des lois et règlements.

En 2004, le COSO a publié un document de gestion des risques dans l'entreprise (Enterprise Risk Management ou ERM), qui étend le diamètre du contrôle interne en tenant compte du/de/des :

- Concept de portefeuille de risques;
- Structuration des quatre catégories d'objectifs (opérations, reporting, conformité et objectifs stratégiques) ;
- Niveau de risque pour les décisions stratégiques de l'entreprise;
- Événements qui influent sur les risques;
- Réponses aux risques ;
- Volume de l'information et de la communication;
- Rôles et les responsabilités des parties prenantes responsables de la sécurité.

En 2013, une mise à jour du COSO a eu lieu pour l'adaptation du système de contrôle interne aux actualités technologiques et managériales des entreprises , à savoir : les risques liés aux technologies de pointes (cybercriminalité, le cloud computing, etc.) ; l'externalisation des actifs ; le focus en gestion (le rôle des comités du conseil d'administration, de la gestion globale, et des risques de la conformité) ; les responsabilités du personnel à tous les niveaux de la hiérarchie ; et les exigences de déclaration en dehors des états financiers (développement durable, protection de l'environnement, qualité, etc.).

I.3.2. IT Balanced Scorecard

Balanced Scorecard (BSC) (Gibbons, 2015) est une performance pour expliquer la vision et la stratégie de l'entreprise, et les traduire en plans d'action. Cela donne un retour sur les processus internes, et sur les contraintes externes, dans une stratégie d'amélioration continue. Ses auteurs, Robert Kaplan et David Norton, le décrivent comme suit: "Le BSC permet des résultats financiers traditionnels, mais ces résultats mettent en lumière le passé, ce qui était normal à l'ère industrielle, avec des investissements à long terme et les relations client peu présentes. Ces éléments financiers sont insuffisants, cependant, pour contrôler les entreprises dans l'ère de l'information, qui devrait se fonder sur leur valeur future en investissant dans ses clients, les fournisseurs, les employés, les processus, la technologie et l'innovation. ". Le BSC offre un outil de gestion, organisé sur quatre aspects:

- Aspect financier : définir le bénéfice financier apporté aux actionnaires ;
- Aspect client : attentes du marché et des clients ;
- Aspect processus internes : les améliorations potentielles à effectuer sur les processus internes de l'entreprise, porteurs de valeur ;
- Aspect construction future : mobilisation des ressources humaines et des infrastructures en guise d'amélioration et d'apprentissage.

À l'heure actuelle, il est supposé que cette approche conduit à une bonne vision de la gouvernance d'entreprise.

Carte stratégique	Processus	Indicateur
Contribution et alignement	Contrôle des coûts Réduction des coûts / Automatisation Valeur adaptative Management de valeur	Budget informatique Benchmarks Performance de l'entreprise
Futur et anticipation	Gestion des compétences Sourcing / Achats Veille technologique Architecture technique Urbanisation	Performances Coûts Niveaux de services
Performances opérationnelles	Approvisionnements Conduite de projets Maintenance des applications Exploitation, Administration Support ...	Performances Benchmarks et tendances Coûts standards
Clients et utilisateurs	Niveaux de services (SLA) Conformité aux besoins Exigences réglementaires Respect du budget Niveau de demande	Qualité du service vs SL A Satisfaction des utilisateurs et clients Réclamations

Tableau 1.3: Ébauche du BSC appliqué à la GTI

I.3.3. Famille ISO 27000

Pour ce qui est du management de la sécurité, plusieurs normes, méthodes et référentiels de sécurité des systèmes d'information sont mis en place. Il s'agit des guides méthodologiques pour assurer une démarche de sécurité uniforme.

L'ISO a mis en œuvre la norme ISO/IEC 27000. Ce nombre correspond à une série de normes relatives à la sécurité, à savoir 27000, 27001, 27002 et 27006 qui sont publiées. Ces normes servent à obtenir une certification, ou sont des guides de bonnes pratiques:

- la norme ISO/IEC 27000 : le vocabulaire et les définitions du domaine de la sécurité, applicables à chacun des standards ;
- la norme ISO/IEC 27001 : la politique du management de la sécurité des systèmes d'information pour l'entreprise en guise de certification ;
- la norme ISO/IEC 27002 : guide de bonnes pratiques de la sécurité des SI ;
- la norme ISO/IEC 27003 : guide d'implémentation;
- la norme ISO/IEC 27005 exigences applicables pour la sécurisation de l'information, basée sur le PDCA (Plan, Do, Check, Act) , complétant l'ISO 27001.

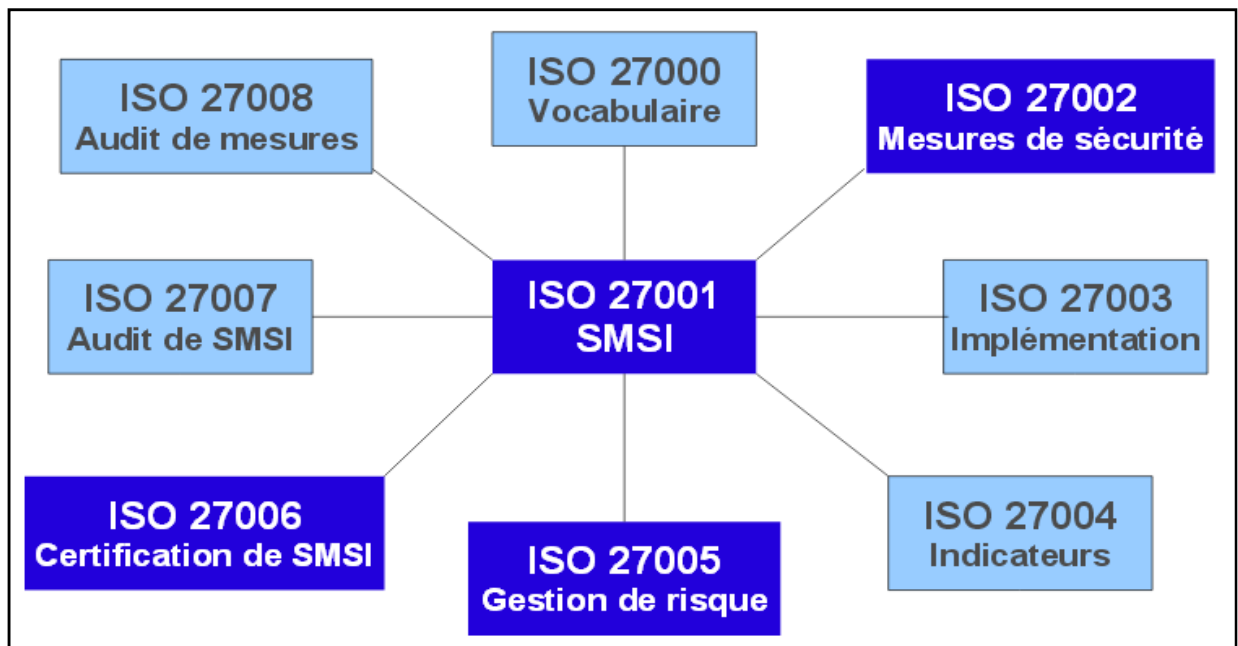


Figure 1: Normes de la famille ISO27

I.3.3.1. ISO 27001

Publié en Novembre 2005, ISO 27001 est une norme ayant pour objectif la mise en place d'un système de management de sécurité de l'information (SMSI). Elle a été développée pour «fournir un modèle de mise en œuvre, d'exploitation, de surveillance, et d'examen de l'établissement, afin de maintenir un système de gestion de la sécurité de l'information ».

ISO 27001 utilise une approche descendante fondée sur les risques qui est technologiquement neutre. Elle définit un processus de planification en six parties:

- Définir une politique de sécurité.

- Définir la portée du SMSI.
- Procéder à une évaluation des risques.
- Gérer les risques identifiés.
- Sélectionner les objectifs et les contrôles à mettre en œuvre.
- Préparer une déclaration d'applicabilité.

La documentation comprend des détails sur la responsabilité de la gestion, l'audit interne, l'amélioration continue, et des mesures correctives et préventives. La norme exige la coopération entre toutes les sections d'une organisation.

La norme ISO27001 ne prescrit pas de contrôles spécifiques de sécurité de l'information, mais fournit une liste de contrôle qui devrait être considérés dans le code d'accompagnement de la pratique. La certification ISO 27001 et elle donne un ensemble de recommandations à suivre obligatoirement.

I.3.3.2. ISO 27002

La norme ISO / IEC 27002, créée en 2005, décrit un ensemble complet d'objectifs de contrôle de la sécurité de l'information et un ensemble de bons contrôles de sécurité de la pratique généralement reconnus.

Sécurisation de l'information qui consiste à assurer la confidentialité, l'intégrité et la disponibilité.

Selon cette norme la sécurisation de l'information passe par quatre phases essentielles à savoir :

1. Périmètre à protéger (liste des biens sensibles).
2. Nature des menaces.
3. Impact sur le système d'information.
4. Mesures de protection à mettre en place.

Pour se faire, ISO 27002 traite douze sections principales:

- I.L'évaluation des risques ;
2. La politique de sécurité ;
3. L'organisation de la sécurité de l'information ;
4. La gestion d'actifs ;
5. La sécurité des ressources humaines ;
6. La sécurité physique et environnementale ;
7. La gestion des communications et des opérations ;
8. Les contrôles d'accès ;
9. L'acquisition du systèmes d'information, son développement et sa maintenance ;
10. La gestion des incidents de sécurité de l'information ;
11. La gestion de la continuité des activités ;
12. La conformité aux lois.

Ce standard n'est pas sujet de certification ; aucune obligation d'application n'est exigée mais propose un ensemble de propositions. Il ne traite pas les systèmes de management de la sécurisation de l'information ni l'approche PDCA.

I.3.4. IT Infrastructure Library - ITIL

Une étude lancée par le gouvernement britannique en 1990, pour définir les meilleures pratiques de gestion des services IT, a donné naissance à une bibliothèque nommée "IT Infrastructure Library" ou ITIL, documentant une approche de gestion des services informatiques, pour supporter les utilisateurs des organisations d'affaires.

En 2004, une version 2 avec 9 livres a été mise en place en mettant l'accent sur le lien entre la technologie et les affaires, et en se basant sur les processus nécessaires pour fournir les services adéquats aux clients de l'entreprise. ITIL fournit un cadre de bonnes pratiques pour guider la gestion des services informatiques; cette méthode est de plus en plus acceptée pour la gestion des services IT à travers le monde.

De nombreuses organisations à travers le monde adoptent ces bonnes pratiques, pour améliorer le contrôle des services IT externalisés par les entreprises. Son acceptation au sein des organisations est bien réputée vu que les sociétés atteignent des améliorations dans les résultats de leurs processus, en l'utilisant.

Le service ITIL est défini comme un moyen de fournir une valeur à un client, ce qui rend plus facile l'atteinte de résultats identiques, sans les coûts de propriété et des risques spécifiques.

En 2007, la version V3 du référentiel ITIL est basée sur cinq livres de bonnes pratiques, proposant des compléments par secteur ou par marché ainsi que des modèles génériques (cartes de processus, etc.) :

- **Service Stratégie** décrit la stratégie générale et l'apport de valeur des services, tout en traitant de l'alignement avec les métiers et la gouvernance des TI.
- **Service Design** propose des procédures, des architectures et des documents pour créer les processus de management des services.
- **Service Transition** propose des guides d'intégration pratique des processus de gestion des services entre métiers et opérations.
- **Service Opération** propose des guides pour réaliser les objectifs de qualité de service dans un souci d'efficacité et d'efficience.
- **Service Développement continu** propose des guides pour identifier et améliorer les processus. Il combine les méthodes du management de la qualité et la boucle d'amélioration PDCA.

Contrairement à la V2 de ITIL, V3 n'est pas sujet de certification, et elle propose une amélioration continue des services offerts aux clients de la DSI.

I.3.5. Control Objectives for Business Information and related Technology-COBIT

COBIT (Control Objectives for Business Information and related Technology), développé en 1994 (et publié en 1996) par l'ISACA (The Information System Audit and Control Association) est un outil de Gouvernance IT qui a été conçu dans un premier temps, pour le contrôle en déclinant les directives de COSO sur des objectifs des technologies de l'information.

En 2000, la publication de la troisième version de COBIT a eu lieu en proposant, parallèlement à un « guide d'audit », un « guide de management » . A l'apparition de la loi SOX en 2002 (Zhang, 2013), engageant les dirigeants, et garantissant la transparence des comptes et les systèmes d'alerte , COBIT a été conforme a ce nouveau contexte, et il a été reconnu par l'aspect conformité à partir de cette version et dans les versions à venir.

La version quatre a vu le jour en décembre 2005, puis la version 4.1 deux ans plus tard qui a été jugée stable en englobant en plus de l'aspect audit, contrôle interne et conformité, le management et la gouvernance des systèmes d'information.

COBIT dans sa version 4.1 est un référentiel de gouvernance des systèmes d'information qui décompose tout système informatique en 34 processus répartis en quatre domaines fonctionnels :

- Planification et organisation composé de 10 processus.
- Acquisition et mise en place composé de 7 processus.
- Fourniture du service et support composé de 13 processus.
- Surveillance composé de 4 processus.

Chaque processus met en œuvre des ressources informatiques (applications, informations, infrastructures et personnes), fournit une information destinée à satisfaire les besoins métiers exprimés sous formes de critères (efficacité, efficience, confidentialité, intégrité, disponibilité, conformité, fiabilité) et concerne un ou plusieurs domaines de la gouvernance des SI (alignement stratégique, apport de valeur, gestion des risques, gestion des ressources, mesure de la performance).

L'utilisation de COBIT permet de distinguer trois étapes :

- ✓ Définition des objectifs,
- ✓ Détermination et gestion des ressources,
- ✓ Gestion des processus.
- ✓ Contrôle et mesure de la performance

COBIT est constitué de plusieurs composants mis au service des parties prenantes internes et externes du SI. Ces composants sont reliés entre eux et visent à répondre aux besoins de gouvernance IT, de gestion, et de contrôle de différents acteurs, (voir figure ci-dessous) :

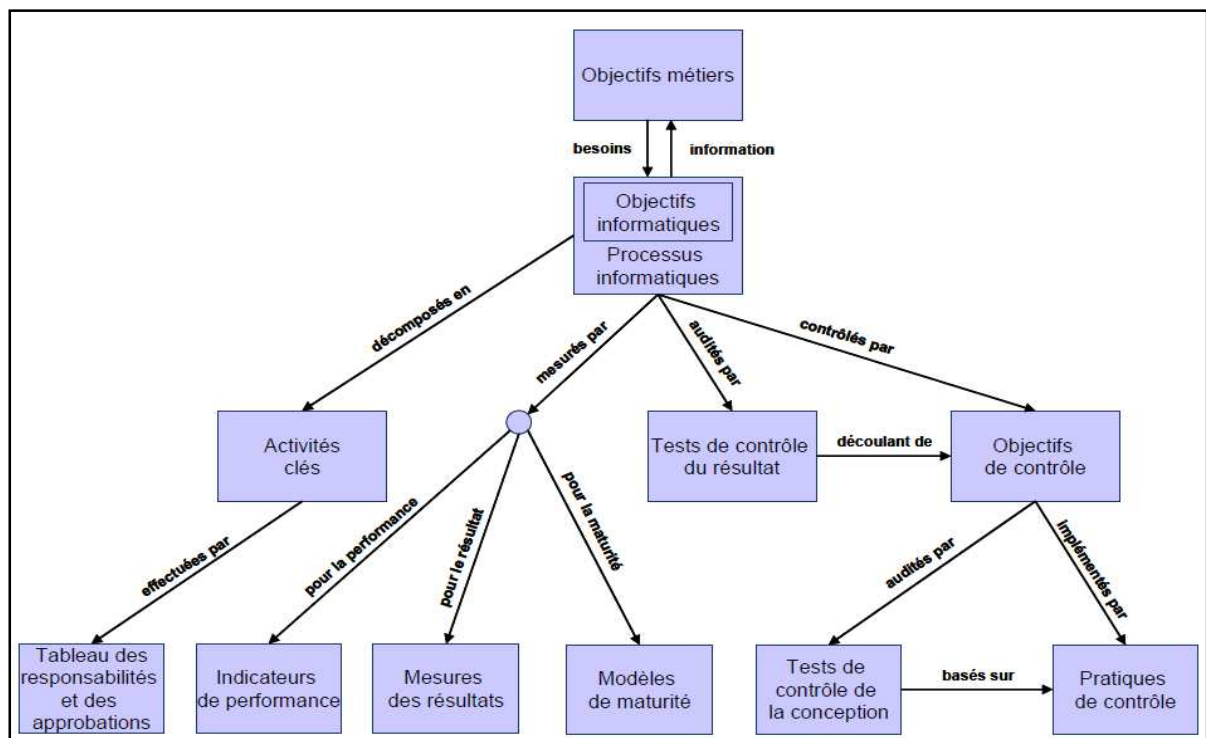


Figure 2: Lliaisons entre composants COBIT

L'apport majeur de COBIT est qu'il mobilise en plus de la Direction des Systèmes d'Information (DSI) les responsables métiers, et le conseil d'administration, en mettant l'informatique en soutien des objectifs métiers de l'entreprise (Chergui M. S., 2015). Pour se faire, COBIT déploie les cinq axes de la gouvernance SI, à savoir :

- L'alignement stratégique : les plans informatiques sont alignés aux plans métiers ;
- L'apport de valeur : la valeur ajoutée des TI pour le chiffre d'affaire de l'entreprise ;
- La gestion des ressources : l'optimisation des infrastructures, des connaissances et du capital humain ;
- La gestion des risques : la conscience des risques potentiels, de la conformité et leur traitement ;
- La mesure de la performance : la surveillance de la mise en œuvre de la stratégie.

En pratique, COBIT comme référentiel de gouvernance SI, permet à l'auditeur de diagnostiquer le SI par processus selon la démarche suivante :

- Choisir les processus en question ;
- Choisir l'interviewé ;
- Poser les questions à partir des objectifs de contrôle des processus ;
- Noter les réponses en pourcentage ;
- Tracer le graphe radar ;
- Décider les processus prioritaires à implémenter ;
- Visualiser les mesures et les indicateurs de performance.

L'avantage du référentiel COBIT - la raison pour laquelle il a été choisi dans le cadre de ce travail - est sa dimension fédératrice et son attitude complémentaire avec la majorité des référentiels précédemment cités. Dans ce qui suit, nous présentons ces deux aspects pour une vision complète des référentiels de la Gouvernance des Technologies de l'Information.

I.4. Complémentarité des référentiels de Gouvernance des Systèmes d'Information

I.4.1. Relations entre référentiels

COBIT a été conçu avec une vision stratégique et une vision de contrôle, liées toutes les deux à l'opérationnel par le biais des processus. En effet, c'est un référentiel basé sur les cadres établis, tels que Capability Maturity Model de Software Engineering Institute, ISO 9001, ITIL et ISO 17799 (cadre de sécurité standard, maintenant ISO 27001). En raison de sa couverture transversale des domaines de l'entreprise, COBIT peut agir comme un intégrateur qui englobe plusieurs pratiques dans un même cadre et contribue à les relier aux objectifs stratégiques de l'entreprise.

COBIT est destiné à être utilisé au niveau stratégique d'IT gouvernance. Il fournit un cadre global de gouvernance fondé sur un modèle de processus générique qui le rend applicable à la plupart des organisations. Les référentiels et les normes qui couvrent des domaines spécifiques de façon plus

détaillée, comme ITIL et ISO 27001, CMMI et PMBOK, peuvent être mis, en correspondance avec le cadre COBIT pour créer une hiérarchie de documents d'orientation.

I.4.1.1. COBIT-BSC

COBIT fournit une interface compréhensible par la direction générale par le biais de ses processus répartis selon les cadrans du pilotage stratégique du BSC (Zhang, 2013).

Le tableau ci dessous illustre les convergences des processus COBIT par rapport aux cadrans du BSC :

Cadrans BSC	IT Process
Contribution stratégique	PO1, PO5, PO9, PO10, SE4
Relation client	DS1, DS8, DS10,
Futur et anticipation	PO7, PO4, PO8, DS2, PO2,AI1, PO3
Excellence opérationnelle	DS13, DS12, AI6,

Tableau 1.4: Correspondance COBIT-BSC

Le tableau montre une répartition équitable des processus COBIT par rapport aux volets traités par les cadrans de BSC. Ce qui reflète sa capacité d'assurer la visibilité permanente en matière du pilotage stratégique et par rapport à une diversité d'acteurs de l'entreprise.

I.4.1.2. COBIT-ITIL

ITIL étant le référentiel type pour le management des services IT, aussi bien pour le support que pour la fourniture. Il s'intéresse aux parties prenantes et à leurs échanges à travers le SI, chose que COBIT aussi prend en charge du point de vue conformité des processus informatiques, en fonction des exigences réglementaires, en plus de la gestion des risques et du contrôle des fonctions informatiques.

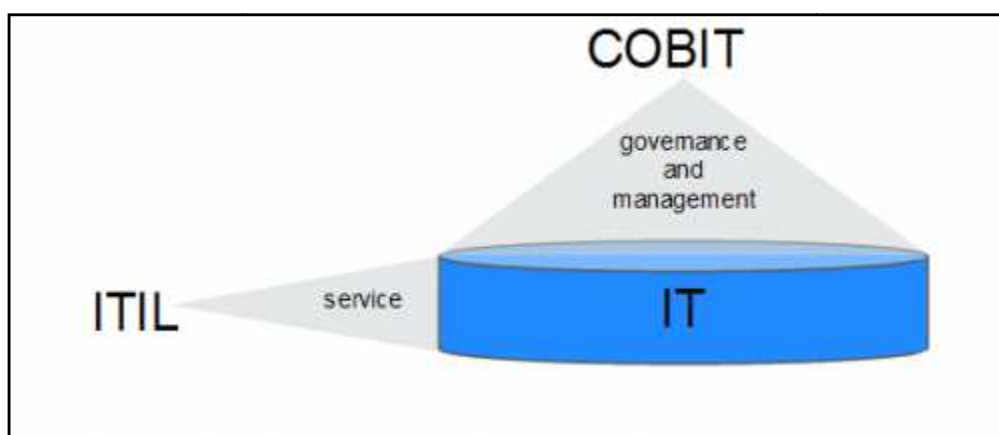


Figure 3: COBIT et ITIL

Par le biais de ce point commun, ITIL (ISACA, Aligning CobiT®4.1, ITIL®V3 and ISO/IEC 27002 for Business Benefit., Management breifing from ITGI and OGC) et COBIT peuvent être déployés

simultanément pour couvrir d'une façon complémentaire, les trois domaines de la gouvernance à savoir :

1. La conformité,
2. La performance,
3. La responsabilité,

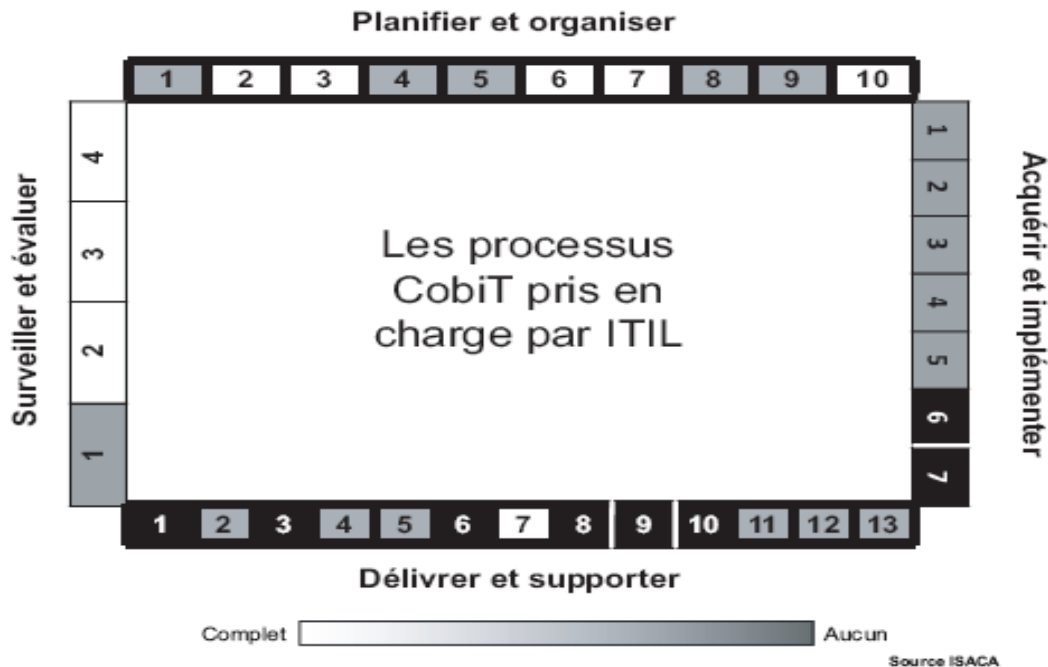


Figure 4 : Couverture des processus COBIT par ITIL

Les intéressés au détail du Mapping COBIT et ITIL doivent s'adresser à (ISACA, 2011). En plus de cette complémentarité, la différence entre ses deux référentiels GTI, réside en l'aspect global et générique de COBIT tout en répondant à quoi faire pour mieux gouverner le SI. ITIL quant à lui il répond au comment avec une focalisation sur le volet organisation des centres de services et de gestion des éventuels incidents.

I.4.1.3. COBIT /ISO 27002/ ISO 27001 / ISO 9001

COBIT s'intéresse aussi à la sécurité au sein du système d'information. En effet, via COBIT les experts ont transformé la sécurité, d'un souci de techniciens à une variante dans la stratégie de la DSI en mettant l'accent d'autant plus sur la sécurisation de l'information. Quatre processus sont dédiés à ce propos :

Processus IT	Description
PO6	Faire connaître les buts et orientations du management
PO9	Évaluer et gérer les risques
DS4	Assurer un service continu
DS5	Assurer la sécurité des systèmes

Tableau 1.5: Processus COBIT de sécurité de l'information

C'est la raison pour la quelle, l'ITGI a mis en place une correspondance entre les 34 processus de COBIT dans sa version 4.1 avec les 133 mesures proposées par la norme ISO27002. Les intéressés trouveront dans (Zhang, 2013) le détail de cette correspondance, mais en résumé ISO 27002

complète le cadre général de gouvernance qu'est COBIT par la description des mesures pour sécuriser l'information au sein de l'entreprise.

Quant à la norme ISO 27001 qui s'appuie éventuellement sur ISO 27002 pour mettre en place un système de management de la sécurité de l'information (SMSI) (Guechtouli, 2014), reprend les axes principaux de la norme qualité ISO 9001 pour la mise en place d'un système de management de la qualité (SMQ). Les processus cités précédemment sont aussi déclencheurs de la mise en place d'un tel système. La matérialisation par le biais d'un certificat est indispensable aussi bien pour le SMSI ou SMQ. COBIT, par rapport à ces normes permet un diagnostic essentiel en amont pour définir le périmètre d'application de l'une ou de l'autre (les activités réalisées pour ISO 9001 et les actifs à protéger pour ISO 27001).

I.4.1.4. COBIT /PMBOK/ CMMI

Il existe de nombreux référentiels de gestion de projet ou ce que nous appelons référentiels de processus en liaison avec le SI.

Le Capability Maturity Model (CMM) (Ehsan, 2010) a été développé par le Software Engineering Institute (SEI) de Carnegie Mellon University et décrit les principes et les pratiques sous-jacentes des processus de maturité de développement de logiciels. Le cadre a été conçu pour aider les organisations de développement de logiciels à améliorer leurs processus en suivant un chemin ad hoc évolutif. Une série de modèles développés par le SEI, y compris CMM Software, l'ingénierie des systèmes CMM et le développement intégré des produits CMM ont été récemment fusionnés et étendus dans l'intégration CMM connu sous le nom « CMMI » (Team Produit CMMI, 2002). Il fournit deux points de vue de la capacité: une vue mise en scène et une vue continue. La vue mise en scène, fournit cinq niveaux d'évolution vers la maturité organisationnelle (initiale, gérée, définie, gérée quantitativement et optimisation). La vue continue comprend six niveaux de capacité de traitement (incomplet, exécuté, géré, défini, quantitativement géré et optimisation) (Team Produit CMMI, 2002). Le point commun avec COBIT est visiblement la notion de modèle de maturité dans l'un et dans l'autre, mais il n'y a pas lieu de confondre ou de concilier l'un avec l'autre, COBIT est un référentiel GTI alors que CMMI est un cadre de gestion de projet. Cependant le déploiement de CMMI sans COBIT présente un nombre important de difficultés ou de dysfonctionnement à savoir :

- La conduite de changement dans les équipes projet ;
- Les systèmes de mesure.

Inversement, le déploiement de CMMI dans le cadre d'une stratégie SI basée sur COBIT est intuitivement réussi dans la mesure où le terrain est mieux préparé pour mettre en place les processus CMMI.

PMBOK a été créé par le PMI (Project Management Institute), pour assurer un ensemble de principes de connaissances en gestion de projet. Le but est de guider un gestionnaire de projet pour mener à bien un projet (PMI, 2008). C'est un cadre détaillé de neuf domaines de connaissances, ventilées dans des activités à travers cinq étapes ou groupes de processus du cycle de vie du projet, qui sont réclamés pour englober la somme des connaissances généralement reconnue comme une bonne pratique dans la profession de gestion de projet. En plus de ces domaines de connaissances détaillées, des outils et des techniques, PMBOK note également que la gestion efficace du projet nécessite une

compréhension de la zone d'application, l'environnement du projet, la connaissance de la direction générale, les compétences, et les relations interpersonnelles.

L'objectif principal de la cartographie COBIT/ PMBOK (proposé par ITGI (Broussard, 2007)) est de fournir des conseils aux propriétaires de processus informatiques en identifiant les processus COBIT pour lesquels PMBOK fournit des indications plus détaillées et de mettre en évidence les domaines de PMBOK qui devraient être considérés pour chaque processus. Par conséquent, la cartographie met en évidence la façon dont les processus PMBOK peuvent soutenir COBIT. Un objectif secondaire de la cartographie est de fournir des conseils à ceux qui utilisent PMBOK comme base pour les pratiques de gestion de projet, quant aux domaines de COBIT ils devraient envisager l'application des pratiques aux projets impliquant les technologies de l'information. La cartographie identifie également les objectifs de contrôle COBIT qui doivent être appliqués au cours de divers processus PMBOK.

En conclusion, COBIT se présente comme un cadre fédérateur contrairement aux autres référentiels qui se spécialisent dans un volet bien restreint. Sa correspondance avec les autres référentiels est conceptuellement définie et documentée tout en gardant cette logique de référentiel global de la DSI.

I.4.2. Aspect orienté processus

Les informaticiens effectuent diverses tâches d'amplitudes de coûts et de durées différentes. Pour une bonne gouvernance du SI, la DSI doit gérer chaque tâche tout en la situant dans son contexte (projet ou fonction) c'est la raison pour laquelle il faut prendre les processus IT comme points de référence.

Exemples:

- Acheter ou déployer un ERP ;
- Mettre à niveau l'infrastructure réseau ;
- Structurer la sauvegarde informatique ;
- Faire appel à un nouveau fournisseur IT ;
- Sécuriser les serveurs ;
- Assurer la protection du système ;
- Résoudre les problèmes des utilisateurs.

Un processus IT est défini comme étant un ensemble structuré d'activités qui se déclenche par un événement spécifique, en générant des résultats propres à des clients ou des parties prenantes et qui peuvent éventuellement être mesurés (Biazzo, 2003).

L'aspect orienté processus dans la quasi-totalité des référentiels SI n'est pas aléatoire, il est plutôt l'élément pivot de transformation des éléments entrants en éléments sortant au sein de l'organisation. D'autant plus, le processus IT est mesurable en efficacité et en efficience tout en détaillant les activités à mettre en place, les responsabilités, les contrôles. A ce niveau, chaque référentiel se focalise sur une ou plusieurs notions qu'il trouve plus prioritaires à définir.

D'autant plus, nous parlons actuellement de l'entreprise orientée processus ou le management par processus qui se définit comme le résultat d'un travail d'équipe permettant d'identifier, de partager, de clarifier et d'améliorer les pratiques créatrices de valeur pour les clients, les actionnaires, le

personnel, la collectivité et les partenaires d'une organisation (Remus, 2000). Ce type de management permet de/d' :

- Comprendre et satisfaire les exigences ;
- Considérer les processus en termes de valeur ajoutée ;
- Mesurer la performance et l'efficacité des processus ;
- Améliorer en permanence les processus sur la base de mesures fréquentielles et objectives ;
- Favoriser la communication entre les différents représentants
- Faciliter le choix des orientations à prendre par le Top Management.

Nous constatons que la notion de processus en général, processus IT en particulier est dynamique et programmable ayant des entrées et des sorties avec des acteurs potentiels, d'où nous vient l'idée de l'informatisation des bonnes pratiques de la gouvernance SI par l'automatisation des processus IT.

I.5. Implémentation de Gouvernance SI dans l'entreprise

Nous avons abordé dans les sections précédentes les atouts des référentiels de la gouvernance des systèmes d'information ainsi que les relations entre eux. Cependant, en pratique les entreprises préfèrent toujours dresser leurs propres référentiels (Broussard, 2007) : étant donné que les cadres existants et les meilleures pratiques ne font aucune référence aux facteurs possibles qui peuvent influencer chaque mise en œuvre GTI et la façon dont les lignes directrices fournies doivent être utilisées (donnant l'environnement des différentes organisations, les organisations élaborent leur propre cadre afin de l'adapter à leurs besoins spécifiques.

La mise en œuvre GTI est influencée par des facteurs externes et internes (Xue, Liang, & Boulton, 2006), cependant, les cadres et les meilleures pratiques de la littérature ne parviennent pas à révéler une identification claire et concise de ces facteurs.

Après avoir traité la revue de la littérature, les facteurs de dépendance identifiés comprennent: la culture organisationnelle et de la structure, la stratégie, la taille, les différences régionales, l'industrie, la maturité, l'éthique et la confiance. Un résumé des facteurs de dépendance identifiés ainsi que leurs références de la littérature par rapport à la Gouvernance des SI est présente dans le tableau ci – dessous.

Facteurs d'implémentation ITG	Définition	Référence
Culture de l'organisation	Ensemble de valeurs, règles, façon de faire face aux problèmes liés à l'amélioration de la performance de l'entreprise (Gerrard, 2009), et influence le succès de la mise en œuvre ITG (Fink et Ploder, 2008)	Brown and Grant, 2005; Fink and Ploder, 2008; Gerrard, 2009; Jiandong and Hongjun, 2010; Maidin and Arshad, 2010; Symons, 2005; Weisinger and Trauth, 2003
Structure de l'organisation	Relations hiérarchiques et fonctionnelles entre les collaborateurs, c'est aussi la façon avec laquelle l'information est échangée entre eux.	Adams, Larson and Xia, 2008; Aagesen, van Veenstra, Janssen and Krogstie, 2011; Cochran, 2010; De Haes and Grembergen, 2008; Bernroider, 2008; Gao, Chen and Fang, 2009; Lunardi,

		Becker and Macada, 2009; Park, Jung, Lee and Jang, 2006; Shpilberg, Berez, Puryear and Shah, 2007; Symons, 2005; Webb, Pollard and Ridley, 2006
Taille	Effectif des collaborateurs et envergure des sites et d'activités	Brown and Grant, 2005; Cochran, 2010; De Haes and Grembergen, 2008; Jacobson, 2009; Lunardi, Becker and Macada, 2009
Industrie	Activité de l'entreprise	Brown and Grant, 2005; De Haes and Grembergen, 2008; Gerrard, 2009; Jacobson, 2009; Jiandong and Hongjun, 2010; Simonsson, Lagerström and Johnson, 2008; Tanriverdi, 2006
Différences régionales	Répartition géographique et impact du segment du marché que l'entreprise occupe	agesen, van Veenstra, Janssen and Krogstie, 2011; Fink and Ploder, 2008; Bernroider, 2008; Shpilberg, Berez, Puryear and Shah, 2007; Weisinger and Trauth, 2003
Maturité	Niveau d'organisation et de conscience des collaborateurs par rapport aux nouvelles variantes techniques et technologiques utiles pour l'activité de l'entreprise.	Cochran, 2010; Dahlberg and Lahdelma, 2007; De Haes and Grembergen, 2008; Park, Jung, Lee and Jang, 2006; Simonsson, Johnson and Ekstedt, 2008; Simonsson, Lagerström and Johnson, 2008
Stratégie	Dans ce contexte c'est plutôt l'alignement des TI avec le business pour la création de valeur	Brown and Grant, 2005; Dahlberg and Lahdelma, 2007; De Haes and Grembergen, 2008; Jacobson, 2009; Park, Jung, Lee and Jang, 2006; Symons, 2005
Ethique	La valeur au-delà du bénéfice, la confiance, la conscience des employés, l'esprit d'équipe.	Maidin and Arshad, 2010; Memiyanty and Putera, 2010
Confiance	Echange sûr et positif de l'information entre collaborateurs.	Memiyanty and Putera, 2010

Tableau 1.6: facteurs de dépendance et importance en GTI

En plus des facteurs de dépendance impactant l'implémentation de la GTI, plusieurs directives sont recommandées aux entreprises pour augmenter les chances de réussir la bonne gouvernance du système d'information à savoir : la vision, le positionnement, les outils et structures, l'évaluation et la comparaison.

I.5.1. Vision

Les entreprises qui cherchent à offrir des performances d'affaires plus élevées en exploitant les TI ont mis l'accent sur l'alignement. De nombreux chercheurs ont étudié le problème de l'alignement de la stratégie et actuellement de nombreuses entreprises reconnaissent également que l'informatique et les priorités d'affaires doivent être étroitement liées pour créer une valeur durable pour les organisations.

Quand il y a un alignement, les TI fournissent des services stratégiques cruciaux pour l'exploitation, et pour les besoins des utilisateurs.

Dans la littérature récente, la stratégie est incluse dans plusieurs cadres proposés par les chercheurs, par exemple (Mintzberg, 1979), l'alignement stratégique est identifié comme un enjeu important pour la GTI. La question qui se pose pour la vision est : Quelle démarche adoptée pour une meilleure GTI?

I.5.2. Positionnement

La GTI peut être déployé à l'aide d'un mélange de différentes structures, de processus et de mécanismes relationnels. Un cadre mature de la GTI devrait être fondé sur trois éléments principaux: les structures, les processus et la communication (Finin & Fritzson, 1994).

Avec des structures GTI, processus et mécanismes relationnels, les organisations visent à garantir que l'informatique apporte de la valeur à l'entreprise d'une manière transparente, que les responsabilités sont organisées pour soutenir la prestation maximale de la valeur commerciale des TI, et que les risques sont atténués en fonction des besoins d'affaires. Les mécanismes relationnels sont essentiels dans un cadre GTI et primordial pour atteindre et maintenir l'alignement des TI de l'entreprise, même lorsque les structures et les processus appropriés sont déjà mis en place. Ceci permet autrement de se situer sur l'échelle de maturité en GTI.

I.5.3. Outils et Structures

D'autant plus, la GTI est composée de processus avec les entrées, les sorties, les rôles et les responsabilités inhérentes à une définition de processus. En fait, la gestion efficace des ressources informatiques de l'organisation, permettant la mise à disposition pour atteindre ses objectifs, se fait à travers un ensemble de processus IT, terme que nous définissons en détail plus tard. Avec l'alignement de la stratégie SI et la stratégie d'entreprise, il est important que l'organisation conçoive des processus de GTI qui doivent être étroitement accordés à ceux de sa gouvernance d'entreprise.

La mise en œuvre et la gestion des processus informatiques sont des dimensions de GTI dans lesquelles les décisions informatiques sont prises et exécutées par le biais d'un ensemble d'actions qui doivent être présentes et gérées dans toutes les fonctions TI. Ceci pour convertir les capacités informatiques en applications et services commerciaux. Autrement dit, il faut se poser la question sur ce dont nous avons besoin pour une meilleure gouvernance du SI, en collectant un maximum d'informations sur les facteurs de dépendance tout en les analysant par rapport aux processus et objectifs IT ou TI et par rapport à la stratégie IT ou TI.

I.5.4. Evaluation et comparaison

Dans une organisation, l'alignement des TI avec le métier devrait être mesuré ainsi que les procédures qui doivent être établies pour la maintenance et l'amélioration (Guizani, 2016). Les implications de la décision devraient également être suivies et mesurées à partir du moment où la façon dont les résultats des décisions sont surveillés fait partie de la GTI.

Non seulement alignement TI/ métier, mais la maintenance aussi doit être mesurée, évaluée et contrôlée afin de mettre en œuvre la GTI. La question qui se pose à ce niveau là est comment se

rendre compte qu'on nous avons atteint le niveau sollicité ? Autrement dit, une quantification est indispensable par exemple le BSC des TI et de l'entreprise.

I.5.5. Référentiels et bonnes pratiques

Les cadres cités dans la section précédente ont été élaborés et proposés pour décrire les objectifs, les processus et les aspects organisationnels du management et de contrôle des TI. Ils sont créés et sont mis à profit dans la pratique (Bordini, Dix, & Seghrouchni, 2005). Bien qu'il n'y a pas de cadre GTI unique, complet, il y a un certain nombre de cadres disponibles qui peuvent être utiles pour l'élaboration d'un modèle de gouvernance, mais jamais du sur mesure pour l'organisation. La question qui se pose à ce niveau là est la conformité de la stratégie IT de l'entreprise avec les référentielles et bonnes pratiques existantes dans le marché actuel, et surtout à quel point la DSI et la direction générale sont elles conscientes de leur importance ? Il s'agit d'une question sur laquelle nous nous sommes beaucoup attardés dans ce travail de recherche afin de proposer un modèle adéquat.

I.5.6. Evolution et innovation

Les technologies de l'information jouent désormais un rôle très important dans l'agilité de l'entreprise, ce qui permet l'innovation rapide et continue des affaires dans les produits, les services, les canaux et la gestion de la chaîne d'approvisionnement. La technologie est en constante évolution et la complexité augmente avec le changement. Aujourd'hui, l'agilité présente un défi pour beaucoup de système d'information. Les entreprises investissent massivement sur les plateformes numériques pour soutenir les innovations dans leurs «écosystèmes» qui sont, leurs partenaires d'affaires avec les clients, les fournisseurs et d'autres entreprises spécialisées (Rousset, Herrmann, & Lang, 2014).

Comme cité précédemment, de nombreuses organisations dépensent 80% de leur budget informatique dans la maintenance, les mises à jour et moins de 20% sur les nouvelles applications et les capacités. Cependant, pour que la mise en œuvre de la GTI soit efficace, les organisations doivent concevoir continuellement la gouvernance, la transformation et le positionnement de l'information pour répondre aux futurs défis commerciaux. Les décisions concernant les innovations commerciales exigent des niveaux significatifs de collaboration et de partenariat entre l'informatique et les dirigeants de l'entreprise. De nombreuses organisations ne savent toujours pas quelles mesures doivent-elles effectuer d'abord pour la mise en œuvre de la GTI. A ce niveau, il faut savoir préserver le niveau de maturité GTI tout en se projetant dans le futur.

Afin de concrétiser les directives de l'implémentation de la GTI, nous avons mené, dans le cadre de cette thèse, une étude empirique sur l'utilisation des TI, pour la prise de décision stratégique, dans les entreprises marocaines. Cette étude fait l'objet de la section suivante.

I.6. Étude empirique: spécificités des SI au Maroc pour une meilleure gouvernance de l'IT

Nous avons mené une étude empirique sur la perception de l'utilisation des systèmes d'information pour la prise de décision stratégique. L'étude concerne 51 entreprises marocaines (Chergui M. ,

Nahla, Chakir, Elhasnaoui, Sekhara, & Medromi, 2016) utilisant un SI et appartenant à des secteurs d'activités différents, comme indiqué dans la figure 5.

L'étude a montré que l'utilité du SI pour plus de 70% des entreprises cibles, est la gestion des activités métiers régulières. 5% des SI s'intéressent à la gestion des risques, 15% traite la maîtrise des règles de gestion ou la réduction des coûts des investissements. Cependant, le marché mondial exige du SI d'autres fonctionnalités pour un meilleur alignement stratégique afin de créer de la valeur.

Pour plus de détail, nous avons étudié respectivement les forces et les faiblesses d'un SI par rapport à la gestion des processus métier de l'entreprise. Par la suite, nous avons analysé son habilité à remonter des informations pour prendre des décisions.

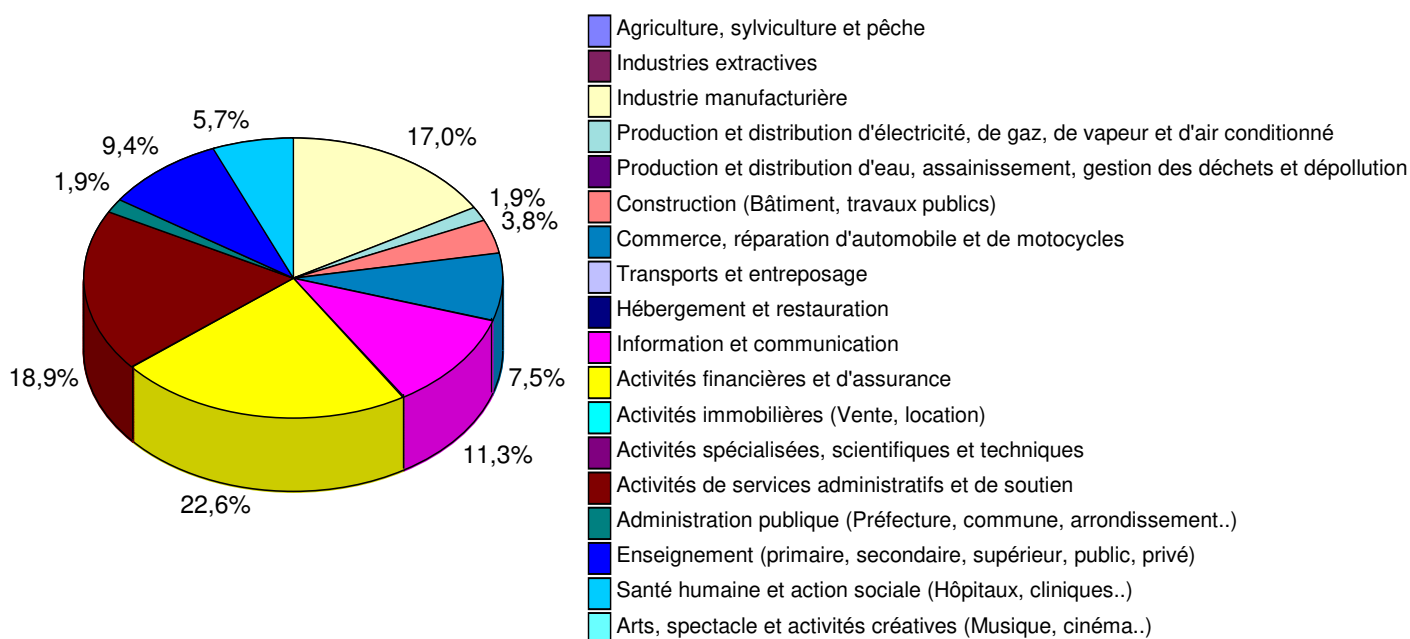


Figure 5: secteur d'activité des entreprises de l'étude

En ce qui concerne la gestion des processus, la plupart des SI gèrent partiellement les processus métier (45,1%). Les non réponses (17,6%) peuvent être interprété par la confidentialité de la réponse pour certain, ou la non efficacité de la gestion proposée (Chergui M. , Nahla, Chakir, Elhasnaoui, Sekhara, & Medromi, 2016).

Nous avons croisé ensuite, cette variable de gestion des processus via SI avec sa force estimée par les utilisateurs potentiels. Nous avons obtenu les résultats suivant :

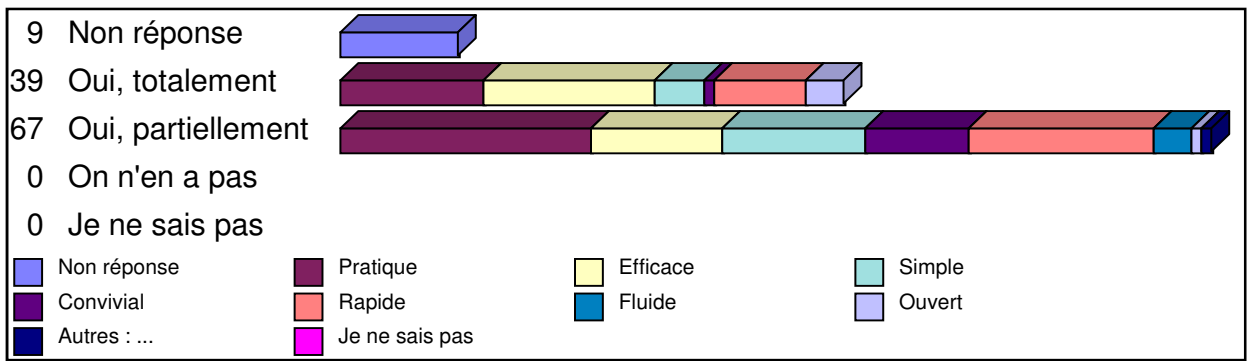


Figure 6: Gestion des processus et force du SI

Nous avons testé ensuite, la relation entre le traitement avancé des processus métiers et la pertinence du SI en termes de prise de décision par l'utilisateur. Nous avons déduit que la couverture des processus métiers par le SI et sa pertinence en termes de prise de décision sont deux variables fortement corrélées pour les entreprises marocaines. Nous pouvons dire que plus les processus métiers sont informatisés de bout en bout, plus l'information pertinente est remontée pour prendre des décisions stratégiques.

Pour vérifier ce constat, nous avons évalué d'abord la force du SI par rapport à sa pertinence et sa couverture des processus métiers :

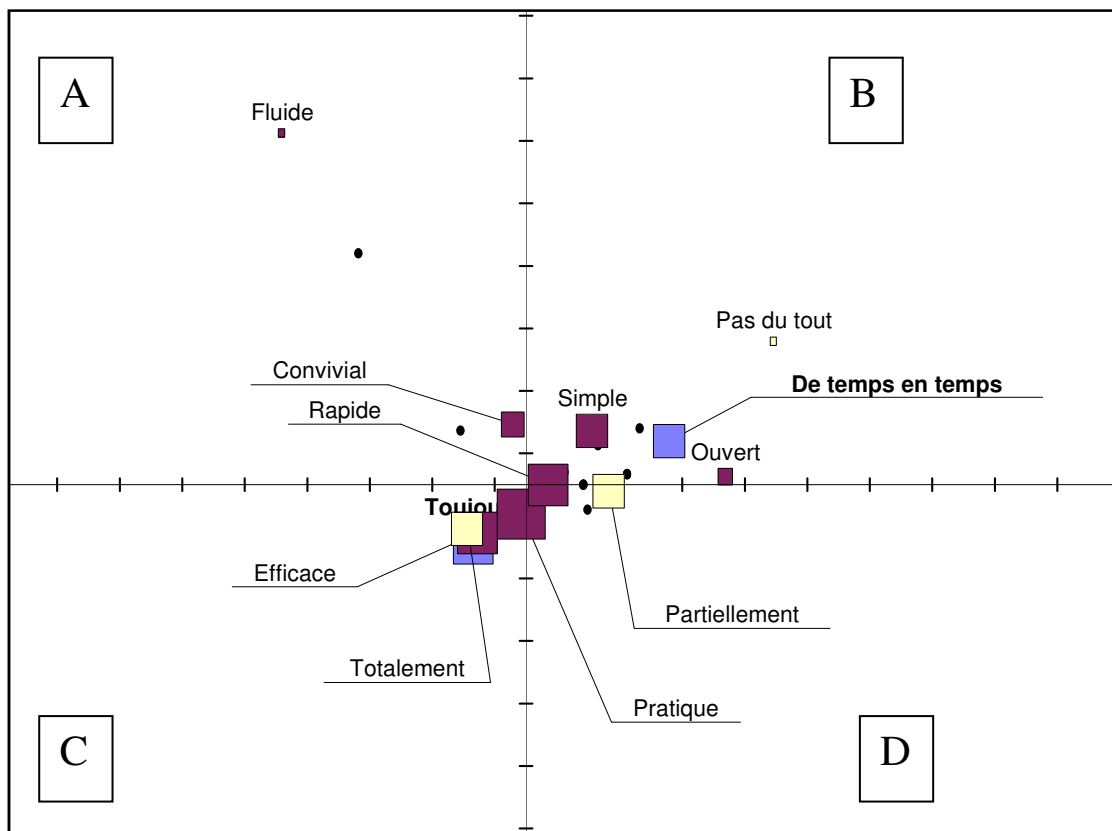


Figure 7: la force du SI par rapport à sa pertinence et sa couverture des processus métiers

Le SI qui gère totalement les processus métiers de bout en bout (cadran C), est efficace et pratique, alors qu'un SI qui les gère partiellement ne l'est pas. D'autant plus que les qualités d'efficacité, de pratique et de rapidité sont fortement corrélées quelque soit le système en question (cadran D). Nous constatons aussi que le fait qu'un SI permet de remonter des informations pertinentes est

positivement corrélé avec la gestion totale des processus métier et que dans la majorité des cas, il manque de simplicité, de convivialité et de rapidité.

De la même façon, nous avons évalué les faiblesses du SI par rapport à sa pertinence et sa couverture des processus métiers.

Nous remarquons d'après la figure ci-dessous au (cadran **A**), qu'un SI qui gère totalement les processus métier et qui permet de prendre des décisions est plutôt compliqué et lent. D'autres problèmes se posent (cadran **D**), à savoir, la limitation et la saturation qui sont tout les deux liées et inversement corrélées à la complexité et la lenteur.

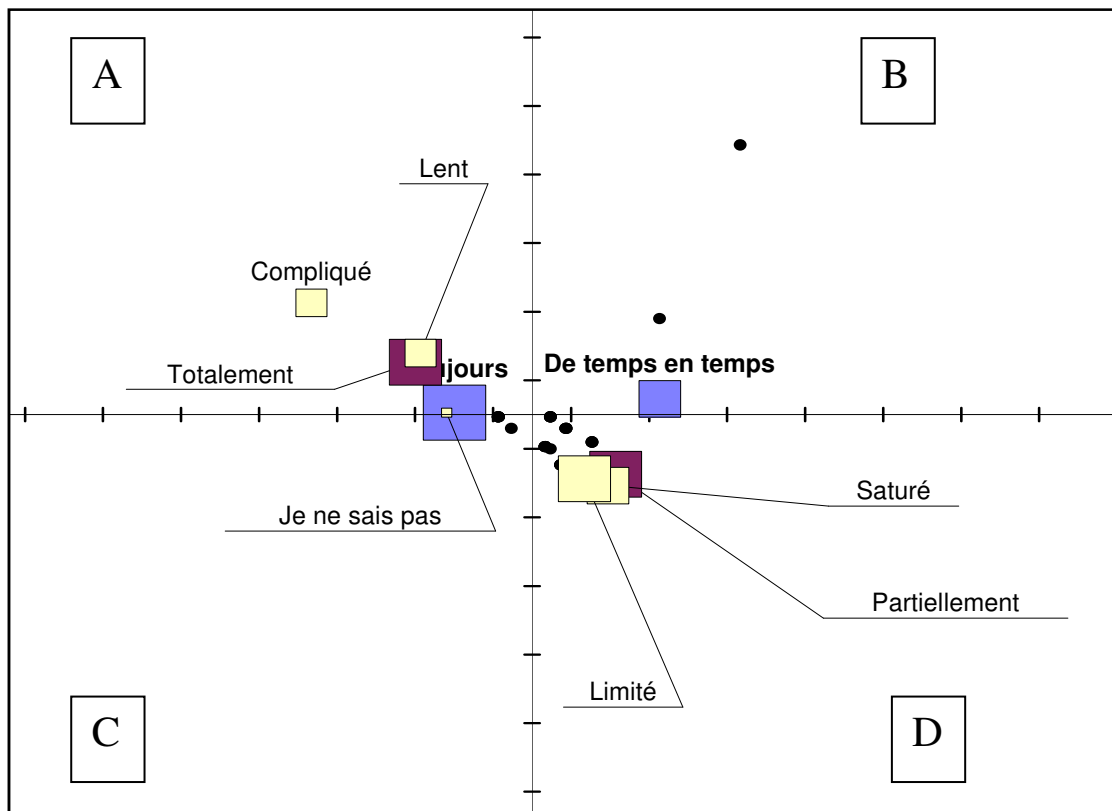


Figure 8: les faiblesses du SI par rapport à sa pertinence et sa couverture des processus métiers

Nous déduisons donc qu'un SI couvrant la totalité des processus métiers de l'entreprise et permettant de prendre des décisions pertinentes est obligatoirement efficace et pratique, il manque pourtant de convivialité, de rapidité et de simplicité d'utilisation au niveau stratégique.

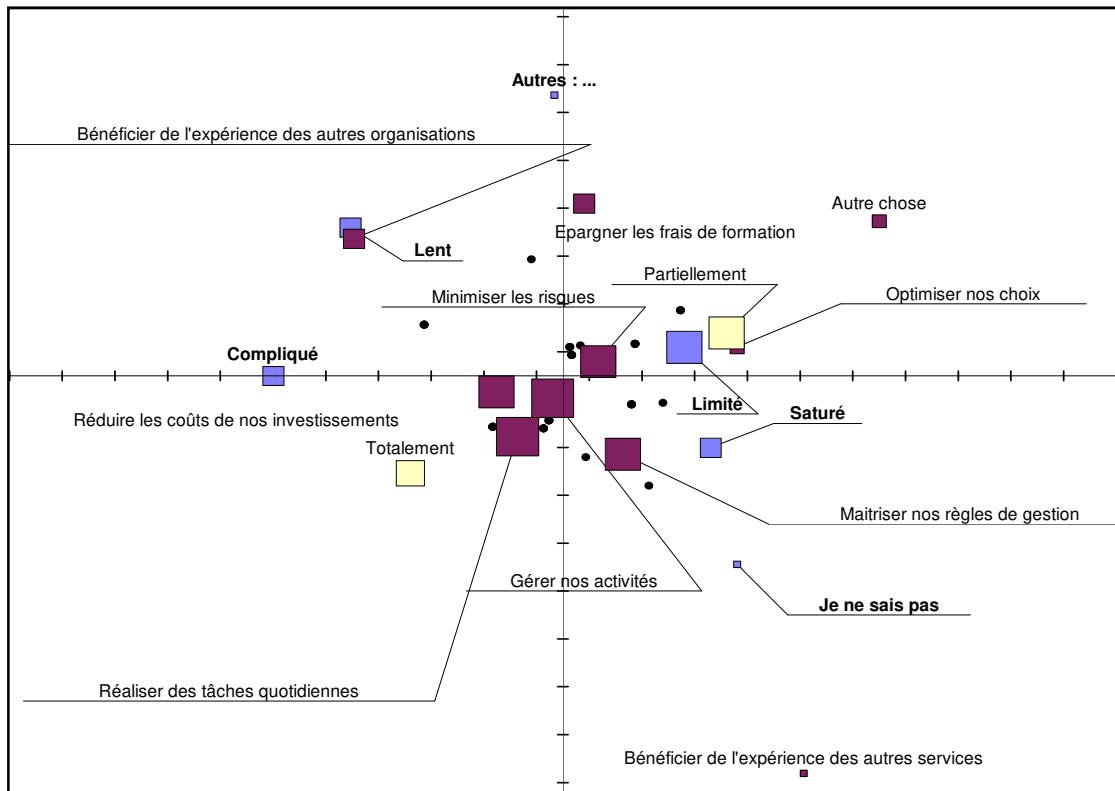


Figure 9: Analyse factorielle de la gestion des processus métiers

Selon l'analyse factorielle présentée ci-dessus, la gestion des processus métiers de bout en bout par le SI est corrélée avec la gestion des activités métiers et la réalisation des tâches quotidiennes. Par contre, elle n'est pas corrélée avec les autres fonctionnalités en relation avec l'alignement stratégique. En plus, le SI, s'avère plutôt compliqué et limité dans le cas d'une gestion partiel.

Ceci montre la limitation des horizons de prises de décision via le SI de l'entreprise marocaine et la nécessité d'une meilleure implémentation de sa gouvernance pour des résultats plus importants.

Cette étude a permis de soulever les limites des SI existants, d'un point de vue stratégique, et de mettre l'accent sur l'importance d'un modèle concret de gouvernance qui couvre les points suivant:

- Évaluation de tous les processus métiers de l'entreprise,
- Compréhension des contraintes de toutes les parties prenantes sur les processus et les objectifs (ELHASNAOUI, CHERGUI, CHAKIR, SEKHARA, Nahla, & Medromi, 2016),
- Mesure de la performance de l'exécution du processus,
- Simplification de la lecture des résultats directs et déduits,
- Adaptation aux contraintes de l'entreprise et aux changements environnementaux (Nahla, Medromi, Chergui, & Elhassnaoui, 2016).

Le tableau ci-après montre la projection de ces fonctionnalités sur les acteurs potentiels et les outils à déployer.

Fonctionnalités du modèle	Acteurs potentiels	Outils à mettre en place
Évaluer tous les processus métiers de l'entreprise	<ul style="list-style-type: none"> • Responsable métier du processus • Directeur métier du Processus • DSI 	<ul style="list-style-type: none"> • Audit interne/externe • Réunion d'équipe • Formalisation des processus
Comprendre les contraintes de toutes les parties prenantes sur les processus et les objectifs.	Direction générale Direction métier DSI Contrôle interne (si existant)	-Réunions des managers -Gestion des incidents quotidiens -Réunions régulière des équipes métiers pour remonter les problèmes de gestion des tâches.
Mesurer la performance de l'exécution du processus	Direction générale Direction métier DSI	-Edition Tableau de bord -Comparaison des résultats avec attentes -Comparaison avec la concurrence.
Simplifier la lecture des résultats directs et déduits	DSI	-outils technologiques -veille techniques (rapidité, convivialité, simplicité d'utilisation et d'échange)
Favoriser l'exécution des processus dès que possible	DSI Direction générale	Objectifs stratégiques urgents
Adapter aux entreprises et aux changements environnementaux,	DSI Direction générale Acteurs externe	Outils techniques Objectifs stratégiques urgents

Tableau 1.7: Projection des fonctionnalités GTI sur les acteurs de l'entreprise

Ces directives permettent de combler les lacunes des SI au niveau stratégique et d'augmenter leurs efficacités.

De ce fait, les grands éditeurs à l'échelle internationale, ont proposés des plateformes de gouvernance de types et de tailles différentes afin de répondre aux exigences du marché. Dans ce qui suit, nous présentons les solutions de gouvernance les plus utilisées et nous établissons un tableau comparatif pour en déduire les forces et les faiblesses.

I.7. Plateforme de Gouvernance des Systèmes d'Information

Les fournisseurs gouvernance, risque et conformité (GRC) continuent de montrer de solides progrès techniques et stratégiques, dans un marché en évolution continue dans ce domaine. Un ensemble de fournisseurs en concurrence souvent un contre l'autre, certains se focalisent sur la prestation de l'étendue des fonctionnalités, certains en création rapide de la valeur, un peu sur la flexibilité, et d'autres sur l'expertise de conseil, mais rarement tout cela à la fois. Forrester, entreprise indépendante spécialisé en études de marché sur l'impact des technologies dans le monde des affaires, (Forrester, 2011) a évalué en 2011 huit des principaux fournisseurs dans l'espace IT GRC en utilisant 59 critères. Il est signalé que le « **Forrester Wave** » est l'une des études les plus crédibles en matière d'IT GRC, c'est la raison pour laquelle nous sommes basé sur ses résultats pour cette étude comparative.

I.7.1. Fonctionnalités et Atouts des solutions

Pour évaluer l'état du marché de la GRC et des plateformes informatiques, (Forrester, 2011) a mené une évaluation rigoureuse des huit principaux fournisseurs comme le montre le tableau ci-dessous :

Editeur	Nom de la Solution	version du Produit	date Version
Agiliance	Agiliance RiskVision	V 6.0	Mai 2011
ANXeBusiness	ControlCase GRC	V4.0	Aout 2010
ControlCase	TruComply	V6	Décembre 2010
Easy2Comply	Easy2Comply	V4.7.5	Juin2011
Modulo	Modulo Risk Manager NG	V7.2	Mai 2011
RSA Archer	RSA Archer eGRC Platform	V5.0.6	Décembre 2010
Rsam	Rsam	V7.2	Octobre2010
Symantec	Symantec Control Compliance Suite (CCS)	V10.5	Décembere 2010

Tableau 1.8: Solutions IT GRC

Des critères de rigueur ont été imposés pour la sélection des solutions candidates. En effet, tous les fournisseurs de cette évaluation ont les capacités pour répondre aux grandes exigences des professionnels IT GRC, d'autant plus, plus de 50 clients titulaires d'une licence à l'aide de la solution IT GRC de chaque vendeur. Et enfin, tous les fournisseurs évalués ont déclaré plus de 50 clients de GRC, ont fourni des exemples de clients utilisant la plate-forme pour de multiples fonctions IT GRC, et ont soumis au moins cinq références clients pour participer à l'enquête auprès de « **Forrester Wave** ».

Agiliance RiskVision offre un ensemble de fonctionnalités de gestion de contenu, ainsi que la gestion GRC et de l'analyse. Elle propose une boîte à outils pour les services financiers, le secteur public, et les soins de santé.

TruComply de ANXeBusiness : c'est une solution en mode SaaS, comme plate-forme ITGRC primaire. Elle aborde les domaines axés sur des entreprises qui cherchent à automatiser les processus GRC à un coût minime.

ControlCase GRC : une plateforme qui cible un ensemble finis de cas d'utilisation GRC, mettant l'accent sur le temps de mise en œuvre des clients et la flexibilité. Le produit permet la gestion d'actifs, l'analyse de la vulnérabilité et la rationalisation du processus de mise en œuvre.

Easy2comply : plateforme flexible avec des cycles de déploiement courts et des capacités fondamentales qui peuvent être adaptées spécifiquement aux besoins du client expert. Elle est dotée d'un moteur de risque de pointe pour l'analyse statistique complexe.

Modulo Risk Manager se focalise sur des risques de l'information et des services de conformité. Son objectif est d'aider les clients à briser les silos entre les fonctions commerciales pour améliorer la gouvernance globale de l'entreprise. Elle vise les industries du secteur public et des services financiers.

RSA Archer eGRC Platform : La plate-forme RSA Archer se concentre sur la collecte du barrage quotidien de l'information et de la traduire en mesures pertinentes qui affectent l'entreprise, et la largeur de la plateforme de fonctionnalité reflète cette vision. La plateforme permet la gestion de

contenu, la gestion des risques et de contrôle, et la gestion des flux de travail. Elle offre des Workflows prédéfinis pour des bases de solutions IT GRC avec possibilité de définir des processus propriétaires en utilisant le glisser-déposer des fonctionnalités de développement.

Rsam: la plateforme très robuste et flexible. Son interface est logique et facile à naviguer, avec des capacités de configuration glisser-déposer. Elle est livrée avec les deux applications prédéfinies à des outils de contenus tiers, en plus elle soutient les processus uniques de l'entreprise. La solution orienté client existe tout au long du cycle de mise en œuvre.

Symantec Control Compliance Suite : plateforme marquée par sa conformité et qui prévoit de continuer à investir d'avantage pour développer le produit. CCS dispose d'une gestion GRC et de fonctionnalités d'analyse, capable d'incorporer des méthodes de quantification des risques fondées sur des données telles que les niveaux de vulnérabilité, la criticité des actifs et des fonctions de contrôle.

I.7.2. Etude Comparative

Un comparatif des points forts de chacune des solutions a été établi à partir de la description Forrester, des fonctionnalités présentées sur les sites officiels et des démos disponibles sur le net, du moment que les solutions sont toutes propriétaires. Ce qui a donné le tableau suivant :

EDITEUR	SOLUTION	POINTS FORTS
Agiliance	Agiliance RiskVision	<ul style="list-style-type: none"> • Gestion et analyse GRC • Services financiers • Secteur public et santé,
ANXeBusiness	TruComply	<ul style="list-style-type: none"> • Automatisation des processus GRC de l'entreprise.
ControlCase	ControlCase GRC	<ul style="list-style-type: none"> • GRC • Gestion d'actifs • Analyse de vulnérabilités.
Easy2Comply	Easy2Comply	<ul style="list-style-type: none"> • GRC • Engin de risque pour les analyses statistiques complexes.
Modulo	Modulo Risk Manager NG	<ul style="list-style-type: none"> • Amélioration de la Gouvernance globale de l'entreprise. • Secteur publics • Services financiers de l'entreprise
RSA Archer	RSA Archer eGRC Platform	<ul style="list-style-type: none"> • GRC • Gestion de contenu, • Gestion des risques • Contrôle de workflows de gestion
Rsam	Rsam	<ul style="list-style-type: none"> • Assistance et outillage GRC (prédisposé pour les industries)
Symantec	Symantec Control Compliance Suite (CCS)	<ul style="list-style-type: none"> • Gestion et analyse GRC, • Méthodes de quantification des risques fondées sur des données (niveaux de vulnérabilité, la criticité des actifs et des fonctions de contrôle)

Tableau 1.9: Comparaisons solutions IT GRC

En plus de ce comparatif d'ordre fonctionnel autour des points forts de chaque solution, Forrester Wave a évalué les plateformes selon les critères suivants :

- Gestion de contenu,
- Gestion des risques et de contrôle,
- Workflow management,
- Gestion et analyse GRC,
- Appui à la gestion d'actifs
- Soutien aux risques de l'entreprise et de la conformité,
- GRC soutien de domaine,
- Fonctionnalités techniques,
- Références clients,
- Vision et stratégie de la société,
- Vision et stratégie du produit,
- Soutien pour les rôles GRC,
- Viabilité financière.

L'analyse de comparatif et la mesure de l'efficacité de ces solutions par rapport aux facteurs de dépendance GTI cités dans la section précédente, permettent de déduire que:

- Il s'agit dans la majorité des cas de solutions GRC et non IT GRC, la gouvernance des SI n'est qu'une fonction supplémentaire.
- La majorité de ces solutions sont des modules de solutions ERP à déployer avec le reste des modules.
- Le coût de la mise en place est élevé (en consultant la liste de leurs références sur le marché et leurs cibles).
- Nécessité de conseil externe pour une meilleure exploitation (éventuellement le conseil du vendeur de la solution).
- Il y a une discordance par rapport à l'implémentation de la GTI dans les SI d'entreprises divers (taille, chiffre d'affaire, type, maturité, localisation, etc.).
- Elles ne concernent pas toutes les parties prenantes du SI et tous les domaines de l'entreprise à la fois.

I.8. Problématique

De ce qui précède, nous constatons que la gouvernance des systèmes d'information au sein de l'entreprise n'est pas seulement un ensemble de directives à mettre en place par la DSI et le top management, mais c'est un échange entre eux, voir même un canal de remontée de l'information à partir des directions métiers. Ces échanges peuvent servir aussi bien pour exiger de nouveaux services que pour évaluer les services informatisés existants.

En effet, pour une meilleure gouvernance du SI de l'entreprise, les acteurs à mobiliser sont :

- Les parties prenantes internes à l'entreprise intéressées par les investissements informatiques générant de la valeur telles que :
 - Les décideurs d'investissements,
 - Les responsables qui définissent les exigences,
 - Les utilisateurs des services informatiques.

- Les parties prenantes internes et externes qui fournissent les services informatiques telles que :
 - Les gérants de l'organisation et des processus informatiques,
 - Les utilisateurs du système d'information au quotidien.

- Les parties prenantes internes et externes qui ont des responsabilités dans le contrôle et le risque telles que :
 - Les responsables de la sécurité, du respect de la vie privée et des risques,
 - Les responsables de la conformité aux lois et aux réglementations,
 - Les auditeurs en général.

Il s'agit donc d'un ensemble de flux d'information (**Workflow**) acheminés de bout en bout entre la DSI et ses utilisateurs potentiels d'une part et la DSI et les décideurs d'autre part.

La mise en place d'une plateforme de gouvernance SI s'avère essentielle pour les raisons suivantes:

- Marché IT ou TI concurrentiel et continuellement changeant.
- Systèmes d'information de composantes hétérogène et divers.
- Flux d'information et processus de plus en plus complexes.
- Manque de visibilité pour une décision stratégique du top management en matière de TI.

Face aux solutions existantes comparées dans la section précédente, notre approche de modélisation consiste à concevoir un Workflow de la gouvernance SI qui gère les flux entre l'ensemble de ces acteurs pour :

- Un meilleur alignement des technologies de l'information avec la stratégie de l'entreprise,
- Une création de la valeur des services métiers par le biais du SI bien gouverné,
- Une gestion des risques SI et métiers,
- Une maîtrise des ressources humaines et matérielles,
- Une mesure de la performance des TI au sein de l'entreprise.

Vu les contraintes du contexte de GTI et face à des systèmes d'information souvent hétérogènes avec des composants faiblement couplés, nous avons opté pour un Workflow inter-organisationnel de type tâche.

D'autant plus que dans la section précédente, nous avons mis l'accent sur l'aspect fédérateur de COBIT en tant que référentiel global de la GTI et sa capacité à déclencher d'autres référentiels, nous fondons cette approche sur COBIT pour tirer profit de ces atouts et surtout de la diversité de ses composants pour réussir la GTI. Cependant pour ne pas limiter cette approche à un seul référentiel ou tomber dans le piège d'informatisation de référentiel qui n'a pas été un vrai succès, nous nous servons de COBIT comme noyau et nous cumulons les spécificités de l'entreprise pour des résultats de plus en plus performants, d'autant plus, COBIT est un référentiel orienté processus, concept qui ouvre des perspectives de programmation et d'informatisation assez étendues.

I.9. Conclusion

Dans ce chapitre, après une étude comparative d'un échantillon significatif de solutions IT-GRC, nous avons soulevé plusieurs limitations, à savoir :

- L'appartenance à une suite ERP couteuse et délicate à mettre en œuvre,
- La prise en charge du volet gouvernance de l'entreprise dans sa globalité sans spécifier pour autant le volet GTI,
- Le manque d'implication des différents acteurs de l'entreprise c.à.d. solution restreinte à la DSI,
- La nécessité des profils experts GTI pour la bonne utilisation sans prendre en considération l'utilisateur simple du SI.

C'est pour y remédier et pour combler l'inexistence d'architecture de fondement théorique et scientifique, qu nous avons proposé la solution qui fait l'objet de cette thèse.

Après une étude bibliographique des référentiels GTI les plus utilisés, nous avons mis l'accent sur deux aspects très importants, à savoir :

- L'aspect orienté processus des référentiels de gouvernance SI,
- L'aspect fédérateur du référentiel COBIT par rapport aux autres référentiels.

Par la suite, nous exploitons ces deux aspects respectivement pour proposer une architecture orienté agent (en analogie avec l'orienté processus) basée sur un noyau COBIT qui permet une meilleure gouvernance SI ainsi qu'une ouverture sur les autres référentiels GTI.

CHAPITRE 2 :
WORKFLOWS INTER-
ORGANISATIONS ET SYSTEMES
MULTI-AGENT

II. Workflows Inter-Organisations et Systèmes Multi-Agents.

II.1. Workflows et Workflows Inter-organisationnels

II.1.1. Introduction

Le workflow est une technologie indispensable pour l'automatisation des processus et la gestion des opérations métiers de l'entreprise.

Dans la première partie de ce chapitre, nous détaillons les différents aspects relatifs aux workflows, à savoir, les architectures, les langages et les coordinations inter-organisationnelles.

Dans la deuxième partie, nous présentons un état de l'art des Systèmes Multi Agents, leurs types, leurs architectures ainsi que leurs langages de communication et techniques de coordination.

Dans la dernière partie de ce chapitre, nous décrivons les plateformes les plus connues de développement des systèmes multi-agents en justifiant le choix de la plateforme MADKIT.

II.1.2. Terminologie

Le but du Workflow (WF) en général est d'automatiser les processus dans les organisations. Actuellement les WFs apparaissent comme une nouvelle technologie dans la gestion des opérations commerciales.

Le WF est la spécification des opérations qui existent dans l'organisation. Cette spécification doit tenir compte de plusieurs aspects, tels que la séquence des opérations, la complexité et la nature des transactions, la disponibilité et l'emplacement des ressources.

La croissance économique et industrielle ainsi que la diffusion des nouvelles technologies de l'information et de la communication, ont permis l'apparition de plusieurs modes de compatibilité de workflow inter-organisationnel, tels que les possibilités d'échange associés à la mise en œuvre, l'externalisation, les cas de transfert, le transfert étendu au public et privé.

Le processus comme défini précédemment, est une série de tâches coordonnées, selon un traitement qui conduit à un résultat spécifique (par exemple, le suivi des dossiers médicaux, le remboursement ou les procédures de recrutement). Nous évoquons le WF pendant le déploiement des processus, d'information, des formulaires ou de documents partagés ou envoyés d'un poste de travail à un autre pour un traitement donné.

Trois modèles sont généralement utilisés pour décrire la structure, les ressources et les informations nécessaires à la mise en œuvre d'un Workflow :

- Le modèle organisationnel qui gère les rôles et les autorisations d'assignation des ressources pour effectuer certaines tâches du processus ;
- Le modèle informationnel qui décrit la structure des formes, des documents et des données qui sont utilisées et produites ;
- Le modèle de processus qui définit les composantes de la tâche, la coordination, l'information et les ressources impliquées dans chaque tâche.

L'association mondiale de normalisation dans le domaine du Workflow (WFMC) a pris en charge la collecte des définitions de la terminologie des workflows. Nous présentons ci-dessous quelques notions clés :

- **Une tâche** est une unité de travail représentant une étape d'un processus qui fait appel à des ressources humaines ou matérielles pour son accomplissement.
- **Un processus d'affaire** ou **processus d'entreprise** est un ensemble de tâches ayant un déclencheur commun, reliées entre elles par des flux d'information ou de matières. Elles se combinent pour fournir un produit matériel ou immatériel important et bien défini que nous rattachons à un client externe ou interne.
- **Un workflow** (Hsieh & Lin, 2016) est un flux d'automatisation, complète ou partielle, des processus métier d'exécution au cours desquels, des documents et des informations sur le problème d'un participant sont échangés.
- **Une définition de processus** décrit l'aspect comportemental d'un workflow tel que : les tâches; la structure des enchaînements des tâches, la dépendance d'exécution entre les tâches ; les critères de lancement et d'achèvement ; les informations relatives aux tâches.
- **Une instance de processus** est un cas correspondant à une exécution d'une définition de processus. Une définition de processus peut être instanciée plusieurs fois, et plusieurs instances peuvent s'exécuter concurremment.
- **Système de gestion de workflow** (SGWF) est un système qui définit, implémente et gère l'exécution d'un ou plusieurs workflows, à l'aide d'un environnement logiciel fonctionnant avec un ou plusieurs moteurs de workflow et capable d'interpréter la définition d'un processus, de gérer la coordination des participants et d'appeler des applications externes.

La figure ci-après montre la liaison entre les différentes notions définies ci-dessus autour des workflows.

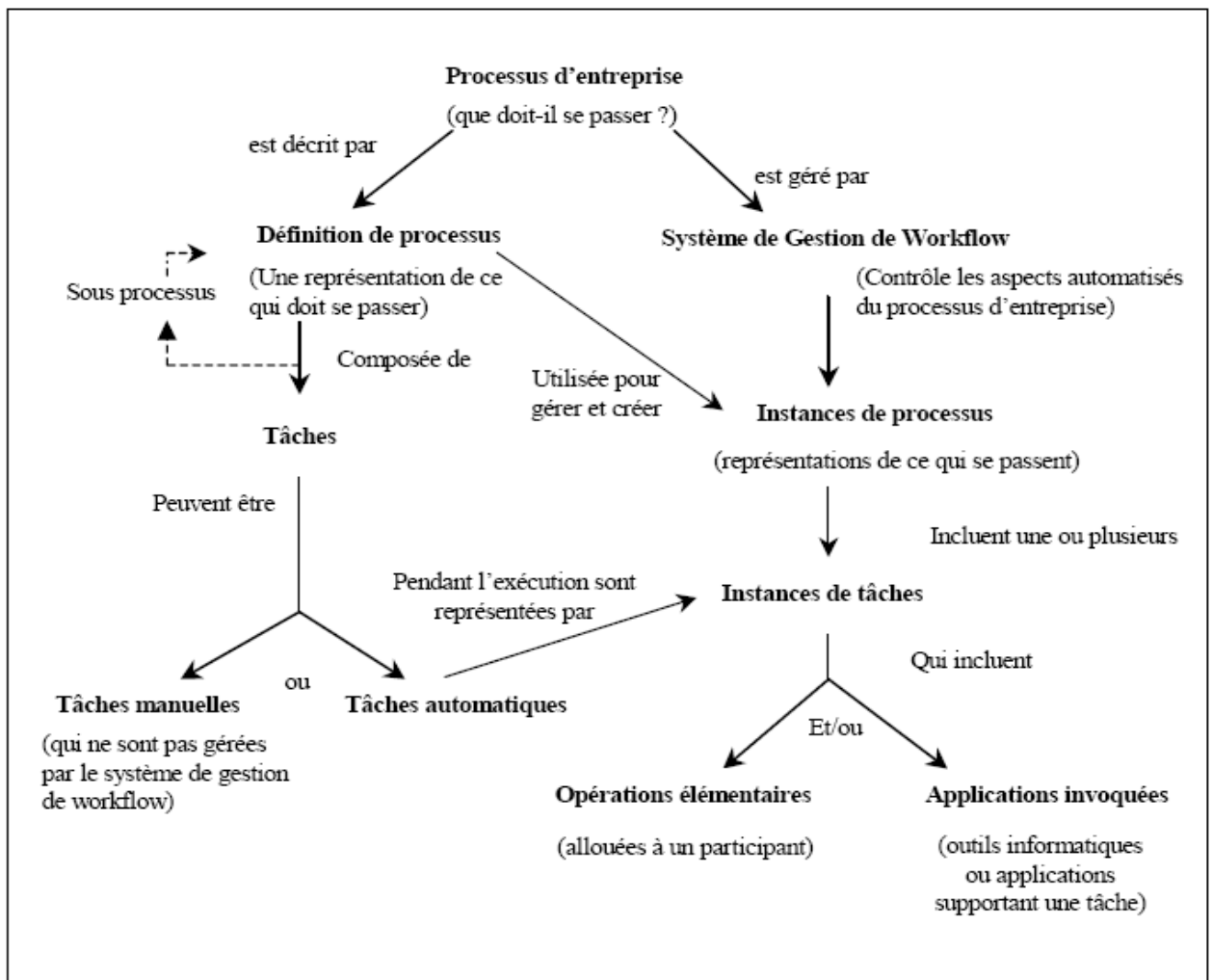


Figure 10: Relation entre notions autour des workflows

Il existe plusieurs types de workflow selon leurs utilisations (Amirreza, 2009) :

1. **Workflow d'administration:** gestion administrative de l'entreprise, plus orienté Gestion Documentaire (GED).
2. **Workflow de production :** permet de gérer les processus de production de l'entreprise, en mettant en valeur les capacités compétitives de l'entreprise (Leymann & Roller, 2000).
3. **Workflow de collaboration:** destiné pour un grand groupe de collaborateurs, caractérisé par la complexité du déroulement de travail, il manque de flexibilité par rapport au changement de l'état initial.
4. **Workflow ad-hoc:** hybride entre 1-2-3, il gère plutôt les situations exceptionnelles. Il n'existe pas de structure prédéfinie (Chebbi, 2007).
5. **workflow inter-organisationnel:** WIO est une extension du workflow traditionnel qui permet de coopérer plusieurs processus issus de plusieurs composantes (organisations) réparties, autonomes et hétérogènes. Il se caractérise par:

- ✓ La répartition des processus d'organisations.
- ✓ L'autonomie des organisations.
- ✓ L'hétérogénéité des organisations à faire coopérer.

Ce type de workflow a vu le jour avec l'insuffisance des workflows classiques (van der Aalst, ter Hofstede, Kiepuszewski, & Barros, 2003) face à la dynamique du marché et l'évolution des technologies de l'information et de la communication. En effet, les processus sont de plus en plus complexes et leur exécution s'étale sur un ensemble d'organisations (Zarcoski, 2015) (exemples : commerce électronique, entreprises virtuelles, projet inter-organisations, multinationale, etc.). Il s'agit de collaborations qui se déroulent à long ou à court terme avec des partenaires prédéfinis ou sélectionnés selon le besoin ou l'objectif. Nous pouvons donc dire qu'un WIO est un ensemble fini de workflow avec Système de Gestion de workflow (SGWF) local plus un modèle de coordination adéquat.

Dans cette perspective, le WIO fait face à plusieurs contraintes d'implémentation dans un environnement de travail, à savoir :

- l'hétérogénéité des plateformes de développement des SGWF devant coopérer, des ressources à mettre en œuvre et de la sémantique des données échangées ;
- l'autonomie de chaque organisation participant au WIO par rapport aux conditions de coopération, tout en contrôlant les tâches locales ;
- la flexibilité de l'allocation des tâches et du choix des partenaires ;
- la distribution des processus, des informations et des ressources mises en œuvre. Cette caractéristique se manifeste par la distribution géographique des processus composants, des informations et des ressources. Elle exige des infrastructures pour faire communiquer et pour coordonner ces différents éléments ;
- l'environnement technologique.

Les WIO disposent généralement de deux catégories de problèmes global et local (Divitini, Hanachi, & Sibertin-Blanc, 2001):

- Problème local : il faut que le SGWF soit capable de communiquer avec les autres SGWF pour partager des modèles organisationnels, informationnels et de processus, tout en gérant les modalités de partage.
- Problème global : le SGWF global doit être capable de collaborer avec les composants, tout en gérant l'organisation et l'architecture du groupe, les protocoles et les engagements communs.

II.1.3. Architecture et langages

Plusieurs dispositifs ont été mis en place pour la modélisation et le développement des workflows en général, les WIO en particulier. Il est d'abord judicieux d'avoir un aperçu sur l'architecture de référence du workflow pour pouvoir ensuite imbriquer les spécificités de chaque contexte. Nous détaillons les langages les plus utilisés dans ce propos.

II.1.3.1. Architecture de référence

L'architecture de référence de workflow (Ouyang, Adams, Wynn, & ter Hofstede, 2015) est représentée comme suit :

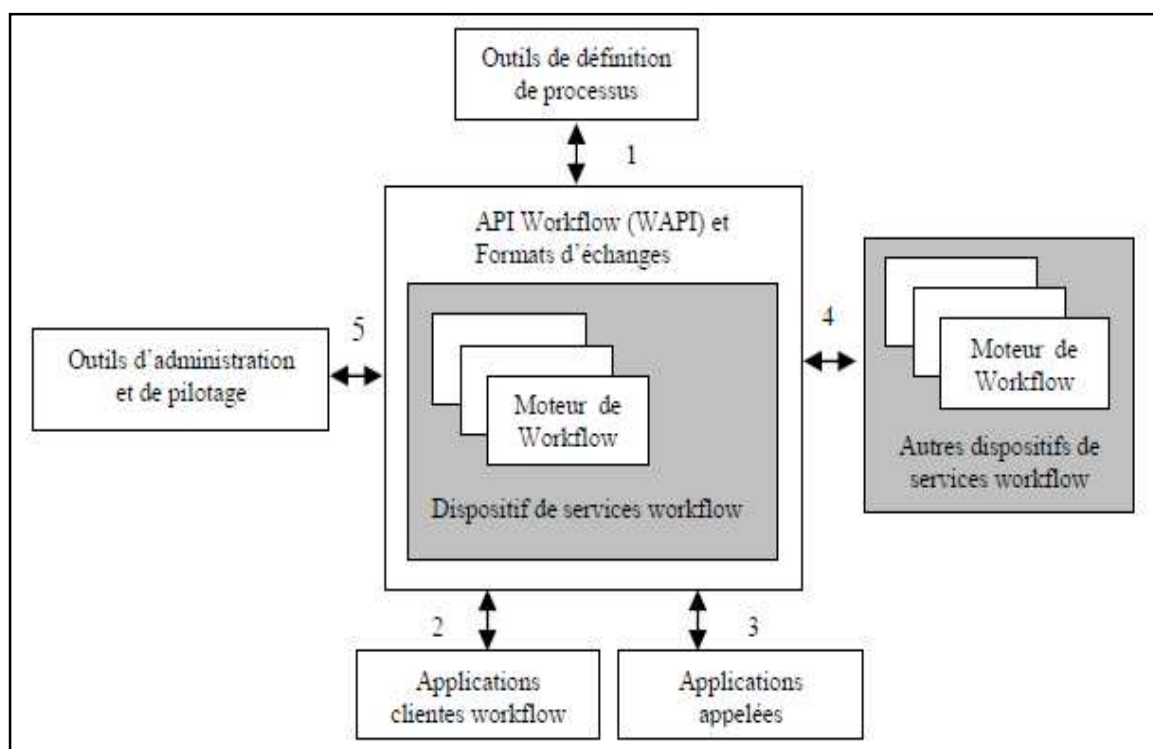


Figure 11 : Architecture de référence du Workflow

1. Outils de définition de processus

Il s'agit de saisir et valider la définition des processus d'entreprise en des schémas de processus via IHM, en mettant l'accent sur les ressources impliquées dans chaque tâche. L'interface 1 permet d'importer/exporter la définition de processus pour le moteur de workflow. Nous pouvons utiliser des langages d'échange, tels que XPD (XML Process Definition Language).

2. Moteur de workflow

C'est l'environnement d'exécution d'une ou plusieurs définition(s) de processus importée(s) par l'interface 1 ou faisant appel à d'autre moteur. Il permet :

- la création des instances de processus (cas) et leur gestion : démarrages, suspensions, réinitialisations, etc ;
- l'interprétation des modèles ou définitions de processus ;
- la synchronisation des instances de tâches et la gestion des corbeilles
- l'allocation des instances de tâches aux ressources (humain ou machine) ;
- l'enregistrement et le contrôle de la cohérence de l'exécution ;

3. Applications clientes workflow

C'est l'interface de l'utilisateur final pour l'exécution d'une instance de processus. Elles permettent d'accéder à la liste des instances de tâches appropriées (worklist ou corbeille). La plupart des systèmes actuels (notamment FlowMind et YAWL) proposent des navigateurs Web comme IHM.

4. Applications appelées par le workflow

Les SGWF doivent s'intégrer dans le SI des organisations dans lesquelles ils sont déployés. Ils doivent notamment pouvoir communiquer avec d'autres applications, nécessaires à l'accomplissement des tâches du processus telles que, l'accès à un SGBD pour recueillir de l'information, l'accès à un LDAP pour retrouver des informations sur le personnel, l'appel au service de messagerie et divers autres outils bureautique.

5. Autres moteurs de Workflow

Le transfert de contrôle entre moteurs de Workflow pour déléguer l'exécution d'un cas dans un autre moteur est possible. Plusieurs niveaux d'interopérabilité existent, allant de l'interopérabilité matérielle à l'interopérabilité sémantique concernant des formats d'échanges partagés et la compatibilité des protocoles ou des environnements de travail.

6. Outils d'administration et de pilotage

Un ensemble d'outils d'administration sont mis à la disposition de l'utilisateur permettant :

- la gestion des participants associés à l'exécution d'une instance de processus Workflow
- la gestion des concepteurs chargés de produire des définitions de processus ;
- la supervision et le contrôle des cas en cours d'exécution ;
- l'audit et l'analyse du fonctionnement du système. A partir des historiques d'exécution (fichier log), des indicateurs de performances peuvent être calculés et conduire à une réingénierie des processus.

Il est à noter que ces six aspects sont gérés partiellement ou totalement lors de la mise en place d'un Workflow. Nous nous en inspirons lors de la mise en place du Workflow Inter-organisationnel de Gouvernance des SI.

II.1.3.2. Langages de modélisation des WF

La phase précédant la conception d'un WF est la spécification des processus métier, dans cette perspective plusieurs langages sont apparus, chacun adopte sa terminologie particulière et définit ces concepts spécifiques en fonction des objectifs visés par le méta-modèle de workflow, du domaine d'intérêt ou du choix de représentation. Les langages sont spécifiques au modèle à décrire comme le montre le tableau ci-dessous (Lemos, Daniel, & Benatallah, 2015) :

Modèle	Langages
Informationnel (MI)	langage de description de données : Objet-relation ; XML
Organisationnel (MO)	langage de description de données Objet-relation ; XML
De processus (MP)	Notations ou formalismes : Réseaux de Petri / Diagramme UML, XPDL, WIDE

Tableau 2.1: Modèles et langages de Workflows

a) XPDL

XML Process Definition language est le langage proposé par la WfMC pour le modèle de processus des workflows. Il utilise le langage XML comme support, un processus Workflow est représenté par (**Workflow Process**), il est fait de plusieurs activités qui sont (**Workflow process Activity**) qui peuvent être atomiques (tâches élémentaires) ou être des appels vers des sous-processus. Ces activités sont ordonnées grâce aux informations de transition (Transition Information) par rapport à des données pertinentes pour le workflow (**Workflow Relevant Data**). Les activités sont effectuées par des ressources humaines ou des entités organisationnelles définies par l'intermédiaire des participants (**Workflow Application Specification**). Elles peuvent appeler des applications externes via des déclarations d'applications (**Workflow Application Declaration**).

b) WSFL

Web Service Flow Language est un langage proposé par IBM, pour la composition des services web, il dispose de deux types de composition :

- ✓ Modèle de flux : Il représente le processus métier et traite de la composition explicite de la succession des étapes et de l'enchaînement des appels aux opérations des services web ;
- ✓ Modèle global : Il décrit une collection de liens entre les opérations duales de services web, ce modèle correspond à un modèle d'interactions de services web deux à deux ;

c) WIDE

Il possède son propre langage de description de Workflow basé sur les données actives qui fournissent en plus du langage de description de données relationnel pour décrire le MI et le MO, des règles pour décrire le comportement dynamique du MP. La syntaxe des règles actives est comme suite:

Quand <Événement> Si <Condition> Alors <Action>.

Quand un événement E se produit, si la condition C est vérifiée alors l'action doit être exécutée.

Les événements peuvent être :

- ✓ des opérations classiques sur la base de données « CRUD »,
- ✓ des événements temporels (date absolue ou relative)
- ✓ des signaux provenant d'autres applications.

La partie condition est un prédicat qui doit être satisfait par les objets de la base de données (donc le MI et le MO). L'action est un programme qui peut inclure des opérations aussi diverses que la notification d'informations à des usagers, les opérations sur la base de données, l'invocation d'autres outils, etc.

d) XLANG

Langage proposé par Microsoft, utilisé pour l'orchestration des activités constituant un processus métier. IL utilise les concepts suivants pour décrire un processus (Thatte, 2001) :

- ✓ **All** : indique une exécution parallèle des sous-processus.
- ✓ **Compensate** : invoque une compensation d'une transaction.
- ✓ **Context** : un contexte forme un cadre pour les déclarations locales, la gestion des transactions et la gestion des exceptions.
- ✓ **Empty** : un processus vide.
- ✓ **Pick** : attend l'arrivée d'événement pour déclencher des processus. L'événement peut être la levée d'un signal ou la fin d'une action.
- ✓ **Sequence** : une séquence de processus et actions.
- ✓ **Switch** : définit des branches conditionnelles, chacune reliée à un processus.
- ✓ **While** : définit une boucle sur un processus.

e) WSCL

Web Service Conversational Language permet de décrire les services web à l'aide des documents XML en se basant sur leurs conversations. Les éléments d'une spécification WSCL sont les suivants (Ammar, Malik, Medjahed, & Alodib, 2015) :

1. Description des types de documents : elle référence les types de documents qui peuvent être échangés ou reçus. Les documents sont référencés par leurs URI respectifs.
2. Interactions : elles modélisent les actions de conversations sous la forme d'échange de documents. Il existe cinq types d'interactions :
 - ✓ *Send* : le service envoie un document.
 - ✓ *Receive* : le service reçoit un document.
 - ✓ *SendReceive* : le service envoie un document et en attend un au retour.
 - ✓ *ReceiveSend* : le service reçoit un document et en renvoie un au retour.
 - ✓ *Empty* : aucun document n'est échangé, le rôle de ce type d'interaction est de modéliser le début et la fin d'une conversation.
3. Transitions : elles spécifient la relation d'ordre entre les opérations. Pour cela, une transition possède une interaction source et une interaction de destination. Il est également possible de spécifier un type de document pour la transition source, ainsi qu'une condition sur la transition.
4. Conversations : elles listent toutes les interactions et les transitions qui forment une conversation. Elles définissent également les propriétés supplémentaires telles que le nom de la conversation et les interactions de début et fin.

f) YAWL (Yet Another Workflow Language)

YAWL (Van Der Aalst & Ter Hofstede, 2005) est un langage de Workflow basé sur les patrons de Workflow. Il se base sur les réseaux de Pétri (Mehandjiev & Grefen, 2010). La spécification de Workflow dans YAWL est un graphe orienté appelé WF-net. Dans ce graphe, il y a des tâches atomiques et des tâches composées. Chaque tâche composée rapporte à un WF-net qui contient son expansion, une tâche atomique correspond à une action atomique, c'est-à-dire, une action qui est exécutée par un utilisateur ou par une application logicielle.

g) Réseaux de Petri et WF-nets

Les réseaux de Petri (RP) sont un outil puissant de modélisation de systèmes à événements discrets, utilisés depuis leur introduction par Carl Petri en 1962 dans sa thèse pour modéliser différents systèmes et applications. Les workflow nets (WF-nets) introduits par Vander Aalst en 1998 sont les

modèles les plus utilisés pour modéliser, analyser et vérifier les workflows. Vander Aalst (1998) justifie l'utilisation des SGWF basés sur les réseaux de Petri (RP) par la représentation de la logique métier par un langage formel et graphique, en plus de la possibilité de modéliser explicitement l'état d'un cas d'exécution d'un processus et l'abondance de techniques d'analyse pour démontrer des propriétés structurelles des RP telles que vivacité, blocage, « soundness », ... qui permettent de vérifier l'absence d'anomalies sans connaissance du domaine du processus en question.

II.1.4. Conception des WIO

Pour modéliser un workflow selon le WFMC. Il faut définir ses quatre niveaux ou modèle :

- Niveau Organisation:** décrit les structures organisationnelles, les acteurs du système et leurs rôles. Ce modèle s'intéresse à l'organisation des ressources. Une ressource peut être un humain, une machine, un logiciel, voir un logiciel agissant pour le compte d'un humain et capable de réaliser des tâches d'un workflow. Le modèle organisationnel a un double objectif. D'abord, il structure les ressources en classes d'objets partageant les mêmes caractéristiques fonctionnelles. Une classe est appelée un rôle lorsqu'elle comporte des ressources ayant les mêmes caractéristiques ou bien une unité organisationnelle lorsque les ressources sont des participants appartenant à une même structure organisationnelle

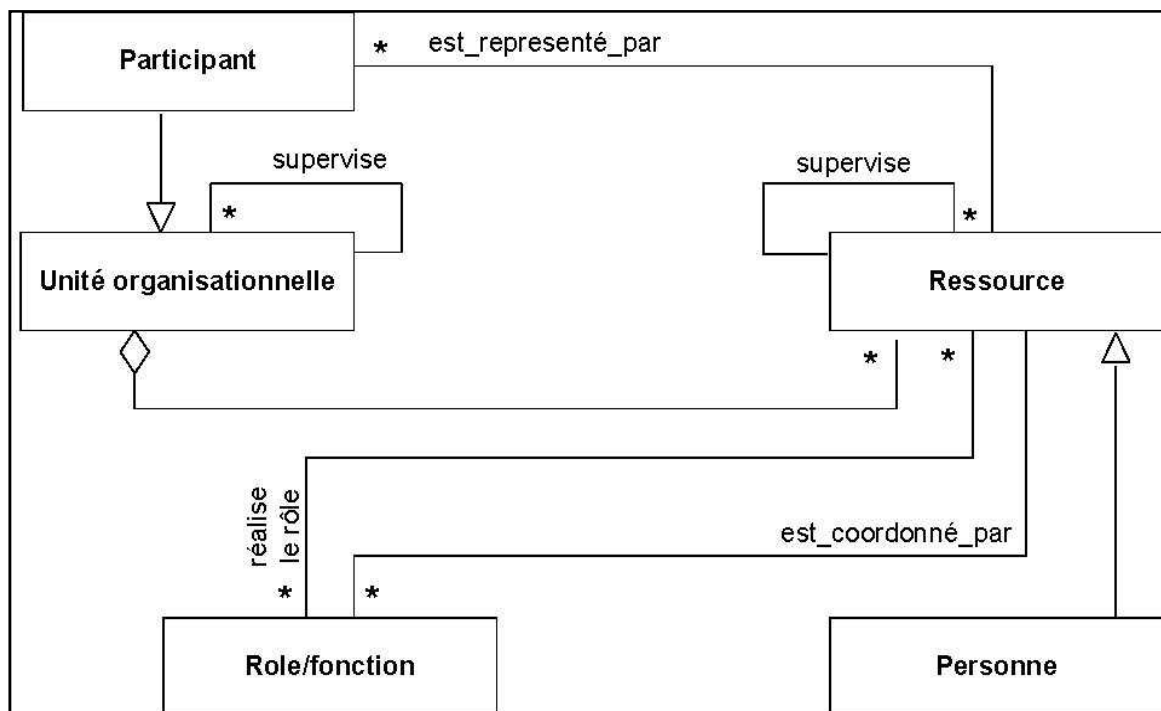


Figure 12: Méta-modèle organisationnel

- Niveau Fonctionnement :** décrit les fonctionnalités du système, en particulier les activités des processus que l'on souhaite modéliser toute en établissant les scénarios de décompositions des

sous-processus. Il permet aussi de représenter le flux et les interdépendances de données relatives aux activités.

- **Niveau Comportement** : décrit les flux de contrôles et événements rattachés aux activités et tâches. En effet, il concerne la dynamique du processus en indiquant la chronologie de l'exécution des activités, leur flux (séquentiel ou parallèle), les points de synchronisation ou de disjonction. D'autant plus qu'il représente les événements qui permettent de déclencher les activités. On l'appelle aussi le modèle de coordination.
- **Niveau Information** : présente la partie du système d'information nécessaire à l'accomplissement du travail. Il décrit en détail les relations qui existent entre les données, leur type et leur structure. Dans une perspective informationnelle, une application est un terme général pour désigner un programme logiciel qui interagit avec un SGWF. Elle est appelée par le SGWF pour automatiser, complètement ou partiellement une activité, ou pour aider un participant à réaliser sa tâche. Une application est dite cliente si elle demande un service au SGWF. Un bon de travail, quant à lui, est une représentation d'un travail à réaliser pour un rôle. Une instance d'activité génère un ou plusieurs bons de travail qui réalise(nt) à leur tour la tâche de cette dernière. Il est en général présenté aux ressources via leur corbeille de bons de travail. Une fois qu'une instance d'activité est créée, un bon de travail correspondant est généré et déposé dans la corbeille de bons de travail de toutes les ressources ayant le rôle nécessaire pour réaliser l'activité. La première ressource qui choisit un bon de travail est responsable de la réalisation de ce travail; cette ressource devient un participant du processus.

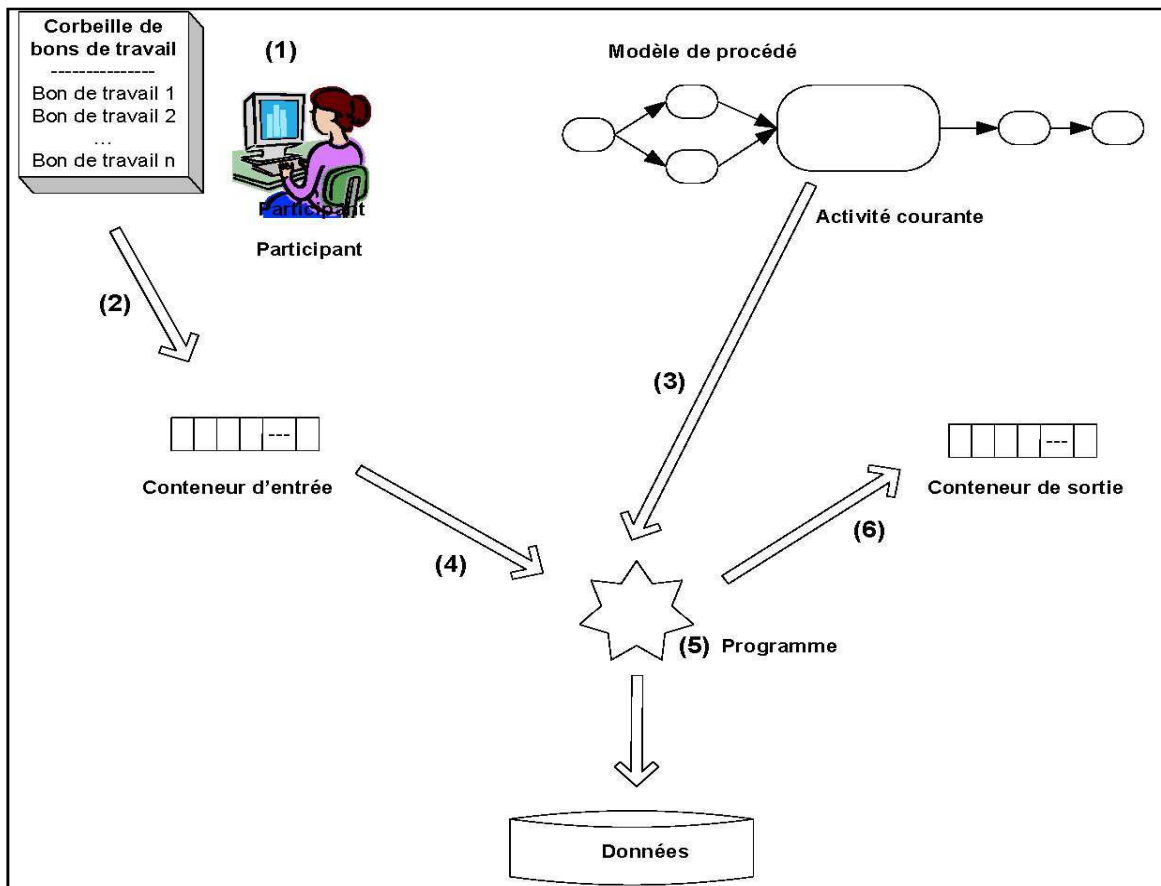


Figure 13: Exécution d'activité au niveau informationnel

Pour conclure, la conception et la vérification des processus des workflows Inter-organisationnels sont essentielles dans la conception des SGWF, la première donne les entités formelles pour le processus à implémenter. La deuxième, vérifie leur cohérence pour éviter les situations de deadlock (impasse) et overlook. A ce niveau là, le souci majeur est la coordination ou l'interopérabilité des partenaires de WIO qui fera l'objet de la partie suivante.

II.1.5. Coordination dans les systèmes WIO

La coordination dans le WIO se définit comme l'interaction des composants WF entre eux de manière à ce que l'ensemble (le WIO) forme un système opérationnel et performant.

La coordination dans le WIO lâche est nécessaire pour plusieurs raisons:

- l'insuffisance individuelle des processus, des ressources ou des informations. En effet, la spécification des WF locaux, les processus, les ressources et les informations à la disposition de chaque WF local considérés individuellement sont insuffisantes pour qu'il atteigne seul l'objectif global : c'est ce qui entraîne la nécessité de les mettre en commun, de les partager et de les échanger. Cela revient à recourir à des mécanismes de répartition des processus, à des règles d'accès et de partage des ressources, et à des protocoles d'échanges d'information entre organisations. La coordination devient ici nécessaire pour réglementer ces interactions ;

- **la répartition du travail pour l'atteinte de buts communs** : Un SGWF inter-organisationnel est constitué de plusieurs moteurs WF exécutant des processus locaux mais devant rendre conjointement un service global. Il faut pour cela qu'en cours d'exécution du processus global, chaque moteur WF ait un rôle bien défini. Chaque organisation peut tenir son rôle soit en accomplissant bien précisément les tâches qui lui ont été attribuées, soit en agissant de manière opportuniste en tirant profit d'observations ou de demandes faites dans l'environnement. La coordination consiste à attribuer un rôle à chaque organisation et/ou contraindre son comportement pour l'orienter correctement vers l'objectif commun ;
- **l'interdépendance entre les processus des organisations** : Il arrive également que les différents processus entretiennent des dépendances mutuelles. Ces interdépendances nécessitent d'entrelacer, de faire communiquer et de synchroniser ces différents processus. Si l'on considère une coopération entre différentes compagnies pour organiser un voyage sous forme de circuit, le processus d'organisation des activités culturelles est en dépendance mutuelle avec le processus de réservation d'hôtels : on choisit des activités aux alentours des hôtels sélectionnés, et en même temps on voudra rechercher des hôtels proches de manifestations culturelles incontournables ;
- **la satisfaction de contraintes non fonctionnelles** : Le regroupement d'organisations autour d'un objectif commun introduit également des contraintes non fonctionnelles à respecter par l'ensemble du WIO, telles que le temps limité dont les composants du WIO disposent pour rendre le service en un délai raisonnable, le coût financier du service... Ici, à nouveau, le respect des contraintes liées à la qualité du service nécessite de coordonner les processus des organisations.

Plusieurs approches d'interopérabilité et de coordination de WIO ont été mises en place, le tableau ci-dessous résume l'essentiel :

Approche de coordination	Description	Phases	Avantages	Inconvénients
CrossWork (Mehandjiev & Grefen, 2010)	Système multi-agents	-déterminer les partenaires -former le workflow -coordonner les activités	la plateforme de modélisation XRL basée sur les réseaux de Pétri	Inadéquation de représentation des workflows par des boîtes noires (manques d'inter-visibilité)
CrossFlow (Boukhedouma, Alimazighi, Oussalah, & Tamzalit, 2012)	Services externalisés	-établissement de contrat, -installation de l'infrastructure -exécution de service	service de «Matching» -Gestionnaire de contrats - Services de support à la coopération	Non conformité à l'architecture de référence.
DERPA	Services	synchronisation de l'exécution des tâches,	- Service d'événements	Ambiguïté

(Andonoff, Bouaziz, Hanachi, & Bouzguenda, 2009)	d'événements	échange de données, notification d'événements importants.	indépendant de tout SGWF	d'événements d'un point de vue sémantiques.
ebXML (Patil & Newcomer, 2003)	Echange de message XML	-implémentation de spécification - recherche et de négociation des informations des partenaires -exécution des transactions.	Assure un cadre sémantique d'interopérabilité des données.	-ne préserve pas le savoir-faire des partenaires et les SGWF -flexibilité réduite
CoopFlow (Chebbi, 2007)	Orienté Service	-la publication, -la recherche -la connexion	-Flexible -Assure l'autonomie	

c

II.1.6. Scénarios de Workflow Inter-organisationnel

Il existe plusieurs formes et scénarios de coordination entre Workflow Inter-organisationnels (Chebbi, 2007) :

Types	description	Avantages	Inconvénients
Partage de Capacité	WF global fixé à la conception, coopération à la charge d'une organisation	simple et utilise une infrastructure commune ainsi qu'une gestion centralisée d'un Workflow unifié	difficile de maintenir la cohérence du système ainsi que la mise en place des changements
Exécution chaînée	WF global = Suite (plusieurs WF locaux) qui s'exécute séquentiellement	ne requiert pas d'exécution parallèle et se base sur un contrôle distribué de Workflow	Non conforme aux applications dynamique, rigide au changement au court d'exécution, un seul scénario de coopération.
Sous-traitance	WF principal délègue la coordination d'une partie de ses tâches à des WF atomiques	Encapsulation du savoir faire des prestataires	Ne supporte pas les coopérations complexes et les différents types d'interaction.
Transfert de cas	Description unique du WF et spécification par partenaire (partition verticale)	Complémentarité des vues et pas de risque de non exécution, maturité des partenaires	Non dynamique, ne préserve pas le savoir faire des partenaires

Transfert de cas étendu	Description unique du WF et spécification par partenaire (partition verticale) + ajout ou changement d'activité possible	Personnalisation des WF des partenaires,	Charge des moteurs de WF partenaire sans préservation total du savoir faire
Faiblement couplé	WF global divisé en sous WF doté d'une interface de communication+protocole + Diagramme de séquence	Nombre illimité d'organisations autonomes (WF local indépendant)	Soucis de mise à jour des activités, de distribution et de flexibilités des organisations
Public-Privé	Conception WF public+partition de WF Public +conception des WF privés	Basé sur l'héritage, répartition par domaine	Ne préserve pas les WF préétablies des partenaires.

Tableau 2.3: forme de coordination WIO

II.1.7. Synthèse

Après avoir étudié les approches d'interopérabilité et de coordination de WIO par rapport à la problématique GSI, nous avons opté pour:

- un workflow inter-organisationnel pour pouvoir coopérer des processus provenant de plusieurs organisations (module du SI ou applicatif métier sans contraintes sur sa dimension et l'acheminement de l'information au niveau local).
- Le type faiblement couplée, pour profiter du nombre illimité d'organisations autonomes mises en jeu et aussi de la possibilité d'extension des partenaires et de services (diversité des référentiels et bonne pratique et changement de version).
- Une approche d'interopérabilité CoopFlow pour préserver le savoir faire des directions métier de l'entreprise et assurer leur autonomie avec une gestion efficace et flexible des services.

II.2. Agent et Système Multi-Agents

Les systèmes multi-agents (SMA) présentent une solution technique et architecturale efficace pour l'implémentation du WIO. En effet, un rapprochement conceptuel se trouve entre les deux concepts en matière d'autonomie et distribution.

Dans cette deuxième partie du chapitre nous présentons les concepts de base de la technologie agent et plus particulièrement les principales techniques de coordination dans les SMA utiles pour la modélisation du WIO lâche de gouvernance SI.

Dans cette perspective, la technologie Agent peut être déployée à deux niveaux : le niveau micro et le niveau macro (Chergui, Sayouti, & Medromi, 2014) :

- *Au niveau micro*, la technologie agent permet d'encapsuler chaque composant d'un système d'information distribué et coopératif comme une entité responsable d'une mission qu'elle est capable d'accomplir de manière autonome et en coordination avec d'autres agents. Elle permet à chaque entité par le biais d'un mécanisme de décider par elle-même *quand, comment et pourquoi* un service doit être rendu.
- *Au niveau macro*, la technologie agent représente un outil puissant pour construire des systèmes modulaires et coopératifs. Face à un environnement complexe, la seule façon de le traiter d'une manière raisonnable est de le décomposer en un certain nombre de composants intelligents. Dans le cas où ces composants sont de plus en plus interdépendants, la technologie agent apporte les outils pour les modéliser en les faisant interagir et coopérer.

Elle apporte notamment des langages de communication de hauts niveaux, des protocoles et des abstractions organisationnelles permettant de concevoir des systèmes comme des sociétés d'agents. La suite de ce chapitre est organisée de la manière suivante.

II.2.1. Terminologie autour des systèmes multi-agents

II.2.1.1. Agent

L'intelligence artificielle distribuée (IAD) est née dans les années 70 pour remédier aux limitations de l'IA classique est via la distribution de l'intelligence sur un groupe d'entités et la modélisation de comportements intelligents produits de coopération entre ces dernières. La première apparition des agents autonomes tels qu'ils sont conçus aujourd'hui est en 1987 par Minsky (Minsky, 1979). Plusieurs définitions du concept d'agents ont apparus depuis dans la littérature. Stan Franklin et Arthur Graesser ont évoqué les différentes définitions des agents et ont différencié entre un agent et un programme classique (Moutaouakkil, 2010).

Nous donnons ici la définition d'un agent la plus utilisée (Benhadou, 2010) :

«Un agent est une entité autonome, réelle ou abstraite, qui est capable d'agir sur elle-même et sur son environnement, qui, dans un univers multi-agents, peut communiquer avec d'autres agents, et dont le comportement est la conséquence de ses observations, de ses connaissances et des interactions avec les autres agents»

A partir de cette définition, un agent est :

- Tout ce qui perçoit son environnement à l'aide de ses capteurs et qui agit sur son environnement à l'aide de ses effecteurs.
- Une entité réelle ou virtuelle plongée dans un environnement sur lequel elle est capable d'agir et qui:
 - dispose d'une capacité de perception et de représentation partielle de cet environnement
 - peut communiquer avec d'autres agents
 - est mue par un ensemble de tendances (objectifs individuels, fonctions de satisfaction, de survie)

- possède un comportement autonome tendant à satisfaire ses objectifs, conséquence de ses observations, de sa connaissance, et des interactions qu'elle entretient avec les autres agents
- est capable éventuellement de se reproduire.
- Autonome : être capable d'agir sans l'intervention directe d'un tiers (humain ou agent), de contrôler ses actions et ainsi que son état interne en fonction d'informations en provenance de l'environnement. Alors qu'un objet répond aux invocations de méthodes, on considère qu'un agent rend des services et possède une procédure de décision qui lui permet de décider et négocier quand, comment et à qui rendre un service (en fonction d'objectifs) ;
- Capable de percevoir les différentes situations d'interaction,
- Capable de raisonner sur les actions possibles et d'agir en conséquence.
- Communiquant avec d'autres agents ainsi qu'avec des utilisateurs humains
- Une entité ayant des aptitudes sociales pour interagir avec les autres agents de façon coopérative ou compétitive pour atteindre ses objectifs ;
- Est pro-activité en se fixant des buts pour atteindre ses objectifs

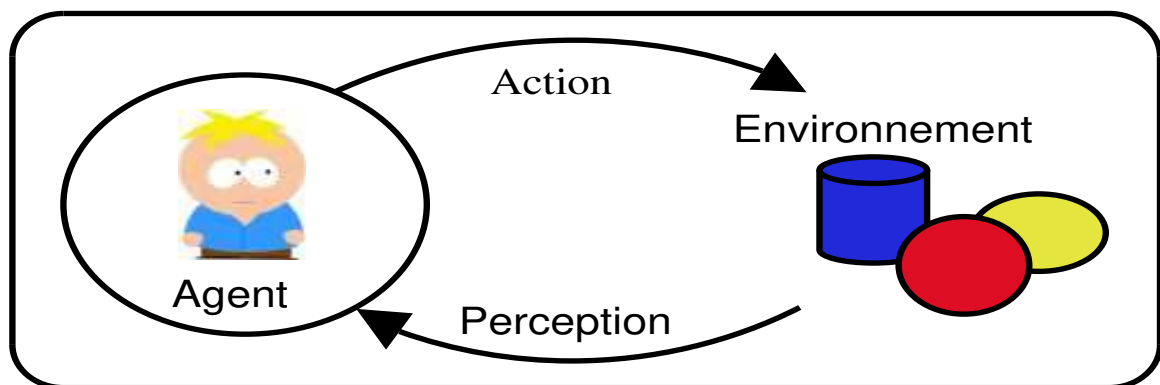


Figure 14: Interaction d'un agent avec son environnement

II.2.1.2. Agent Versus Objet

Toutes les propriétés citées dans la section précédente valorisent un agent par rapport à un objet. En effet, un agent est une entité autonome interagissant avec son environnement, l'objet quant à lui est une entité passive possédant un état et sur lequel on peut effectuer des opérations.

Le tableau ci-dessous montre plus en détail la différence entre un objet ou un agent ou plus précisément entre la programmation orienté objet par rapport à celle orientée agent qui a vu le jour comme étant une évolution de la programmation orientée-objet (Sayouti, Qrichi Aniba, & Medromi, 2009) :

Critères	Objet	Agent
Paramètres de définition	Libre	Croyance, décision, obligation habilité.
Type de message	Libre	Informé, demander, offrir, accepter, rejeter...
Langages de communication	Envoi de message	Actes de langages
Mécanisme d'interaction	A la charge du programmeur	Fourni
Raisonnement	Calculatoire	Intentional

Tableau 2.4: Différences entre un Objet et un Agent

Nous déduisons donc qu'un agent est à un degré d'abstraction plus élevé qu'un objet d'où l'importance de l'utilisation de la technologie agent, comme un nouveau paradigme de programmation basé sur une « vue sociale ».

II.2.1.3. Systèmes multi-agents

Un Système Multi-Agent (SMA) est défini comme : «un système distribué composé d'un ensemble d'agents interagissant les uns avec les autres» (Sayouti, Moutaouakkil, & Medromi, The Interaction-Oriented Approach for Modeling and Implementing Multi-Agents Systems, 2012).

Pour mieux définir les SMA, nous suivons la décomposition Voyelles AEIO (Agent, Environnement, Interactions, Organisation) d'Yves Demazeau (Boissier & Demazeau, 1992) :

- a) Agent : section précédente.
- b) Environnement (Ferguson, 1992).

L'environnement est le milieu dans lequel les agents sont définis. Il permet la communication entre agents, supporte leurs actions en définissant les règles et les contraintes, est observable partiellement ou totalement par les agents et prend en charge l'activité propre des objets et des ressources présentes. Lorsque les agents sont réactifs, l'environnement détient une importance capitale car il est le médiateur de leurs interactions, vu les limitations qu'ils présentent.

Un environnement peut être :

- *Accessible* si un agent peut, grâce à sa perception, déterminer l'état de l'environnement et ainsi procéder.
- *Inaccessible* si l'agent est doté de moyens de mémorisation afin d'enregistrer les modifications qui sont intervenues.
- *Déterministe*, ou non, selon l'état futur de l'environnement s'il est fixé ou variable par rapport à son état courant et les actions de ses agents.
- *Episodique* si le prochain état de l'environnement ne dépend pas des actions réalisées par les agents.
- *Statique* si l'état de l'environnement est stable (ne change pas) pendant que l'agent prend des décisions. Dans le cas contraire, il est dit *dynamique*.
- *Discret* si le nombre des actions faisables et des états de l'environnement sont finis et définis.

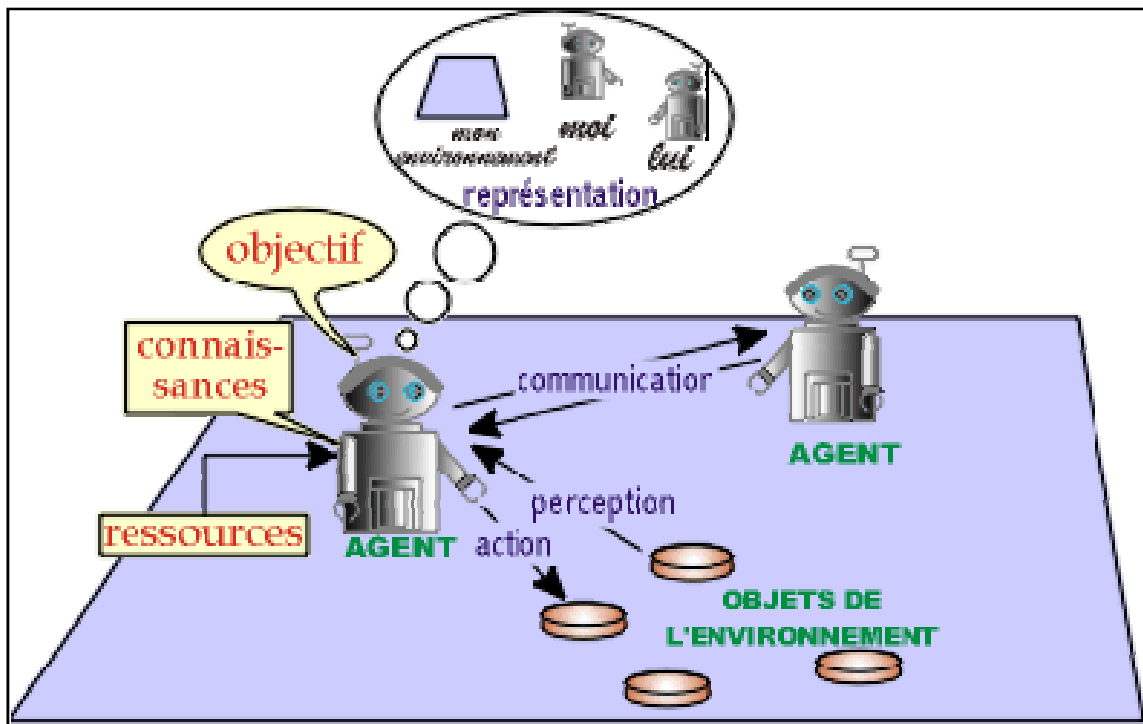


Figure 15: Système multi-agents

c) Interaction

La définition des SMA met l'accent sur un élément essentiel et fondateur des systèmes multi-agents, qui est l'interaction. C'est une mise en relation dynamique entre agents via des actions réciproques permettant le transfert d'informations entre agents ou entre environnement et les agents, à l'aide de deux mécanismes : perception et communication.

Nous détaillons dans une partie suivante les différentes formes d'interaction dans les SMA.

II.2.2. Problématiques et apport des systèmes multi-agents

Certaines problématiques dépassent la modélisation objet pour une éventuelle résolution et requièrent en contre partie l'utilisation de plusieurs entités autonomes et indépendantes. Les systèmes multi-agents sont dans ce cas très bien adaptés à ce type de situations. Parmi les utilisations les plus fréquentes, on cite :

- **Systèmes géographiquement distribués** : la coordination entre plusieurs frégates
- **Systèmes de production** : ordonnancement d'ateliers, conduite de processus industriels, systèmes multi-capteurs.
- **Diagnostic** : diagnostic à multiples niveaux (diagnostic médical, mécanique ou autres).
- **Taches de contrôle** : contrôle du trafic routier, trafic aérien, distribution d'énergies.
- **Taches d'interprétation** : interprétation de signaux, reconnaissance de la parole, cristallographie, reconnaissance et compréhension des formes.
- **Télécommunications, systèmes de transports, réseaux** : routage, équilibrage de charges, recouvrement d'erreurs, management et surveillance de réseaux.
- **Travail collaboratif assisté par ordinateur** : agents assistants, agents médiateurs, gestion des rendez-vous, personnel digital assistants (PDA).
- **Robotique distribuée** : planification multi-robot, robots autonomes mobiles.

- **Télématique (Internet)** : agents "intelligents", agents d'interface, agents mobiles.
- **Simulation de systèmes complexes** : simulation individu-centrée.
- **Commerce électronique** : organisations virtuelles.

Les systèmes multi-agents sont aussi requis, lorsque les différents systèmes et les données qui s'y rattachent appartiennent à des organisations indépendantes qui veulent garder leurs informations privées et sécurisées tout en répondant à un problème global (le cas des WIO). Une solution à ce problème est de permettre à chaque sous système de concevoir ses propres agents qui représenteront adéquatement ses buts et ses intérêts. Par la suite, ces agents pourront communiquer avec l'ensemble pour répondre à l'objectif global. Dans ce cas, les agents ne transmettront que les informations nécessaires à une bonne coordination du système global sans pour autant partager les processus propres à leur sous système.

Ainsi, les SMA s'avèrent utiles pour des problèmes possédant de multiples méthodes de résolution, de multiples perspectives ainsi que ceux simulant le raisonnement humain.

Ils disposent également des avantages traditionnels de la résolution distribuée et concurrente de problèmes :

- La modularité : permet la simplification des programmes et l'extensibilité et la réutilisabilité des SMA face aux systèmes monolithiques. D'autant plus que les applications modulaires à base d'agent ont un autre avantage qui est plus important que la distribution des tâches lors du codage, il s'agit de la maintenance et les mises à jour en général, si on découvre une anomalie dans une fonctionnalité, il suffit de corriger le module d'où elle parvient, et de recompiler le module en question sans affecter la totalité des parties fonctionnelles
- La vitesse : est principalement due au parallélisme, car plusieurs agents peuvent travailler en même temps pour la résolution d'un problème en s'exécutant sur un même processeur.
- La fiabilité : due au partage de responsabilités entre les différents agents, le système peut dépasser la défaillance d'un ou de plusieurs agents. Si une seule entité contrôle tout, alors une seule défaillance de cette entité fera en sorte que tout le système tombera en panne.

Enfin, les systèmes multi-agents héritent aussi des bénéfices envisageables du domaine de l'intelligence artificielle et l'intelligence artificielle distribuée à savoir :

- le traitement symbolique des connaissances,
- la facilité de maintenance,
- la réutilisation
- la portabilité.

Le SMA s'intéresse donc aux comportements intelligents et collectifs produits par les interactions de plusieurs entités de résolution de problème. Que ces interactions tournent autour de la coopération, de la négociation ou de la communication. Pour se faire la technologie agent déploie plusieurs types et catégories d'agent selon le problème et l'environnement. Nous présentons dans la partie suivante ses différents types.

II.2.3. Catégories d'agents

Les agents peuvent être classés selon leurs degrés d'autonomie, de coopération et d'adaptation. En effet, quand l'agent a un but à atteindre, il doit être suffisamment autonome pour pouvoir prendre des

initiatives permettant d'atteindre ce but. D'autant plus que pour l'ensemble des agents constituant un système cohérent, chacun d'entre eux doit avoir un degré de coopération, et doit agir en fonction de son environnement en s'y adaptant (Moutaouakkil, 2010).

Le plus haut niveau d'autonomie permet à l'agent de planifier ses actions, le plus haut niveau de coopération accorde à l'agent des capacités de négociation, le plus haut degré d'adaptation permet à l'agent d'adapter et d'acquérir des connaissances.

II.2.3.1. Agents réactif

Les agents réactifs ne sont pas « intelligents » pris individuellement. Ils ne peuvent que réagir à des stimuli simples provenant de leur environnement, et leur comportement est alors simplement dicté par leur relation à leur entourage sans que ces agents ne disposent d'une représentation des autres agents ou de leur environnement. Cependant, du fait, de leur nombre, ces agents réactifs peuvent résoudre des problèmes qualifiés de complexes.

Les travaux sur ces agents s'intéressent plus à la modélisation d'une société d'agents qu'à l'agent lui-même. Les analogies que les chercheurs ont établies sont celles de la vie artificielle, de l'éthologie (la fourmilière, la termitière, la ruche d'abeille), etc.

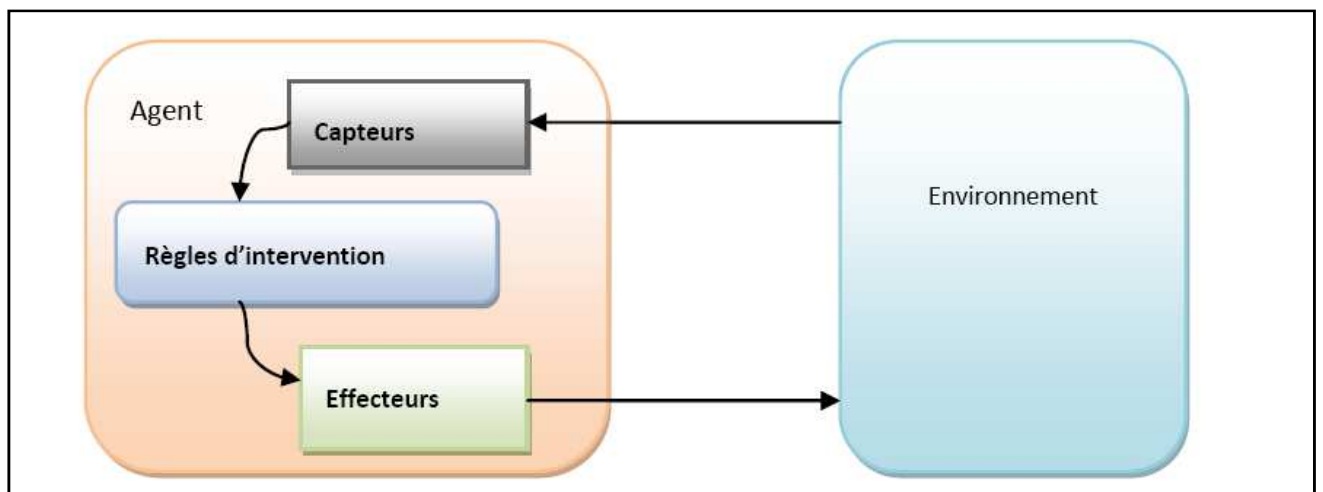


Figure 16: Architecture d'agent réactif

II.2.3.2. Agent cognitif

Les agents cognitifs disposent d'une base de connaissances comprenant les diverses informations liées à leurs domaines d'expertise et à la gestion des interactions avec les autres agents et leur environnement. Les agents cognitifs sont généralement « intentionnels » c'est-à-dire qu'ils possèdent des objectifs et des plans explicites leur permettant d'accomplir leurs buts. Dans ce cadre, comme le précise J. Ferber, les problèmes de coopération ressemblent étonnamment à ceux de petits groupes d'individus, qui doivent coordonner leur activité, et sont parfois amenés à négocier pour résoudre leurs conflits. Dans la littérature des agents, l'architecture d'un agent cognitif n'est autre que l'architecture BDI. En fait, un agent cognitif ou BDI doit mettre à jour ses croyances avec les informations qui lui proviennent de son environnement. Il doit décider quelles options lui sont

offertes, filtrer ces options afin de déterminer de nouvelles intentions et poser ses actions au vu de ses intentions (voir section suivante).

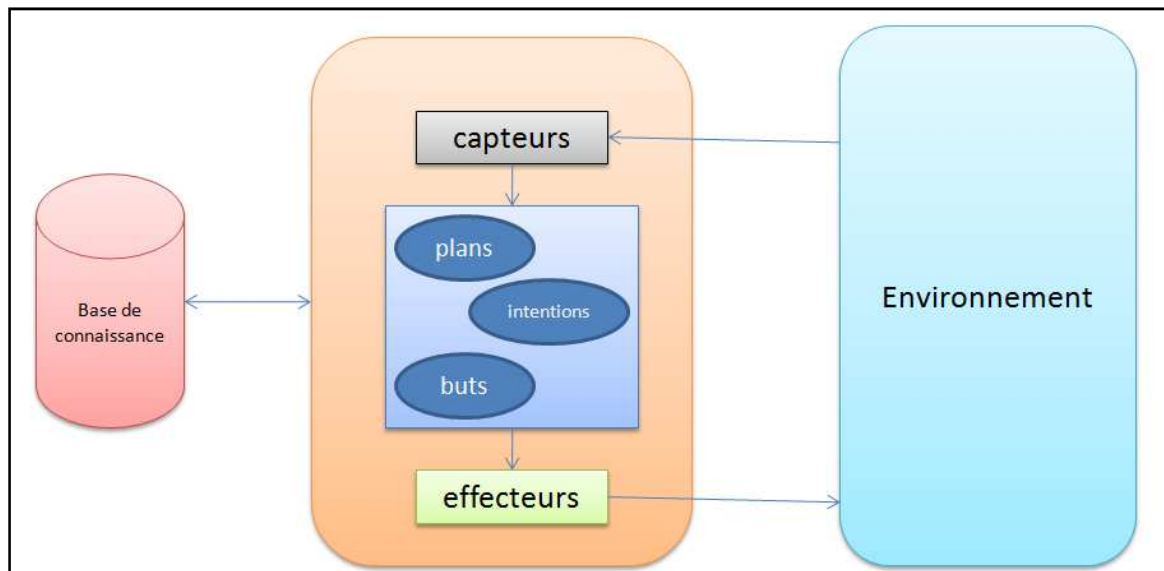


Figure 17: Architecture d'agent cognitif

II.2.3.3. Agent hybride

En général, la différence entre des agents réactifs et des agents cognitifs peut être expliquée par le compromis efficacité/ complexité. La complexité des systèmes réactifs exige le développement de nouvelles théories dans le domaine de la coopération, de la communication et de la compréhension de nouveaux phénomènes telles que l'émergence.

Toutefois, il est maintenant possible de concevoir des systèmes hétérogènes comportant les deux types de comportements (cognitif et réactif) : on parlera alors d'agents hybrides.

Dans ce sens, nous pouvons citer les travaux de [(Ferguson, 1992) ; (Müller, Pischel, & Thiel, 1994); (Bussmann & Demazeau, 1994)]. La majorité des modèles d'agents hybrides présentés par ces auteurs, propose de décomposer chaque agent en différents modules réactifs et cognitifs avec un module spécifique qui contrôle l'activation des autres modules. Cette approche est intéressante et semble apporter une solution adéquate pour modéliser les systèmes complexes dont l'environnement est dynamique, mais elle ne résout pas clairement le problème d'interaction entre les différents modules. Le problème, pour M.J. Wooldridge et N.R. Jennings (Jennings, Sycara, & Wooldridge, 1998), est alors de définir les mécanismes et les stratégies du module de contrôle interne de l'agent pour soit gérer les interactions entre ses différents modules, soit imposer un séquençement temporel global interne à l'agent.

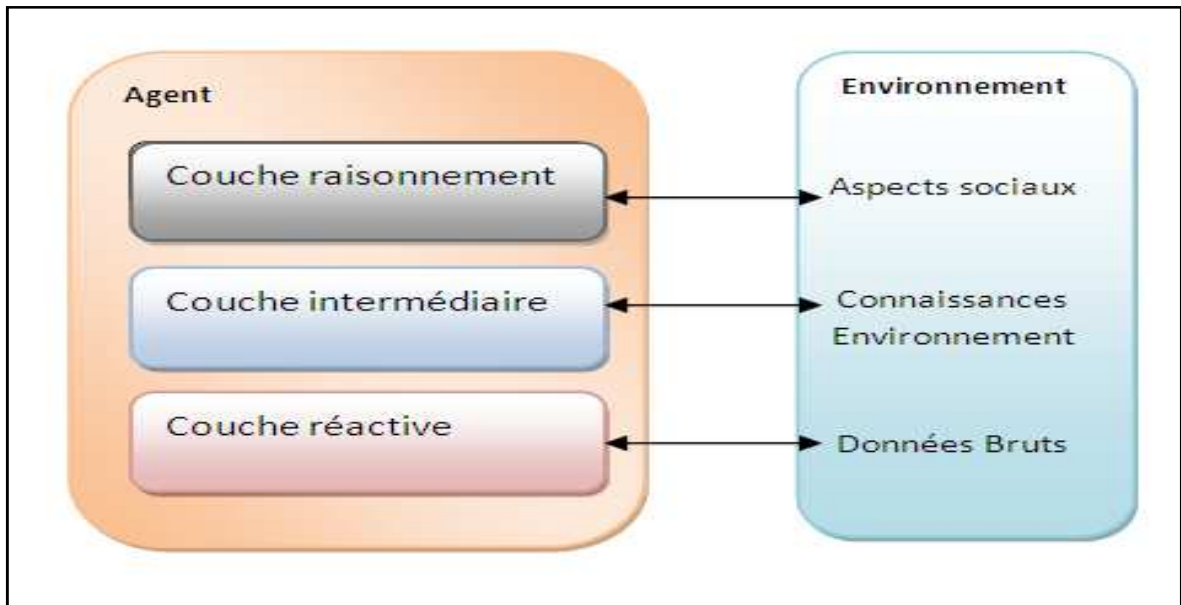


Figure 18: Architecture d'un agent hybride

II.2.4. Architecture et Méta-modèle d'agents intelligents

II.2.4.1. Belief Desire Intention - BDI

Une architecture BDI se base sur le modèle "Croyance-Désir-Intention" (Moussaoui, 2015), en anglais "Belief-Desire-Intention", synonyme de la rationalité d'un agent.

a) B /Belief / Croyance

Les croyances d'un agent sont les informations que l'agent possède sur l'environnement et sur d'autres agents qui coexistent avec lui. Les croyances peuvent être incorrectes, incomplètes ou incertaines ; c'est la raison pour laquelle elles sont différentes des connaissances, qui quant à elles sont des informations toujours vraies. Les croyances peuvent changer au fur et à mesure que l'agent, par sa capacité de perception ou par l'interaction avec d'autres agents, recueille plus d'information.

b) D /Desire /Désir

Les désirs d'un agent représentent les états de l'environnement, et parfois de lui-même, que l'agent aimerait voir réalisés. Un agent peut avoir des désirs contradictoires ; dans ce cas, il doit choisir parmi ses désirs un sous-ensemble qui soit consistant. Ce sous-ensemble consistant de ses désirs est parfois identifié avec les buts de l'agent.

c) I /Intention / Intention

Les intentions d'un agent sont les désirs que l'agent a décidé d'accomplir ou les actions qu'il a décidé de faire pour accomplir ses désirs. Même si tous les désirs d'un agent sont consistants, l'agent peut ne pas être capable de les accomplir tous à la fois.

Les plans dans le modèle BDI sont des connaissances qui concernent l'ensemble des actions qu'il faut entreprendre pour atteindre un but (désir). Le plan effectivement choisi et mis en œuvre correspond à une intention.

La théorie BDI de l'action rationnelle, a été proposée pour la première fois par Michael Bratman, comme étant une théorie du raisonnement pratique qui essaie de surprendre comment les gens raisonnent dans la vie de tous les jours, en décidant, à chaque moment, ce qu'ils ont à faire. En développant sa théorie, Bratman montre que les intentions jouent un rôle fondamental dans le raisonnement pratique, car elles limitent les choix possibles qu'un humain (ou un agent artificiel) peut faire à un certain moment.

Le schéma classique d'une architecture BDI est le suivant.

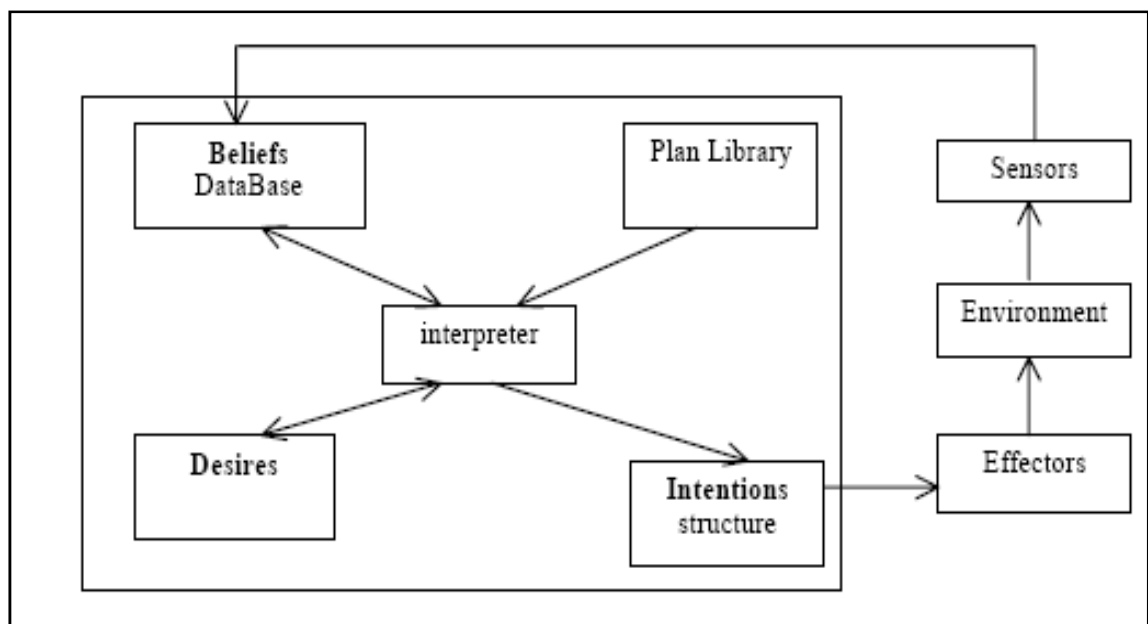


Figure 19: Architecture Belief-Desire-Intention

L'architecture BDI présente plusieurs avantages à savoir:

- L'appui sur une théorie connue et appréciée de l'action rationnelle des humains ;
- La formalisation en une logique symbolique formelle, rigoureuse ;
- L'implémentation réussie dans plusieurs modèles d'agent exemple : Intelligent Resource-bounded Machine Architecture (Georgeff, Pell, Pollack, Tambe, & Wooldridge, 1998), Procedural Reasoning System (Huber, 1999).

II.2.4.2. Agent Groupe Rôle - AGR

C'est un méta modèle de base de la littérature orienté agent qui définit la dimension organisationnelle des SMA (Ferber & Gutknecht, 1998). AGR repose sur trois concepts primitifs, Agent, Groupe et Rôle qui sont structurellement reliés et ne peut pas être défini par d'autres primitives. Il satisfait à un ensemble d'axiomes qui unissent ces concepts

Il considère l'agent comme une entité active et communicante qui peut jouer un ou plusieurs rôles à travers son appartenance à un ou plusieurs groupes sans aucune contrainte sur son architecture.

Un groupe est un ensemble d'agents ayant des caractéristiques communes, utilisé comme un modèle d'activité. Deux agents ne peuvent communiquer que s'ils appartiennent au même groupe.

Un rôle est la représentation abstraite de l'activité d'un agent dans un groupe ; il peut être joué par plusieurs agents, Il est propre au groupe et est demandé par l'agent qui le joue.

Les axiomes du méta-modèle AGR se résument comme suit :

- Un agent doit obligatoirement appartenir à au moins un groupe ;
- Deux agents peuvent communiquer s'ils appartiennent au même groupe ;
- Un agent joue au moins un rôle dans un groupe ;
- Un agent est membre du groupe au sein duquel il joue un rôle;
- Un rôle est défini dans la structure d'un groupe.

D'autant plus, deux contraintes structurelles sont proposées : la correspondance et la dépendance. Une contrainte de correspondance indique que les agents qui jouent un rôle automatiquement jouent un autre rôle. Par exemple, pour exprimer la correspondance classique entre les délégués des groupes plus petits (états, départements, régions) qui sont automatiquement membres d'un autre groupe où ils agissent en tant que représentant (députés, ambassadeur, etc.) . Les contraintes de dépendance quant à elles, expriment les dépendances entre les membres du groupe et le jeu de rôle.

Le diagramme UML ci-dessous montre la liaison entre les différents concepts du modèle :

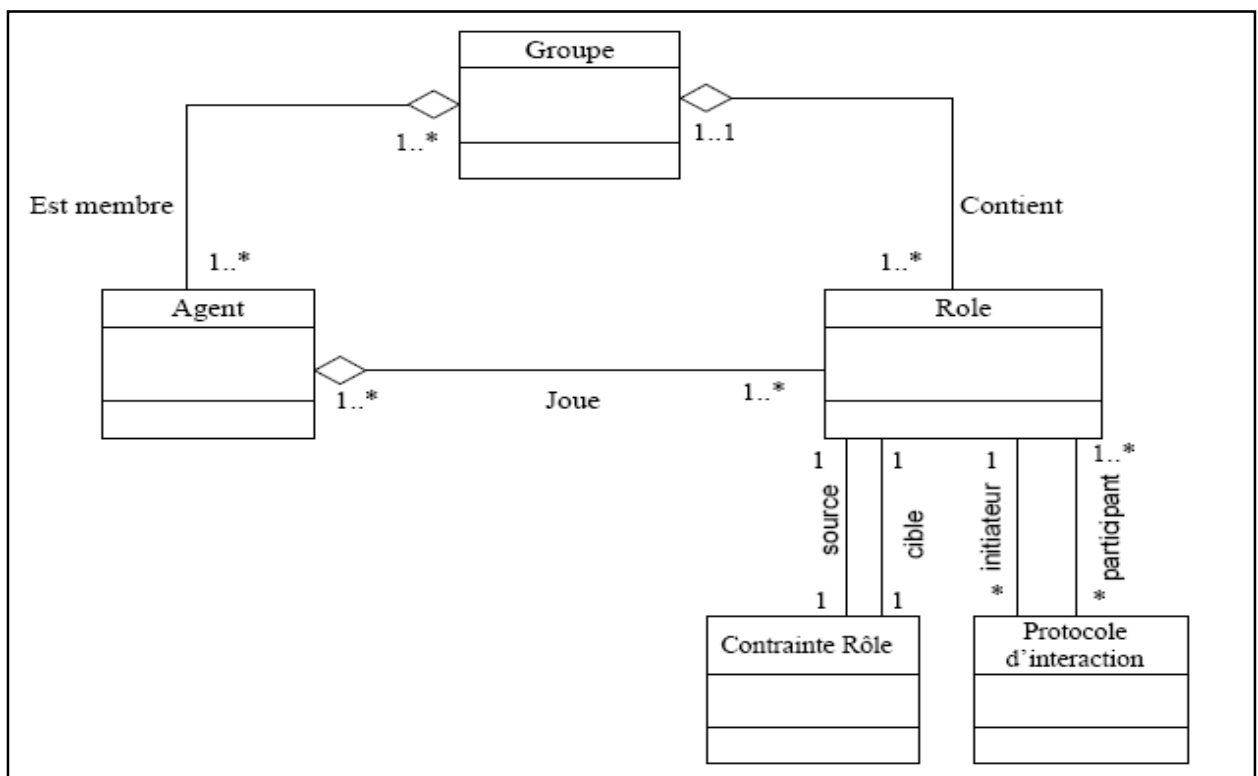


Figure 20: Diagramme UML du Méta-Modèle AGR

Le méta-modèle AGR présente de nombreux avantages :

- Il intègre les concepts clefs d'agent, de rôle et de groupe ;
- Il préserve l'autonomie des agents en fixant uniquement le rôle que l'agent peut jouer d'une manière abstraite, sans conditionner la procédure et les actions internes, pour jouer ce rôle ;
- Il permet l'hétérogénéité des agents puisqu'il n'impose aucune contrainte sur la structure interne des agents.

Nous optons pour le modèle AGR pour l'organisation des agents du WIO de Gouvernance IT et de l'architecture BDI pour les agents cognitifs de la plateforme. Le lecteur intéressé par le comparatif des architectures des SMA doit se référer à (Wooldridge & Jennings, 1994) et celui intéressé par les différents méta-modèles doit lire (Gherbi, Borne, & Meslati, 2013).

II.2.5. Interaction entre Agents et Environnement

II.2.5.1. Communication

La communication, dans les systèmes multi-agents, est à la base des interactions et de l'organisation. Si les agents peuvent coopérer, coordonner leurs actions, réaliser des tâches en commun c'est parce qu'ils communiquent. Deux modes de communication sont identifiés (Huget & Koning, 2001):

- la communication indirecte via l'environnement ou le partage d'information.
- la communication directe par passage de message.

Tout message repose sur un langage de communication. Dans ce contexte, la théorie linguistique des actes du discours offre des concepts de haut niveau pour définir la communication entre agents. Elle s'attache moins à la syntaxe qu'au pourquoi et au comment de la communication, qu'elle met en évidence dans l'usage performatif du langage : même si les conversations sont habituellement considérées comme des échanges d'informations, la communication n'est pas seulement informative mais aussi performatives, c'est-à-dire qu'elle constitue *un acte*, dans la mesure où elles servent à produire des effets sur le destinataire.

Un acte de langage est donc défini par trois composantes :

- locutoire : reliée à la formulation physique de la communication,
- illocutoire : reliée à l'intention de l'émetteur
- perlocutoire : reliée à l'effet sur le destinataire.

L'intention d'un message est représentée sous la forme F(P) où F est un performatif, appelé force illocutoire, représenté par un verbe (ex. affirmer, questionner, demander, ...), et P est le contenu associé. Ainsi la même information peut être transmise avec des intentions différentes : affirmer (un passager est dans le transport), questionner (un passager est-il dans le transport ?)

L'usage des performatifs facilite la conception des agents car elle structure leur comportement en distinguant dans une communication le message de son intention, et donc respectivement l'information de l'activité à mettre en œuvre pour émettre et interpréter cette information.

Plusieurs tentatives de normalisation de la communication inter-agents ont été effectuées au sein de la communauté multi-agents ces dernières années. Les principales propositions sont :

KQML et ACL.

a) KQML :

Le langage KQML (Finin & Fritzson, 1994) a été proposé pour supporter la communication inter-agents. Ce langage définit un ensemble de types de messages (appelés “performatifs”) et des règles qui définissent les comportements suggérés pour les agents qui reçoivent ces messages.

Les types de messages de KQML sont de natures diverses :

- simples requêtes et assertions (ex. “ask”, “tell”);
- instructions de routage de l’information (“forward” et “broadcast”);
- commandes persistantes (“subscribe”, “monitor”);
- commandes qui permettent aux agents consommateurs de demander à des agents intermédiaires de trouver les agents fournisseurs pertinents (“advertise”, “recommend”, “recruit” and “broker”).

Selon Cohen et Levesque (Wiederhold, 1992), ce langage a été développé de façon ad-hoc pour les besoins des développeurs d’agents logiciels : le terme "performatif" a été utilisé pour nommer diverses commandes qui ont une certaine ressemblance avec des verbes utilisés de façon performative dans le langage naturel.

KQML est un langage à trois couches :



Figure 21: Représentation en couches de KQML

- Couche contenu : les agents doivent s'accorder sur le langage de contenu (KRSL, KIF, LOOM, prolog, Clips) et l'ontologie à utiliser.
- Couche message: codage du message qu'un agent souhaite transmettre constitué de données, de déclarations (type d'information échangée, type d'acte du langage (assertion, requête, réponse, message d'erreur)).
- Couche communication: échange de "packages" - enveloppe autour du message spécifiant les agents expéditeur et destinataire(s).

Un message KQML a en général le format suivant :

```
<performative>  
sender <expéditeur>  
receiver <destinataire>  
in-reply-to < label attendu dans une réponse à un précédent message >  
ontology <nom-de-l'ontologie utilisée>  
language <langage du contenu>  
content <contenu>
```

A ce niveau là, il faut dire un mot sur les performatives KQML du moment qu'il est l'ancêtre des langages de communication des agents. Il considère tout agent comme gérant d'une BD, du coup une variété de performatives se présente comme le résumé le schéma ci-dessous :

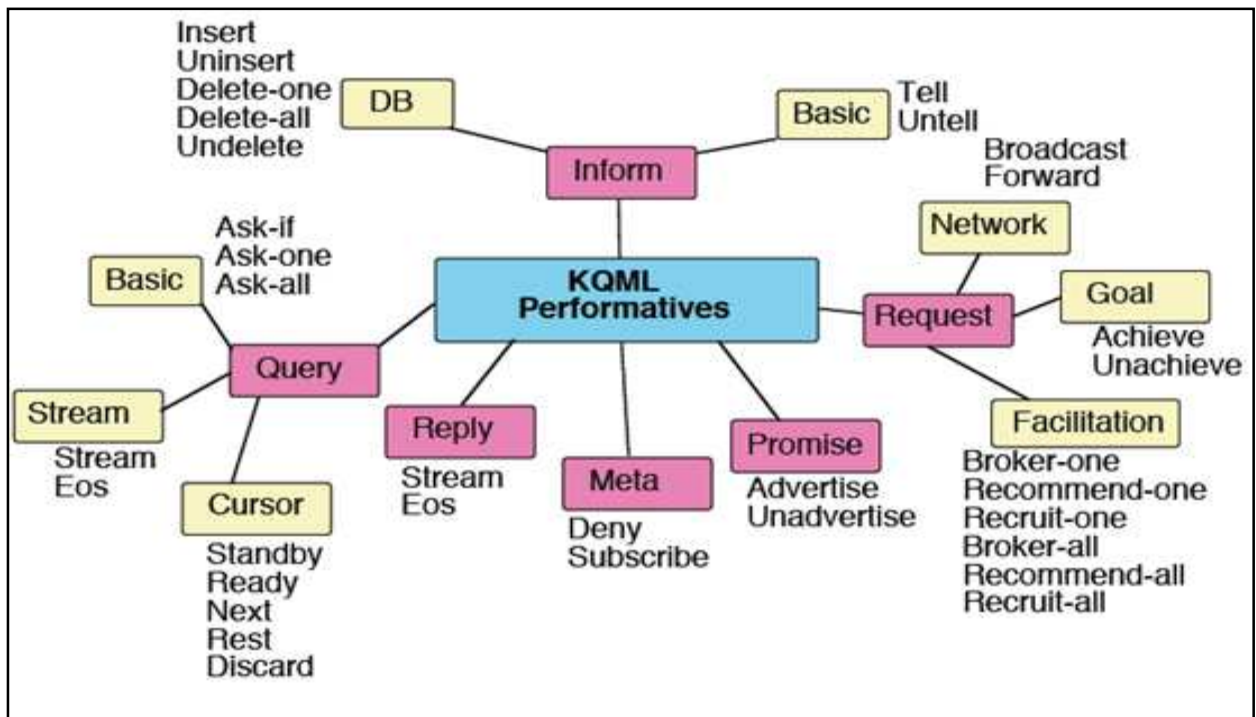


Figure 22: Performatives KQML

Un dernier point à évoquer en parlant de KQML c'est la notion de facilitateur qui est une classe spécifique d'agents distribuant des méta-informations sur les autres agents et offrant des services de communication tels que (la retransmission et distribution des messages ; la découverte des ressources ; le routage basé sur le contenu du message). On reviendra en détail dans le chapitre de médiation à cette notion.

b) ACL

Ces dernières années, KQML a été amélioré et enrichi en un langage sémantiquement plus riche ACL (Agent Communication Language). Un langage mis en place par la FIPA (Foundation of Intelligent Physical Agents), organisation destinée à établir des standards pour favoriser l'interopérabilité des applications, des services et des équipements informatiques à base de système multi-agents.

Dans cette perspective, la théorie du langage a bénéficié largement des résultats de la recherche de KQML. Si toutefois, les deux langages se rapprochent au niveau des actes du langage, il n'en ait rien au niveau de la sémantique et il semble qu'un grand soin a été apporté au niveau de ACL tant au niveau de certains protocoles qui sont plus explicites qu'au niveau de la sémantique des actes eux mêmes.

ACL se base sur deux notions de base :

- Les actes de communication primitifs, en plus des actes de communication obtenus par la composition des actes de base (performative KQML)
- Les messages prédéfinis que tous les agents peuvent comprendre

Pour les actes de communication, ils sont représentés par des performatives regroupées en plusieurs catégories à savoir :

- **Passage d'informations** : inform, inform-if, inform-ref , confirm, disconfirm
- **Réquisition d'information** : query-if, query-ref, subscribe
- **Négociation** : accept-proposal, propose, reject-proposal
- **Exécution de tâches**: request, request when, request whenever, agree, cancel, refuse
- **Gestion des erreurs** : failure, not-understood

Le format du message ACL est similaire au message KQML à des détails supplémentaires près :

performative	type de l'acte communicatif
sender	l'émetteur du message
receiver	le destinataire du message
reply-to	participant à l'acte de communication
content	le contenu du message (l'information transportée par la performative)
language	le langage dans lequel le contenu est représenté
encoding	décrit le mode d'encodage du contenu du message
ontology	le nom de l'ontologie utilisé pour donner un sens aux termes utilisés dans le contenu
protocol	contrôle la conversation
conversation-id	identificateur de la conversation
reply-with	identificateur unique du message, en vue d'une référence ultérieure
in-reply-to	référence à un message auquel l'agent est entrain de répondre (précisé par l'attribut reply-with de l'émetteur)
reply-by	impose un délai pour la réponse

Figure 23: format de message ACL

II.2.5.2. Coopération

La coopération est une caractéristique très importante dans les SMA. En effet, une résolution distribuée d'un problème est le résultat de l'interaction coopérative entre les différents agents. Ferber (Ferber & Gutknecht, 1998) énumère trois situations de coopération selon les valeurs des deux paramètres suivants:

- Les compétences de chaque agent qui peuvent être suffisantes ou insuffisantes ;
- Les ressources nécessaires à chaque agent qui peuvent être suffisantes ou insuffisantes.

Le tableau 2.1 ci-dessous identifie ces situations de coopération selon les valeurs de ces deux paramètres. Nous avons écarté la solution qui considère que les agents ont des compétences suffisantes et des ressources suffisantes car dans ce cas, ils n'ont pas besoin de coopérer. Ce tableau est extrait de celui proposé par Ferber, qui quant à lui généralise ce tableau aux différentes situations d'interaction entre agents.

Buts	Ressources	Compétences	Types de situation	Catégorie
Compatibles	Suffisantes	Suffisantes	Indépendance	Indifférence
Compatibles	Suffisantes	Insuffisantes	Collaboration simple	Coopération
Compatibles	Insuffisantes	Suffisantes	Encombrement	Coopération
Compatibles	Insuffisantes	Insuffisantes	Collaboration coordonnée	Coopération

Tableau 2.5: Types d'interaction SMA

En effet, pour établir la coopération dans un groupe d'agents quatre buts essentiels doivent être mis en place :

- Augmenter le taux de réalisation des tâches via le parallélisme;
- Augmenter le nombre de tâches grâce au partage de ressources ;
- Augmenter les chances de finaliser des tâches en les dupliquant et via des modes de réalisation différents;
- Diminuer les interférences entre tâches en évitant les interactions antagonistes.

En fait, on peut caractériser un système par le type de coopération mis en œuvre qui peut varier selon le tableau ci-dessus de la coopération totale à l'antagonisme total.

Des agents totalement coopératifs peuvent changer leurs buts pour répondre aux besoins des autres agents afin d'assurer une meilleure coordination entre eux. Cela peut résulter en des coûts de communication élevés.

Les agents antagonistes quant à eux, ne coopèrent pas et dans ce cas, leurs buts respectifs vont se bloquer les uns les autres. Dans de tels systèmes, les coûts de communication sont minimes. La plupart des systèmes réels se situent entre les deux extrémités : coopération totale et antagonisme total.

La coopération totale est mise en œuvre dans les approches de résolution coopérative de problèmes distribués où les agents coopèrent pour résoudre des problèmes qu'ils ne peuvent pas résoudre individuellement via les étapes suivantes :

1. Chaque agent utilise ses connaissances et ressources pour résoudre localement un ou plusieurs sous-problèmes.
2. Les solutions partielles sont intégrées aux sous-problèmes.

La coopération est donc une attitude adoptée par les agents qui décident de travailler ensemble ou on peut adopter le point de vue d'un observateur extérieur au système multi-agent qui interprète a posteriori les comportements des agents pour les qualifier de coopératifs ou non suivant des critères préétablis tels que l'interdépendance des actions ou le nombre de communications effectuées.

Dans le cas de la coopération vue comme une attitude intentionnelle, les agents s'engagent dans une action après avoir identifié et adopté un but commun. C'est par exemple le cas d'un groupe de

personnes acceptant de travailler dans une entreprise : l'adoption d'un but commun est donc coopération qu'ils considèrent comme un élément essentiel pour l'activité sociale.

II.2.5.3. Coordination

La coordination a une fonction cruciale qui détermine l'utilité des SMA : *garantir le comportement cohérent et efficace des agents du système*. Elle revêt deux aspects essentiels :

- Le premier aspect est de rendre possible la réalisation de l'objectif global du système, en assurant le respect des contraintes sans lesquelles il ne peut être rendu. En effet, un SMA étant constitué d'agents autonomes, il est fort probable que, sans coordination, le comportement du système devienne chaotique, c'est-à-dire que les agents vont s'ignorer, se gêner, avoir des comportements redondants ou du moins ne pas agir en synergie, ce qui ne permet pas d'atteindre l'objectif assigné au système. De ce point de vue, la coordination peut être considérée comme une contrepartie de l'autonomie, visant à contrer le désordre naturel.
- Le deuxième aspect de la coordination vise à optimiser le fonctionnement global du système, c'est-à-dire à le rendre plus efficace en minimisant la quantité de ressources (en coût, en temps de calcul) utilisée, en favorisant les actions qui facilitent, les synergies, et en limitant les actions redondantes.

Pour rendre la coordination entre agents possible, plusieurs techniques de coordination dans les SMA ont été proposées.

De nombreux exemples de coordination existent dans la vie quotidienne : deux personnes creusant un puit, les joueurs de foot de la même équipe, etc.

- Il est à noter que deux des composantes fondamentales de la coordination entre agents sont :
- l'allocation de ressources rares : les agents doivent être capables de faire des transferts de ressources
- la communication de résultats intermédiaires : les agents doivent être capables de communiquer entre eux de façon à pouvoir échanger les résultats intermédiaires.

Ceci peut d'ailleurs imposer certains comportements à des agents particuliers.

En étudiant les communautés humaines, trois processus fondamentaux de coordination sont identifiés (Mintzberg, 1979):

- Ajustement mutuel : forme de coordination la plus simple qui se produit quand deux ou plusieurs agents s'accordent pour partager des ressources en vue d'atteindre un but commun ;
- Supervision directe : un ou plusieurs agents ont déjà établi une relation dans laquelle un agent a un contrôle sur les autres, l'agent superviseur contrôle l'utilisation des ressources partagées (comme par exemple les ressources humaines, le temps de calcul ou l'argent) par les agents subordonnés. Il peut aussi imposer certains comportements ;
- Coordination par standardisation : le superviseur coordonne les activités en établissant des procédures que les subordonnés doivent suivre dans des situations identifiées. On trouve par exemple de telles procédures dans les entreprises, mais aussi dans les systèmes informatiques.

II.2.6. Plateforme de développement des Systèmes multi-Agents

Il existe une multitude de plates-formes multi-agents dédiées à différents modèles d'agent. Les plates-formes fournissent une couche d'abstraction permettant d'implémenter facilement les concepts des systèmes multi-agents. D'un autre côté, elle permet aussi le déploiement de ces systèmes. Ainsi, elles constituent un réceptacle au sein duquel les agents peuvent s'exécuter et évoluer. En effet, les plates-formes sont un environnement permettant de gérer le cycle de vie des agents et dans lequel ils ont accès à certains services.

Comme le choix d'une plate-forme d'agent a une grande influence sur la conception et la mise en œuvre d'un SMA, FIPA a produit les normes qui décrivent comment une plate-forme d'agent devrait être. Ces normes existent pour assurer une conception uniforme des agents indépendamment de la plate-forme

II.2.6.1. La plate-forme ZEUS

Zeus est une plate-forme dédiée pour la construction rapide d'applications à base d'agents collaboratifs. Elle se prête bien aux systèmes économiques qui utilisent des applications de planification ou d'ordonnancement. Pour implémenter les agents collaboratifs, Zeus se base principalement sur les concepts agents, buts, tâches (que les agents doivent réaliser pour atteindre leurs buts) et faits (qui représentent les croyances des agents). Un agent dans Zeus est constitué en trois couches : la couche de définition, qui contient les capacités de raisonnement et des algorithmes d'apprentissage, la couche organisationnelle, qui contient la base de connaissances et des accointances de l'agent, et la couche de coordination, qui définit les interactions avec les autres agents. Zeus propose aussi un ensemble d'agents utilitaires (serveur de nommage et facilitateur) pour faciliter la recherche d'agents. Zeus fournit aussi une méthodologie qui se base sur quatre phases pour la construction d'agents :

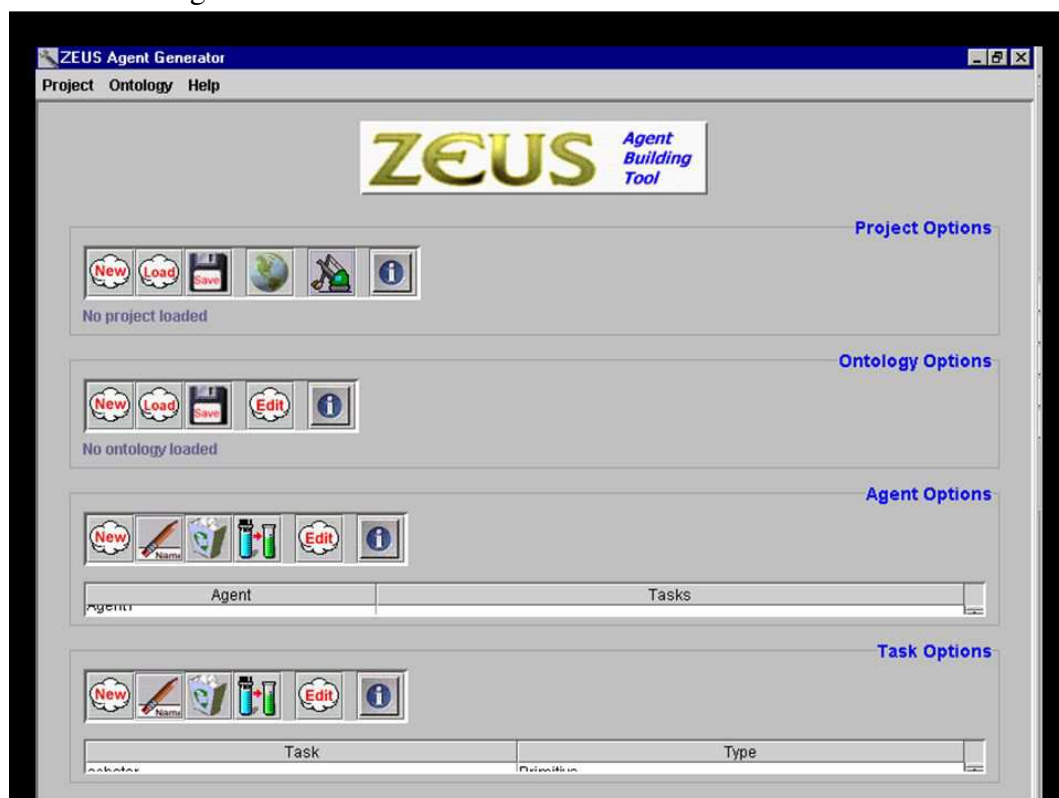


Figure 24: Plateforme Zeus

- ✓ l'analyse du domaine : consiste à modéliser des rôles. A ce stade aucun outil logiciel n'est fourni et le langage UML est utilisé.
- ✓ la conception : consiste à spécifier des buts et des tâches qui permettent de les réaliser. Cette spécification se fait par le langage naturel et n'est supportée par aucun outil.
- ✓ le développement : Cette phase est réalisée en quatre étapes : création de l'ontologie, création des agents, configuration de l'agent utilitaire, configuration de l'agent tâche et implémentation des agents. Des outils supportant des notations graphiques sont fournies afin d'aider à l'élaboration de ces étapes.
- ✓ le déploiement : dans cette phase, le système multi-agents est lancé. Un outil de visualisation permet le suivi de l'exécution. Cet outil permet de visualiser l'organisation, les interactions ayant lieu dans la société d'agents, la décomposition des tâches et l'état interne des agents.

Zeus fournit un environnement de développement d'agents grâce à un ensemble de bibliothèques Java que les développeurs peuvent réutiliser pour créer leurs agents.

II.2.6.2. La plateforme Jade

La plate-forme JADE (Java Agent Development framework) (Guizani, 2016) est celle qui est la plus utilisée par la communauté des systèmes multi-agents.

JADE permet de développer et d'exécuter des applications distribuées basées sur le concept d'agents et d'agents mobiles. Elle est compatible avec la plate-forme FIPA. Les agents dans JADE sont implémentés selon 6 propriétés :

- Autonomie : les agents ont leurs propres threads de contrôle qui leur permettent de contrôler leurs actions, de prendre leurs propres décisions afin de réaliser leurs buts mais aussi de contrôler leur cycle de vie.
- Réactivité : les agents peuvent percevoir les événements de leur environnement et réagissent en fonction de ces événements,
- Aspects sociaux : les agents exhibent des aspects sociaux qui leur permettent de communiquer et d'interagir entre eux. La communication se fait à travers le passage de messages asynchrones. La communication est considérée comme un type d'actions et peuvent de ce fait intégrer un plan d'actions. Les messages ont une sémantique et une structure définis par le standard FIPA.
- Dynamique : les agents ont la possibilité de découvrir dynamiquement d'autres agents et de communiquer avec eux.
- Offre de service : chaque agent offre un ensemble de services, il peut enregistrer ses services et les modifier. Il a aussi la possibilité de chercher des agents qui offrent les services dont il a besoin.
- Mobilité : les agents dans JADE ont la possibilité de se déplacer. Ils sont implémentés dans des conteneurs et ils peuvent se déplacer.

Un composant complémentaire est ajouté à JADE pour permettre d'intégrer l'implémentation de l'architecture interne des agents ; il s'agit de Jadex (Bordini, Dix, & Seghrouchni, 2005) qui prend en compte l'architecture des agents hybrides (c'est-à-dire des agents qui sont à la fois réactifs et proactifs). Pour les agents proactifs, JADEx se base sur le modèle BDI. JADE assure la sécurité en offrant aux applications des systèmes d'authentification qui vérifient les droits d'accès des agents.

JADE n'offre pas de méthodologie, par contre plusieurs méthodologies la prennent comme plate-forme cible lors de la génération de code tel que Gaia et PASSI.

L'implémentation de JADE est basée sur Java. La plate-forme peut être répartie sur un ensemble de machines et configurée à distance. La configuration du système peut évoluer dynamiquement puisque la plate-forme supporte la mobilité des agents.

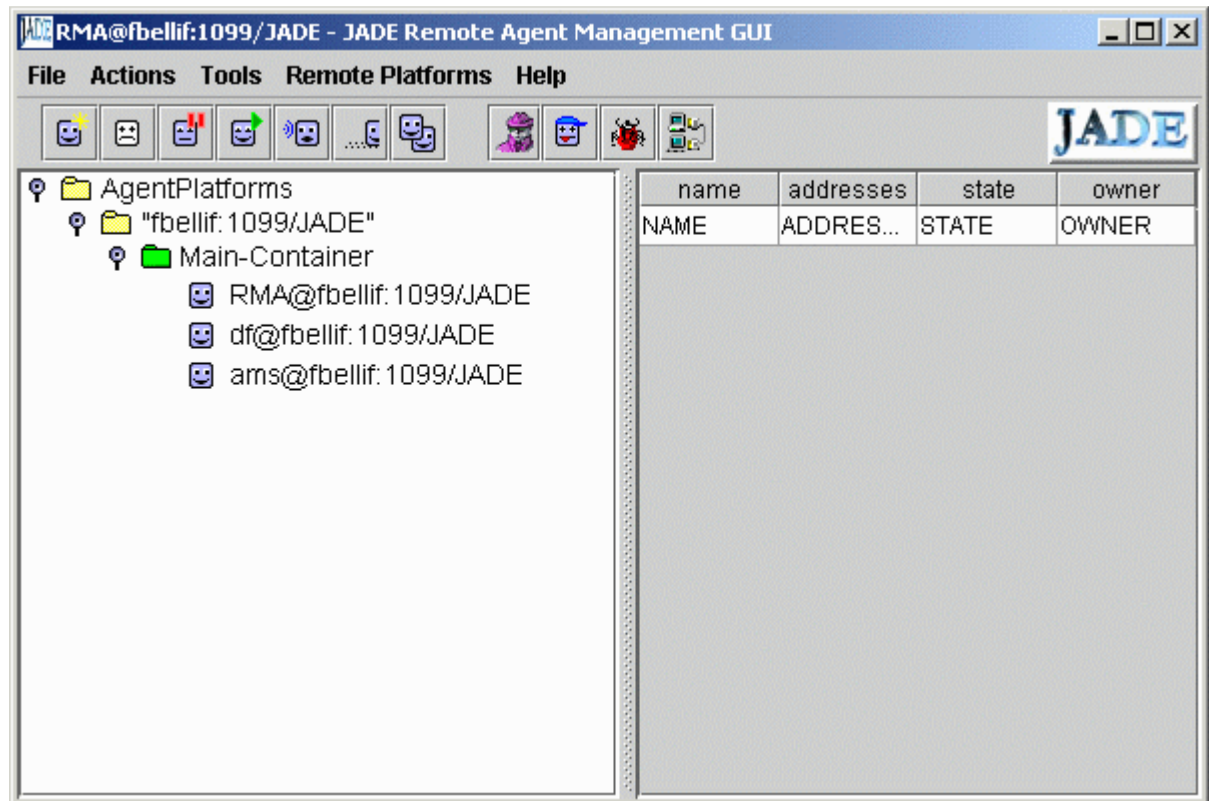


Figure 25: Plateforme Jade

II.2.6.3. La plate-forme MadKit

La plate-forme MadKit (Multi-Agents Development kit) (Rousset, Herrmann, & Lang, 2014) est développée à l'Université de Montpellier II. Bien qu'elle puisse supporter le développement de divers systèmes, elle semble bien adaptée pour les applications de simulation. La plate forme MADKIT est basée sur les concepts agent, groupe et rôle. Ces concepts sont appliqués de la manière suivante :

- ✓ L'agent : il est décrit comme une entité autonome communicante qui joue des rôles au sein de différents groupes. La faible sémantique associée à l'agent est volontaire, l'objectif étant de laisser le choix au concepteur à choisir l'architecture interne appropriée à son domaine applicatif.
- ✓ Le groupe : chaque agent peut être membre d'un ou plusieurs groupes. Il sert à identifier la structure organisationnelle d'un système multi-agent usuel.
- ✓ Le rôle : il est considéré comme une représentation abstraite d'une fonction ou d'un service. Chaque agent peut avoir plusieurs rôles et un même rôle peut être tenu par plusieurs agents. Les rôles sont locaux aux groupes.

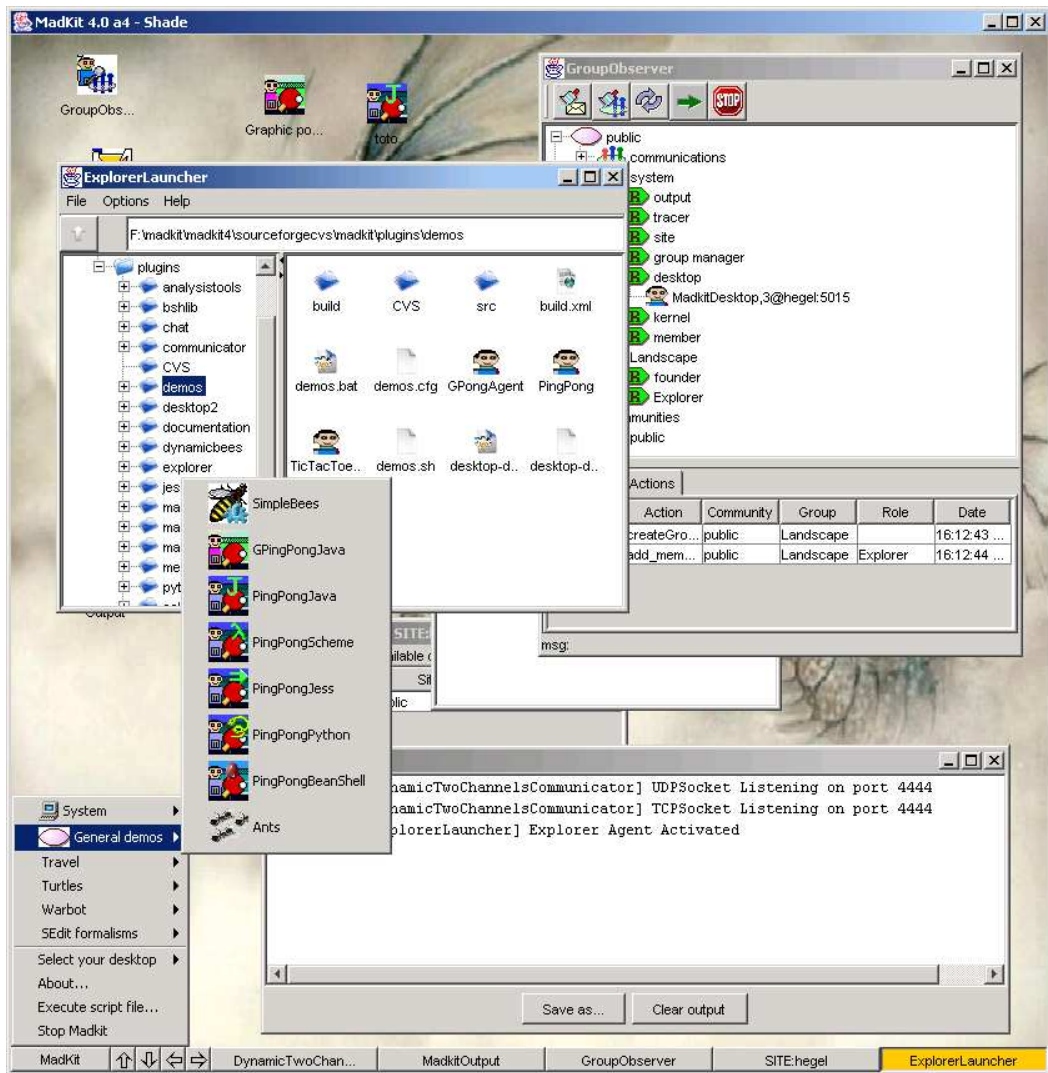


Figure 26: Plateforme Madkit

Cette implémentation correspond à la conception réalisée au niveau de la méthodologie AALAADIN (Benhadou, 2010). En effet, à partir des concepts AGR, cette méthodologie définit une démarche de développement axée sur la spécification du cadre organisationnel des applications multi-agents. Cette démarche définit l'ensemble des rôles possibles, spécifie les interactions, et décrit les structures abstraites de groupe.

MadKit fournit une API permettant la construction d'agent en spécialisant une classe d'agent abstraite. Les agents sont lancés par le noyau de MadKit, qui propose notamment les services de gestion des groupes et de communication.

L'échange des messages se fait à travers le rôle que l'agent est en train de jouer.

II.2.6.4. La plate-forme AgentBuilder

AgentBuilder est une suite intégrée d'outils permettant de construire des agents intelligents. Cette plate-forme est adaptée pour tous types de systèmes. L'élaboration du comportement des agents se fait à partir du modèle BDI et du langage AGENT-0. KQML est utilisé comme langage de communication entre les agents. AgentBuilder est composé d'une interface graphique et d'un langage orienté agent permettant de définir des croyances, des engagements et des actions. Il permet

également de définir des ontologies et des protocoles de communications inter-agents. Les agents sont décrits avec le langage Radl (Reticular Agent Definition Language), qui permet de définir les règles du comportement de l'agent. Les règles se déclenchent en fonction de certaines conditions et sont associées à des actions. Les conditions portent sur les messages reçus par l'agent tandis que les actions correspondent à l'invocation de méthodes Java. Il est aussi possible de décrire des protocoles définissant les messages acceptés et émis par l'agent.

AgentBuilder propose l'utilisation de la méthode OMT (Object Modeling Technique) durant la phase d'analyse. OMT permet de spécifier des objets du domaine et les opérations qu'ils peuvent effectuer. A partir de cette spécification une ontologie du domaine est créée. Des outils graphiques sont fournis pour effectuer ces tâches. Durant la phase de conception, les agents sont identifiés et des fonctionnalités leur sont assignées. Leurs rôles et leurs caractéristiques sont définis ainsi que les protocoles d'interaction auxquels ils participent. Durant la phase de développement, le comportement de l'agent est défini ainsi que ses croyances, intentions et capacités. Durant le déploiement, le code de l'agent est interprété par le *Run-Time Agent Engine*.

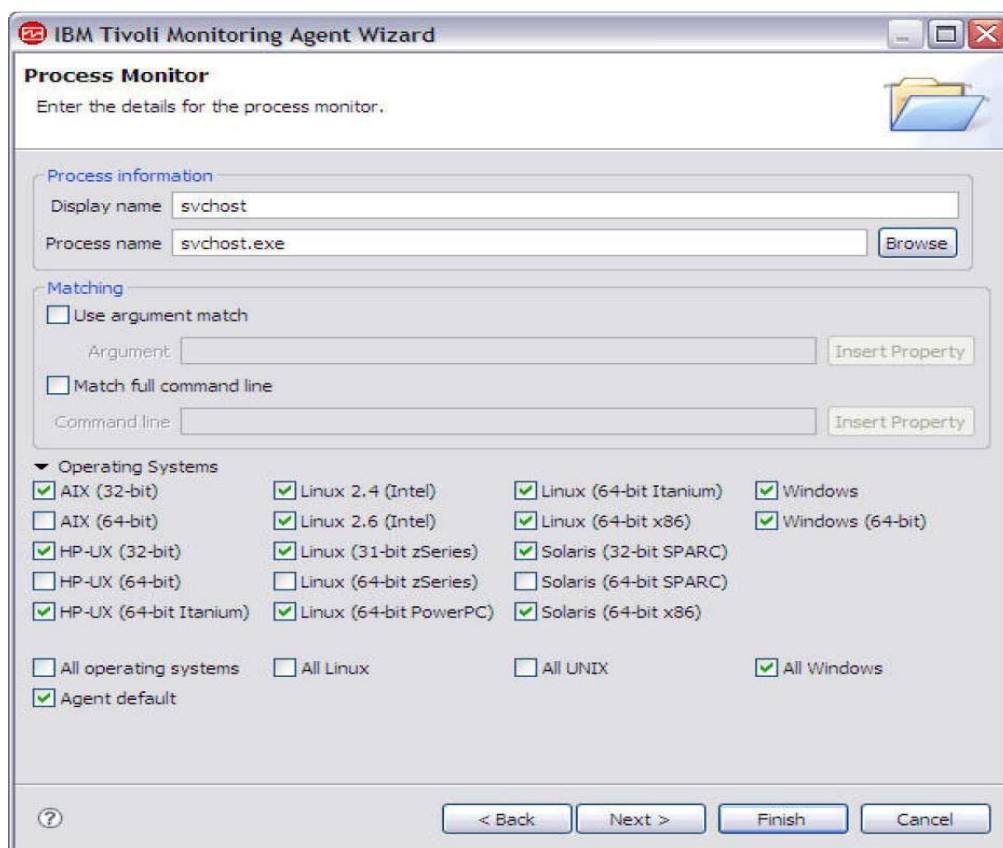


Figure 27: Plateforme Agent Builder

II.2.6.5. Choix de la plateforme multi-agents

Comme le choix de la plate-forme multi-agent a une grande influence sur la conception et la mise en œuvre du SMA, la FIPA (Foundation for Intelligent Physical Agents) a produit des normes qui décrivent comment une plate-forme de l'agent devrait être. Ces normes existent pour assurer que les agents ont une conception uniforme quelle que soit la plate-forme.

Notre choix de la plate-forme repose sur le tableau comparatif ci-dessous :

Plateformes	TYPE SMA	Modèle d'agents	Méthodologie	Langage
ZEUS	Système économique/ planification et programmation	Agent Collaboratif	Agent, objectif, Tâche : Zeus agent architecture	Java
MADKIT	Tout type	Aucune contrainte	AGR	Java/
JADE	Système simple / illustration pédagogique	Hybrid agent (JADEX)	Aucun	Java
Agent-Builder	Tout type	BDI	OMT	Java

Tableau 2.6: Comparatifs des plateforme Multi-agents

La plate-forme MadKit a été choisie car elle est générique supportant le modèle AGR sur lequel notre conception de WIO est fondée. En plus de cela, Madkit permet de construire des systèmes complexes, de contrôler le cycle de vie de l'agent et de fournir une couche complète de communication multi-agents.

II.3. Conclusion.

Dans ce chapitre, nous avons présenté les caractéristiques principales du concept de workflow Inter-organisationnel et le concept d'agent et nous avons décrit leurs atouts selon leur utilisation.

Dans la première partie de ce chapitre, nous avons défini la notion de workflow et ses différents types en l'occurrence WIO. Puis, nous avons présenté les architectures, leurs phases de conception et leurs scénarios potentiels.

Dans la deuxième partie, nous avons présenté les systèmes multi-agents, leurs apports, types, architectures et méta-modèle. Nous avons mis en évidence le rôle déterminant de l'interaction, et ses différentes formes : communication, coopération et coordination.

Ce chapitre s'est clôturé par une présentation des plateformes de développement des SMA avec une étude comparative de ces dernières qui nous permet de justifier le choix de la plateforme Madkit dans la phase réalisation.

Partie II:
SOLUTION PROPOSEE

CHAPITRE 3:
ARCHITECTURE PROPOSEE DU
WORKFLOW DE GOUVERNANCE
SI

III. Architecture du Workflow de Gouvernance SI proposée.

III.1. Introduction

Après avoir donné un aperçu sur les WF, nous proposons dans ce chapitre, un WIO de gouvernance SI à base des Systèmes Multi-Agents.

Dans la première partie de ce chapitre, nous exposons les fonctionnalités et les avantages de la solution proposée.

Dans la deuxième partie, nous détaillons les phases de modélisation du workflow. Ensuite, nous présentons les différentes versions de l'architecture du WIO en mettant en relief les avantages et les perspectives de chacune d'elles (Chergui, Sayouti, & Medromi, 2013) (Chergui, Sayouti, & Medromi, 2014) (Chergui, Chakir, Sayouti, & Medromi, 2016).

III.2. Plateforme EAS IT- GRC

III.2.1. Présentation

EAS IT GRC est un projet de l'équipe EAS depuis Septembre 2011. Il s'agit d'une plateforme d'aide à la bonne gouvernance, la gestion des risques et la conformité des technologies de l'information et de la communication au sein d'une entreprise, comprenant un ensemble de systèmes distribués. La plateforme permet de/d' :

- Assurer et évaluer l'alignement des objectifs métiers de l'entreprise avec la stratégie SI ;
- Gérer les processus IT ;
- Prioriser les investissements informatiques en adéquation avec l'apport de valeur métier ;
- Gérer les risques IT et les évalue ;
- Assurer la conformité aux cadres légaux;
- Choisir pour un processus donner le meilleur référentiel de la gouvernance, risque et conformité des Systèmes d'information.
- Mettre à jour les référentiels selon les dernières versions existantes sur le marché.

La plateforme se base sur les référentiels et les méthodes de la gouvernance, risque et conformité des Systèmes d'information (IT GRC), les plus répandues, à savoir, COBIT, ITIL, PMBOK, ISO27001, ISO27002, ISO27005, MEHARI, EBIOS. Elle aide une entreprise de taille, d'activité et de Système d'information (SI) quelconque, à gérer ses processus IT en adéquation avec sa stratégie métier en impliquant les responsables métier et la direction des Système d'information (DSI). Elle permet

d'optimiser les investissements informatiques en contrôlant les stratégies métiers afin d'y créer plus de valeur, rendre les processus informatiques plus performants et contrôler les risques et la conformité liés au SI. Elle déploie une variété de référentiels IT GRC et fait un choix intelligent, selon les contraintes et paramètres de l'entreprise, du meilleur référentiel afin d'évaluer les objectifs et les processus en question.

La plateforme communique avec les parties prenantes du SI : le DSI et les Responsables métiers de chaque département. Elle possède un moteur sémantique intelligent qui permet de traduire les objectifs exprimés par ses utilisateurs en langage compréhensible par les référentiels à fin de mettre en œuvre le traitement IT GRC adéquat. Un système de décision intelligent, multicritère permet de choisir le meilleur référentiel pour une demande donnée (Chakir, Chergui, Elhasnaoui, Medromi, & Sayouti, 2016). La plateforme encapsule chaque référentiel dans un système expert, permettant de faire une évaluation complète de la demande, d'une façon interactive avec l'utilisateur concerné, journalisée et documentée. Ces référentiels sont actualisés à l'aide d'un système intelligent et incrémental de mise à jour. Les différents composants de la plateforme communiquent via une couche de communication qui fonctionne en deux modes synchrone et asynchrone (Elhasnaoui, Chakir, Chergui, Iguer, Faris, & Medromi, 2015).

III.2.2. Description

Il s'agit d'une plateforme de gouvernance, risque et conformité des systèmes d'information comportant :

- Une couche stratégique (Chergui, Medromi, & Sayouti, Intelligent audit of information systems by inter-organizational workflow, 2014), **sujet de cette thèse**, permettant de : persister les configurations dynamiques et statiques de l'entreprise, encapsuler les objectifs métiers liés aux technologies de l'information des divers départements de l'entreprise et les faire correspondre avec les objectifs informatiques et les processus informatiques adéquats. Elle permet également d'éditer la matrice des responsabilités, le modèle de maturité et les objectifs de contrôles de la stratégie en question. A la fin de son traitement, cette couche envoie la synthèse des résultats à la couche communication pour une éventuelle redirection vers les composants de traitement dans un but de spécialisation.
- Une couche de décision (Chakir, Chergui, Medromi, & Sayouti, 2016) : capable de choisir pour une demande provenant de la couche stratégique, le meilleur référentiel de gouvernance IT. Ceci pour détailler les activités et les mesure à exécuter pour un processus IT selon sa catégorie (Gouvernance, Risque, Conformité). Pour se faire, elle se base sur la configuration

de l'entreprise et des critères d'évaluation de processus IT par référentiel. Elle offre deux modes de décision : un mode orienté processus et un mode orienté activité. Chaque mode est pris en charge par des agents intelligents exécutants deux algorithmes de choix le premier par critères et le deuxième par référentiel. Une notification de meilleurs référentiels est envoyée immédiatement à la couche communication.

- Une couche de traitement (Iguer, Medromi, Sayouti, & Tallal, 2016) : représente chaque référentiel IT GRC par un système intelligent et autonome, qui déploie les actions et met en place l'ensemble des recommandations du référentiel d'une manière interactive. En effet, l'interaction se fait par un envoi d'une requête à la couche stratégique, pour demander une information statique préalablement configurée, ou ouvrir un formulaire d'échange avec un utilisateur potentiel, dont les réponses sont redirigées vers la base de connaissance du système en question. Du point de vue technique, elle dispose d'un ensemble de sous systèmes experts liés à des bases de connaissances propres au référentiel implémenté. Ses sous-systèmes disposent d'un ensemble d'agents intelligents, communicants, permettant de décortiquer la demande, d'interroger l'utilisateur concerné et d'analyser l'existant pour donner une réponse efficace et documentée. Chaque sous système de la couche traitement est amené à envoyer une demande de spécification en message à la couche stratégique, passant par la couche de communication.
- Une couche de communication: assurant la communication de bout en bout entre les différentes couches de la solution en deux modes différents : synchrone par envoi de message et asynchrone par partage d'information. Chaque mode est déclenché selon les spécificités de l'organisation et de la stratégie en question. Elle comporte un bloc de communication par couche, et ce pour la particularité des flux de chaque couche et la spécificité des traitements à lancer en amont avant de rediriger le flux d'information à la couche suivante.
- Une couche de mise à jour : prenant en charge l'actualisation des versions des référentiels utilisés pour une mise à niveau périodique de la plateforme. Cette mise à niveau est assurée à partir d'une correspondance entre les processus de l'ancienne et la nouvelle version, injectant l'information nécessaire aux bases de connaissance des différents blocs de la plateforme. Du point de vue technique, cette couche actualise les versions des référentiels et enrichie les bases de connaissances à l'aide des nouvelles version, à l'aide d'un script de chargement de fichiers plats de correspondances, entre l'ancien et la nouvelle version de chaque référentiel, dans la base de connaissance appropriée. Nous précisons que chaque sous système contient son propre mécanisme de mise à jour.

En bref, la plateforme IT-GRC est une solution WEB, sécurisée ne nécessitant aucune installation locale, elle est basée sur les systèmes multi-agents (SMA) dans ses différentes parties à savoir l'interface utilisateur, la configuration statique et dynamique de l'organisation, la gestion des profils, le choix du meilleur référentiel et le traitement des processus. Elle tire profit de l'aspect autonomie et apprentissage des SMA ainsi de leur communication et coordination de haut niveau.

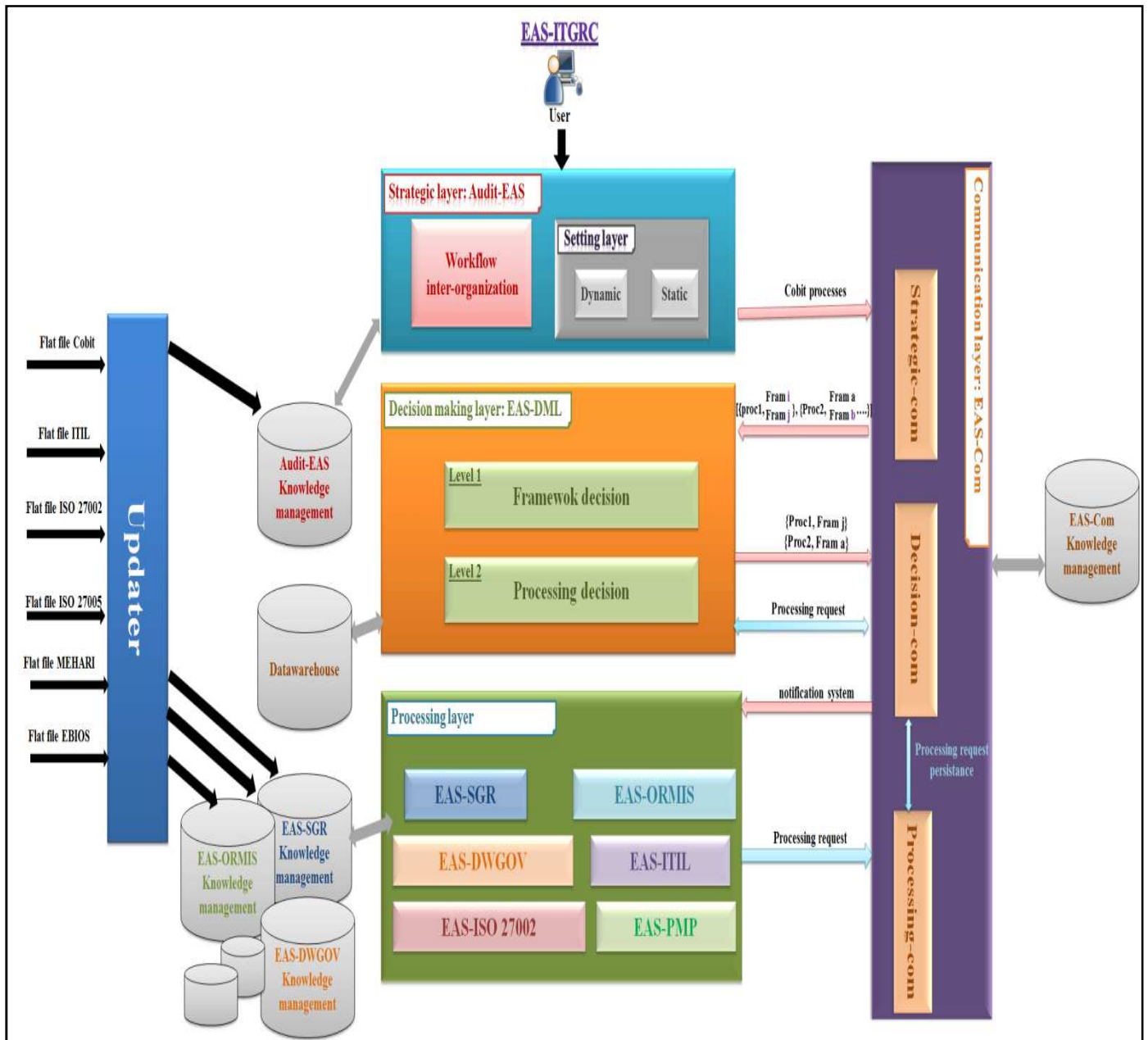


Figure 28: Architecture Globale EAS IT-GRC

III.2.3. Contribution de la solution proposée dans la plateforme EAS IT-GRC

Le workflow de gouvernance IT nommé « EAS IT Stratégie », sujet de cette thèse, est la couche stratégique de la plateforme EAS IT-GRC qui se charge de la configuration statique de l'environnement. Il représente l'interface avec les parties prenantes pour exprimer leurs demandes à propos des applicatifs du système d'information qu'ils utilisent. L'interprétation de ces demandes en processus IT permet l'évaluation des:

- Demandes métiers via un système expert de médiation (Chergui, Chakir, Sayouti, & Medromi, 2015) ;
- Processus IT via les systèmes de traitement de la plateforme.

Autrement dit, cette solution permet de traduire les demandes des utilisateurs en requêtes compréhensibles par l'ensemble des référentiels IT GRC. La demande traitée par le workflow de gouvernance SI est encapsulée en service, est envoyé via sa couche de Médiation au système Strategic-COM de la couche communication. La demande est ensuite transmise à la couche décision qui choisira le meilleur référentiel pour détailler le traitement des processus IT. Ce service peut être envoyé en deux modes : synchrone et asynchrone.

Pour synthétiser le fonctionnement de la couche stratégique, ainsi que ses échanges avec la plateforme globale, nous présentons ci-après le modèle de communication.

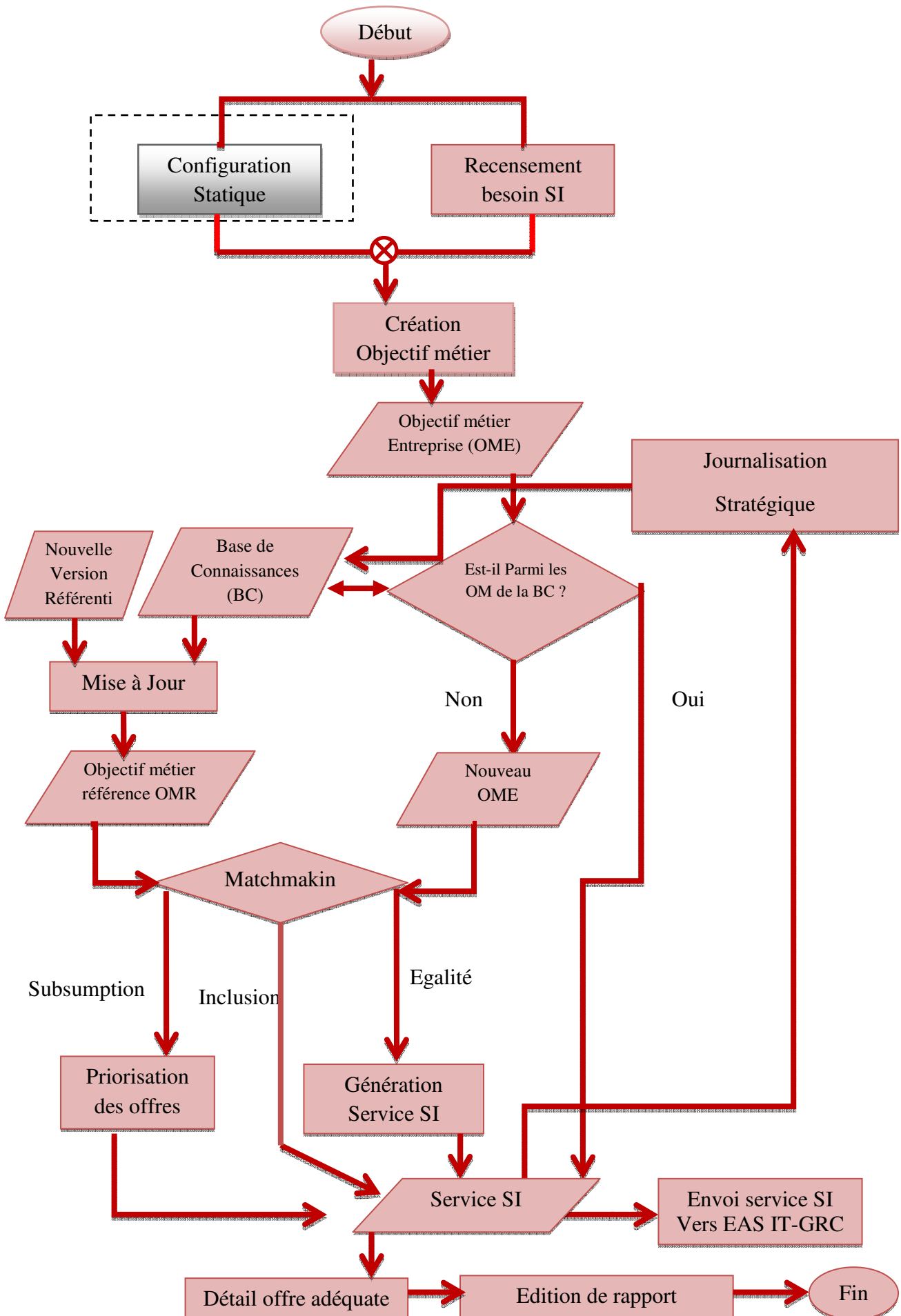


Figure 29 : Modèle de Communication du Workflow de Gouvernance IT

III.3. Fonctionnalités et Avantages

L'objectif de ce travail est de proposer un workflow de gouvernance SI, qui couple l'appui sur les bonnes pratiques GSI et l'adaptabilité à la complexité et aux changements des SI.

L'architecture proposée consiste à concevoir une solution orientée processus qui permet :

- L'analyse stratégique d'un système d'information de taille et de type quelconque grâce au WIO ;
- L'énumération des activités informatiques à mettre en place et leur optimisation ;
- La présentation des niveaux de maturité, des mesures et des indicateurs de performance;
- La définition de la matrice des responsabilités et des tests de contrôle adéquats ;

Parmi les avantages de cette architecture, nous citons la distribution, l'autonomie l'apprentissage grâce aux SMA, l'efficacité sémantique et la portabilité sur le web grâce à l'ontologie ITGovernanceOntology (Chergui, Sayouti, & Medromi, 2014).

En plus, cette solution est destinée à tous les utilisateurs du SI, pour s'auto-auditer en temps réel en couplant les directives du référentiel COBIT et le savoir faire de l'entreprise.

III.4. Modélisation du WIO

Comme nous avons vu dans la partie état de l'art, la modélisation d'un workflow en général, un WIO en particulier, passe par la définition de trois niveaux à savoir : le niveau organisationnel, le niveau fonctionnel et le niveau comportemental. Dans ce qui suit, nous définissons ces trois niveaux pour le WIO de Gouvernance SI proposé :

III.4.1. Niveau Organisationnel

Nous considérons le WIO de gouvernance IT comme une organisation virtuelle ayant comme acteurs les agents suivant :

Acteurs	Rôle
Agent Workflow SI	-Représenter un objectif métier réel. -Recueillir les réponses et notification utilisateurs. -Envoyer une demande au serveur de connexion SI. -Déclencher l'audit autour d'un objectif métier.
Agent Gestionnaire SI	-Ajouter, Modifier, supprimer un agent Workflow SI. - Paramétrer l'interface client relié à l'agent selon le niveau de maturité. -Suspendre des agents Workflow SI.
Serveur de connexion SI	-Gérer les demandes d'audit et les demandeurs. -Envoyer les demandes d'audit à l'entité de médiation. -Recevoir les offres d'audit de l'entité de médiation. -Envoyer les coordonnées de l'agent COBIT auditeur aux Agent Wf SI.
Agent Médiateur	-Recevoir les demandes d'audit. -Recevoir les offres d'audit. -Etablir la correspondance entre l'offre et la demande. -Enregistrer les nouvelles correspondances.
Serveur de connexion COBIT	-Gérer les offres d'audit et les fournisseurs. -Envoyer les offres d'audit à l'entité de médiation. -Recevoir les demandes d'audit de l'entité de médiation.

		-Envoyer les coordonnées de l'agent Wf SI aux Agents COBIT.
Agent COBIT	Gestionnaire	-Ajouter, Modifier, supprimer un agent COBIT. -Suspendre des agents COBIT.
COBIT		-Représenter un objectif métier COBIT. -Envoyer une offre au serveur de connexion COBIT. -Auditer un Objectif métier selon le référentiel COBIT.

Tableau 3.1: Agents et rôle du WIO de GTI

Nous avons proposé un WIO basé sur le méta-modèle AGR des SMA et nous avons défini cinq Groupes d'agent :

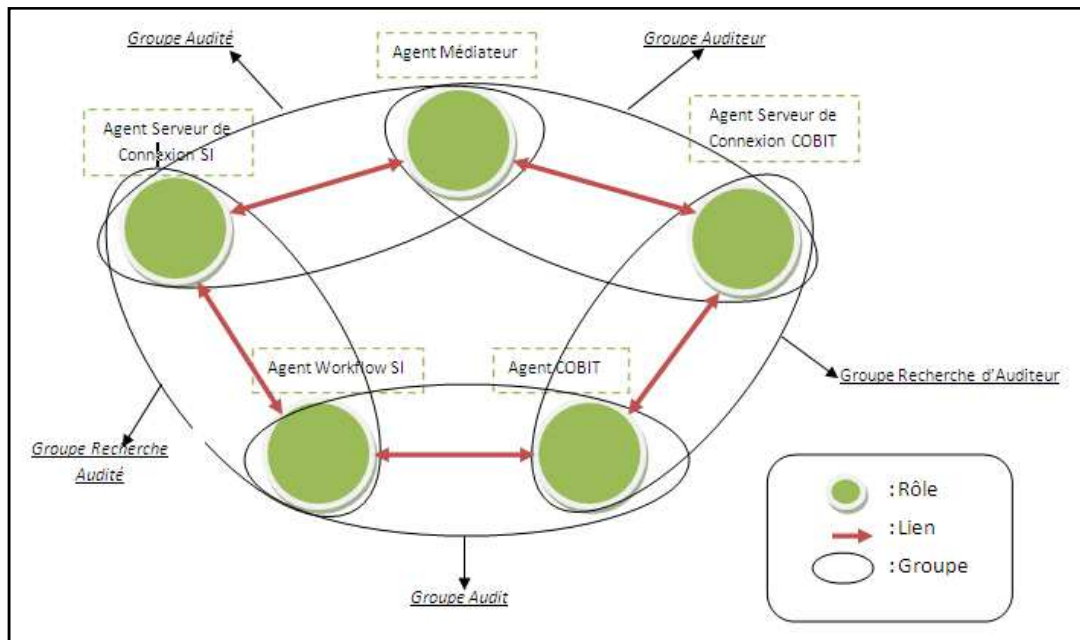


Figure 30: Modèle Organisationnel du WIO

- **Groupe Auditeur** : Contient l'agent médiateur et le serveur de connexion COBIT.
- **Groupe Audité** : Contient l'agent médiateur et le serveur de connexion SI.
- **Groupe Recherche Auditeur** : les agents COBIT, le serveur de connexion COBIT et l'Agent Gestionnaire COBIT.
- **Groupe Recherche Audité** : les Agents workflow SI, le serveur de connexion SI et l'agent gestionnaire SI.
- **Groupe Audit** : l'agent COBIT et l'agent Workflow SI.

III.4.2. Niveau fonctionnel

L'aspect fonctionnel décrit les offres et les demandes d'audit indépendamment des règles, des événements et des acteurs. Ce modèle est inspiré de la relation entre les composants du référentiel COBIT.

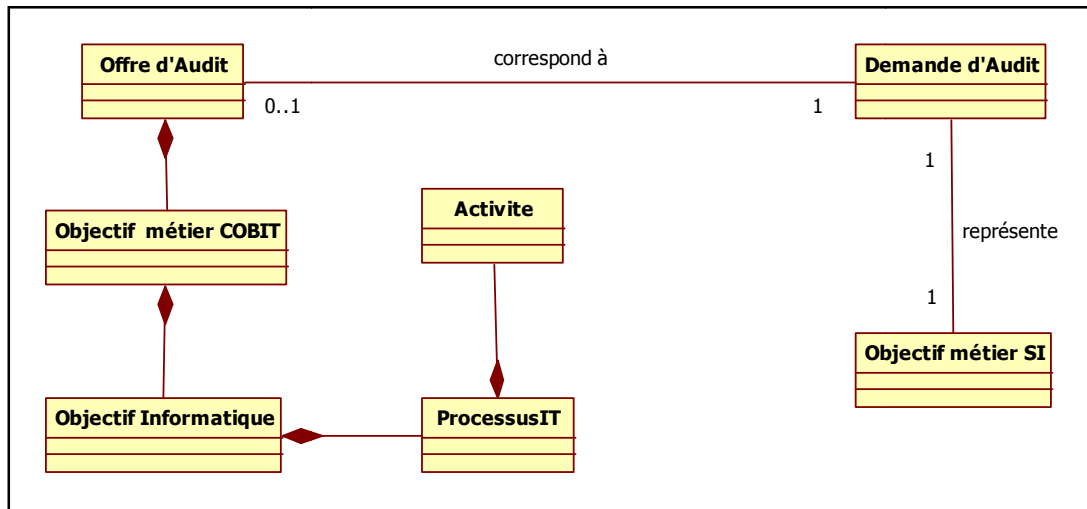


Figure 31: Liaisons fonctionnelles des composants COBIT au sein du WIO

III.4.3. Niveau Comportemental

A ce niveau, nous décrivons le comportement des agents au sein du WIO dépendamment des entrées et sorties avec un diagramme de séquence. Il est à noter qu'un éventuel diagramme d'activité avec les acteurs est aussi possible pour représenter ce niveau (Gruninger, Schlenoff, Knutilla, & Ray, 1997).

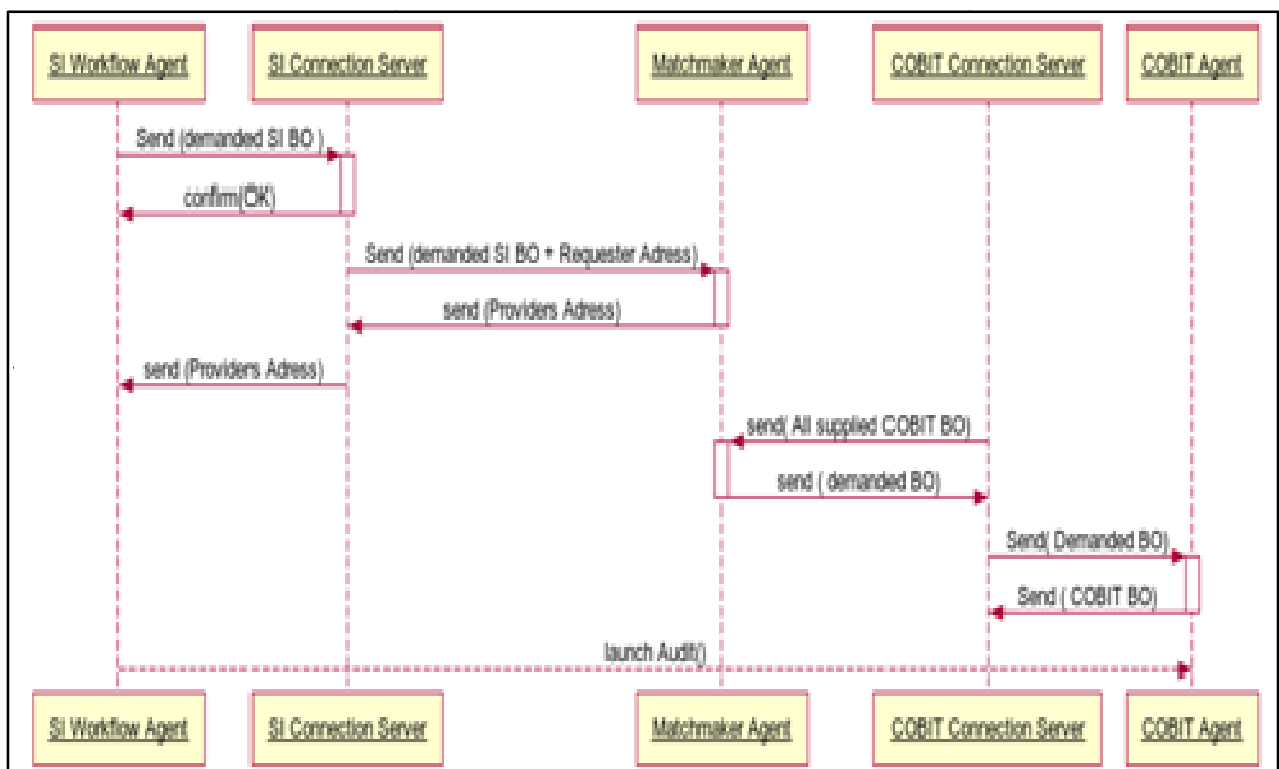


Figure 32: Diagramme de séquence WIO

III.5. Architecture proposée

III.5.1. EAS IT Stratégie –Version 1

L'architecture du workflow Inter-organisation de type tâche de gouvernance IT, dans sa première version, est une solution qui permet de gouverner en temps réel chaque composant du SI sans prendre en considération ses spécificités techniques et son interconnexion avec le reste des composants. Elle est basée sur les systèmes multi-agents et le référentiel COBIT dans sa version 4.1.

L'architecture proposée implémente l'architecture de référence de WIO, cité précédemment, et la liaison entre les composants du référentiel COBIT 4.1, elle contient :

- **Agents Workflow SI** qui représente une application métier du SI non obligatoirement en communication avec les autres applications du SI.
- **Agents COBIT (Objectif Métier/Objectif Informatique/Processus IT)** qui existent en trois catégories, un agent qui représente un objectif métier COBIT appelé agent objectif métier. Ce dernier gère un ensemble d'agents objectifs informatiques, qui représentent des objectifs informatiques et qui font appel à des agents processus IT.
- **Agent Manager SI** qui gère les agents workflow SI (création/suspension/partage de ressources).

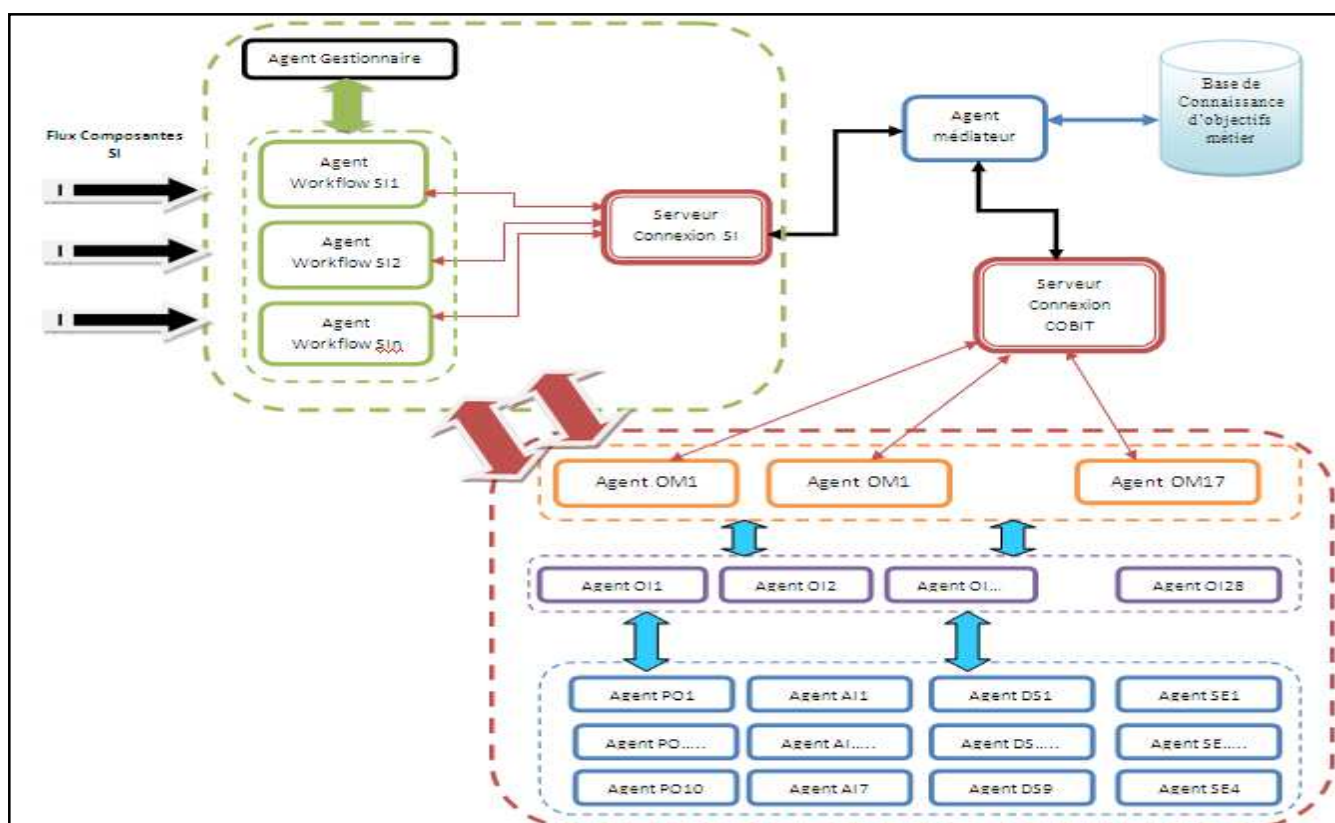


Figure 33: Architecture proposée version 1

- **Agent Manager COBIT** : qui gère les agents COBIT (création/suspension/partage de ressources).
- **Agents Serveurs de Connexion** qui offre le service de pages jaunes pour la publication des offres et des demandes respectivement des agents COBIT et des agents Workflow SI.
- **Agent Médiateur** qui établit la correspondance entre la demande des Agents Workflow et l'offre des Agents COBIT.

Cependant, cette architecture nécessite des améliorations par rapport aux points suivant :

- Très grand nombre d'agent COBIT (problème de temps d'exécution) et rigidité par rapport au modèle d'audit (difficulté de migration vers de future version ou modèle).
- Communication entre agent restreinte à l'envoi de message.
- Nécessité de détail au niveau médiation qui utilise uniquement des tests logiques (si/sinon)

III.5.2. EAS IT Stratégie –Version 2

L'évaluation de la première version, nous a poussé à opter pour cette deuxième version d'architecture qui améliore la partie COBIT, détaille la partie médiation et optimise les types des agents utilisés.

En effet, la correction de la partie COBIT consiste à encapsuler les objectifs métiers COBIT, dans des agents réactifs, au lieu de représenter chaque composant COBIT, par un agent. En plus, les objectifs informatiques, les processus IT sont des ressources lancées au déclenchement de ces agents.

Pour la partie médiation, après un état de l'art des agents médiateurs, nous avons opté pour un matchmaker car l'opération d'audit en question, doit se faire directement entre le fournisseur « Agent COBIT » et le demandeur « Agent Workflow SI ». Le médiateur sert uniquement à établir la liaison entre ces deux parties.

La deuxième version de l'architecture proposée, présente les améliorations suivantes:

- **Composant COBIT** : dans la première version, nous avons « agentifié » les composants du référentiel COBIT, nous avons donc eu dans cette couche 34 agents processus IT, 17 agents objectif métier et 24 agents objectif informatique. A l'exécution nous avons un nombre important d'agent qui se lance par requête. D'où une lourdeur d'exécution du système. Nous avons donc exploité la liaison fonctionnelle entre ces composants pour définir ce qu nous avons appelé **Agent COBIT**, qui est l'agent objectif métier de la première version, avec un appel systématique des objectifs informatiques et des processus IT correspondants. Ces derniers sont considérés comme des objets ressources de l'agent COBIT. L'agent objectif métier est un agent réactif ayant comme tâche de se déclencher à l'appel et se présenter l'objectif métier en question.

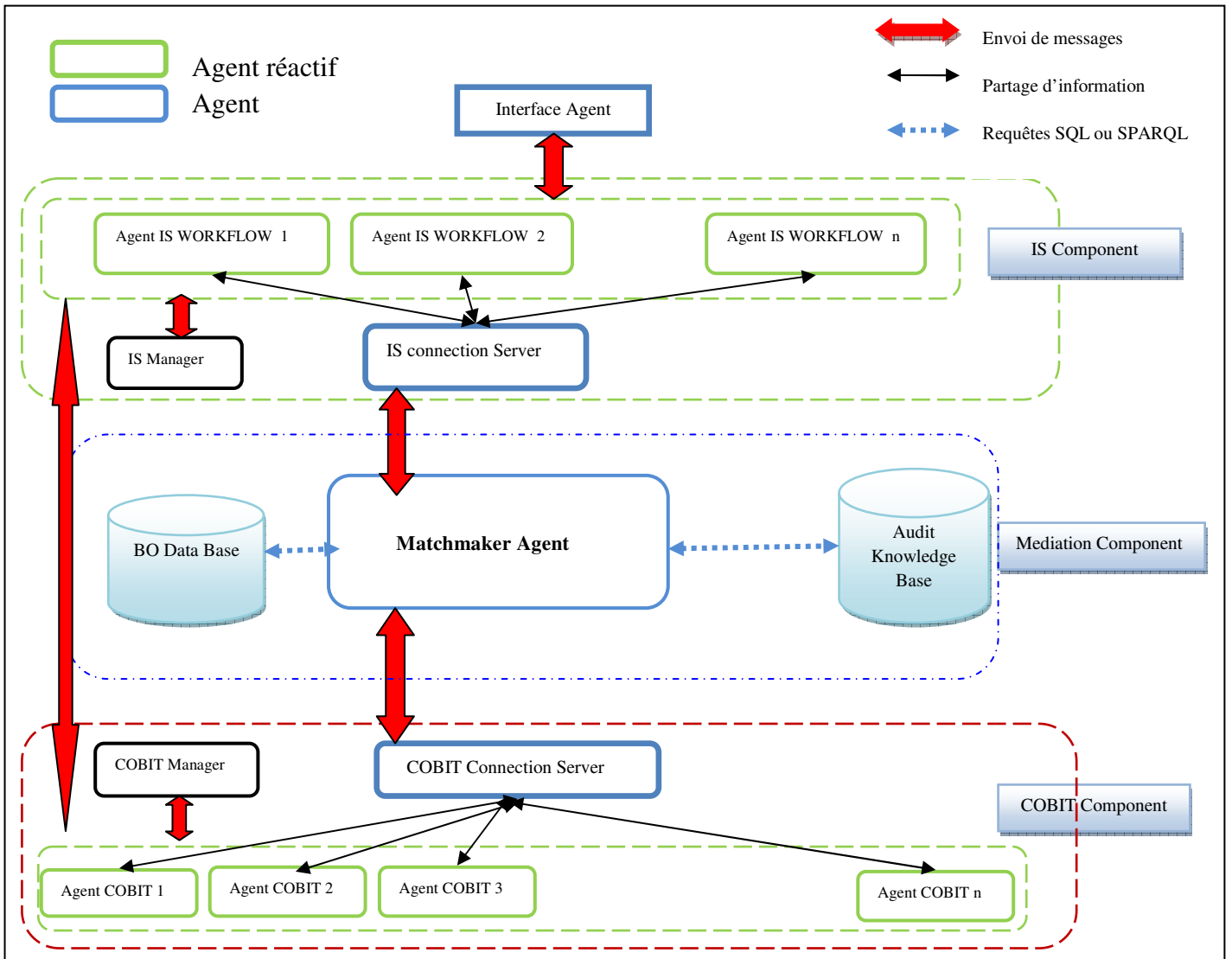


Figure 34: Architecture proposée version 2

Agent médiateur : Il existe principalement trois types d'agents de médiation, « Matchmaker », « Facilitator » et « Broker ». La différence entre un « Matchmaker » et un « Facilitator » est que le premier permet d'échanger les identités du demandeur et du fournisseur, puis les deux parties communiquent directement, le second est un intermédiaire de transaction. Quant à un « Broker », il présente des services délégués aux préférences du demandeur, puis il demande au fournisseur des résultats et envoie directement ce résultat au demandeur. Dans notre cas, un agent Matchmaker est le meilleur choix afin de relier l'agent IS workflow et l'Agent COBIT. Les informations autour de la requête s'échangent directement entre les deux agents sans qu'une partie tierce n'interfère, ceci permet de simuler une opération réelle basée sur les entrevues de l'utilisateur et de proposer des recommandations pratiques.

Le rôle de l'agent médiateur est donc de trouver le ou les meilleurs agents COBIT pour l'agent Workflow SI.

- **Types d'agent** : Dans cette deuxième version, nous avons défini les types des agents selon les rôles et les fonctionnalités attribués à chacun, le tableau ci-dessous présente les agents, leurs types et la justification du choix.

Nom de l'Agent	Type de l'Agent	Justification du choix
Agent Workflow SI	Réactif	Comportement stimulus réponse face à une requête de l'utilisateur, ne mémorise pas le résultat ni la requête, n'a pas une représentation de l'environnement.
Agent Cobit	Réactif	Comportement stimulus réponse pour chaque offre, présents en nombre important (autant d'agent COBIT que d'objectif métier), aucune mémorisation ni connaissance de l'environnement n'est nécessaire.
Serveur de Connexion	Cognitif	Deux serveurs dans la plateforme un pour les Workflow SI et un pour les agents COBIT, ayant comme fonction la collecte des adresses des agents (mémorisation et connaissance de l'environnement) et la redirection des identités et des requêtes vers la couche médiation (interaction pour résolution du problème).
Agent Manager	Cognitif	Deux agents Manager sont présents, Manager SI et Manager COBIT, ayant comme fonction, la gestion des agents Workflow SI et COBIT (connaissance de l'environnement et mémorisation des états).
Agent médiateur	Cognitif	Une seule entité est présente ayant comme tâche la persistance des demandes et des offres, leur traitement et la correspondance ou le matchmaking (voir architecture Matchmaker).

Tableau 3.2: Types des agents et justification du choix

Pour l'agent de Médiation, nous avons proposé l'architecture suivante :

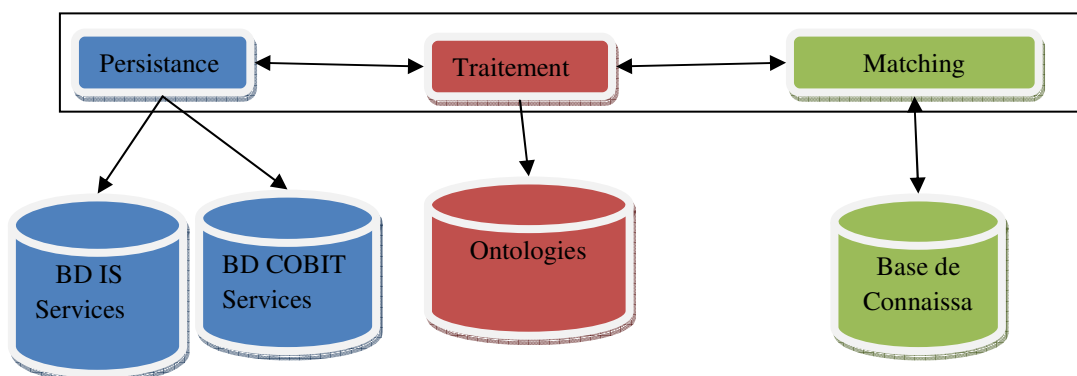


Figure 35 : Architecture agent médiateur

Couche Persistance se charge du stockage des processus métier SI (services demandés) et des processus métiers COBIT (services offerts) avec identités des agents créateurs, la date de création et la priorité de la demande.

Couche Traitement : permet la décomposition hiérarchique des services demandés en utilisant les concepts de bases du référentiel COBIT, afin de les comprendre et à les classer.

Couche Correspondance : permet d'effectuer le choix des offres correspondantes à la demande initiale, à travers un algorithme de comparaison

Chaque couche est reliée à une source de données, une base de données des services pour la couche persistance, une base de connaissance avec des règles et des faits pour la médiation et un fichier de terminologie pour le traitement.

En ce qui concerne le mode de communication entre les agents, le mode d'envoi de messages n'est pas optimal en temps d'exécution. Le problème se pose surtout pour les serveurs de connexion ayant à gérer des messages de déclencheurs simultanés, pour les agents workflow SI et les agents COBIT. Nous avons donc varié les modes de communication en partage d'information et envoi de message selon les situations. Le tableau ci-dessous présente les modes de communication des agents.

<i>Agents</i>	<i>Mode de communication</i>	<i>échange</i>
Agent Manager -Agent Workflow SI	Envoi de message	Instruction de gestion.
Agent Workflow SI- Serveur de Connexion SI	Partage d'information	Requête « demande »
Serveur de Connexion SI- Matchmaker	Envoi de message	Requête « demande »+ adresse demandeur.
Serveur de Connexion COBIT- Matchmaker	Envoi de message	Réponse « offre »+ adresses fournisseurs.
Agent Manager- Agent COBIT	Envoi de message	Instruction de gestion.
Agent COBIT – Serveur de Connexion COBIT	Partage d'information	Réponse« offre ».

Tableau 3.3: Mode de communication des agents

Malgré les avantages que présente la deuxième version de l'architecture, il reste des points à améliorer à savoir :

- Un enrichissement de l'aspect sémantique de la couche médiation est indispensable. En effet, nous utilisons dans la deuxième version sur un fichier XML contenant les concepts de bases de la gouvernance SI pour comparer les demandes réels avec les objectifs métiers du noyau COBIT : ceci présente une limitation dans le sens où la couverture des domaines de la gouvernance SI n'est pas totale.
- Une nécessité d'apprentissage des requêtes précédentes pour construire un référentiel propriétaire de l'entreprise basé sur le noyau COBIT.
- La mise à jour officielle du référentiel COBIT doit être prise en considération.

III.5.3. EAS IT Stratégie –Version 3

III.5.3.1. Présentation

Dans cette version et pour améliorer la deuxième version:

1. Nous avons détaillé l'architecture de l'agent Workflow SI.
2. Nous avons remplacé l'agent médiateur par un système expert de médiation qui assure :
 - l'édition des services demandés et offerts,
 - l'explication de ces services via l'ontologie « IT Governance Ontology »
 - le matchmaking sémantique.
3. Nous avons remplacé l'agent Framework Manager par un Système multi-agent contenant en plus des fonctionnalités de cet agent, l'apprentissage et la mise à jour.
4. Nous avons remplacé les agents COBIT par des agents Business Goal qui peuvent être créés, en plus du noyau COBIT, à partir d'une requête traitée propre à l'entreprise.

III.5.3.2. Description

Nous avons décrit dans ce qui précède, l'agent Workflow SI comme étant un agent réactif qui communique avec l'utilisateur d'une composante du SI pour recevoir sa demande et la transmettre au serveur de Connexion SI. Techniquement nous avons implémenté cette fonction par un affichage directe de la requête utilisateur sur l'interface d'un agent qui se déclenche et envoie le message au serveur.

La structure de référence des agents interface (Lieberman, 1997) s'impose, pour une optimisation de la structure interne de l'agent et une éventuelle évolutivité. Elle consiste en ce que l'agent interface ait une entrée, un traitement et une sortie interconnectés les uns aux autres avec possibilité de revenir de la sortie à l'entrée.

Nous avons donc proposé une architecture de l'agent workflow SI, conforme à la structure de référence des agents interface et adaptée à sa fonctionnalité dans l'architecture globale. (Voir figure ci-dessous)

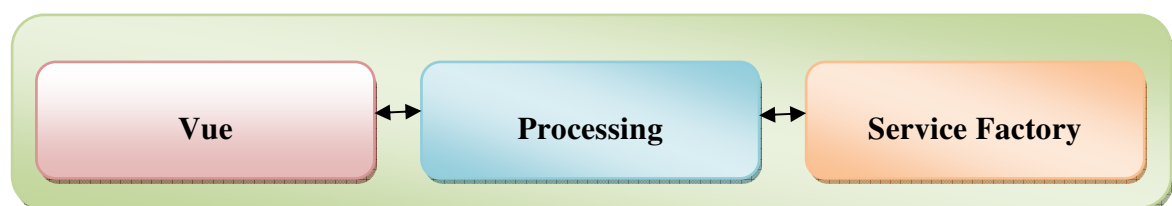


Figure 7: Architecture Agent workflow SI

L'agent Workflow SI est constitué de trois parties :

- **Vue** est l'interface d'expression de la demande du client concernant un composant du SI, qui se lance à partir de l'interface globale une fois l'utilisateur évoque la composante en question.
- **Processing** est la partie responsable de la reformulation de la demande client selon le modèle des demandes traitées par la plateforme en captant les différentes

variantes du système (date, identité du demandeur, état de la demande, priorité de la demande, etc.).

- **Service Factory** est l'entité responsable de la création des services de la plateforme.

En ce qui concerne la médiation de l'offre et de la demande, un seul agent médiateur dans la plateforme n'est pas suffisant, dans la mesure où plusieurs fonctionnalités s'imposent en plus du parallélisme des tâches à exécuter. En effet, l'architecture proposée, soulève trois fonctionnalités à savoir : la persistance des demandes et des offres, l'interprétation sémantique et la médiation. Ces trois tâches dépassent sur le plan pratique les capacités d'un agent quoique cognitif. D'autant plus que dans la partie médiation, une consultation de base de connaissance est nécessaire pour faire correspondre la meilleure offre à la demande. D'où l'idée de remplacer l'agent médiateur par un système expert de médiation dont le moteur d'inférence est un moteur sémantique et l'éditeur est l'entité de persistance.

Nous consacrons le chapitre 5 à la modélisation de ce système avec le détail des règles et des faits de la base de connaissance, l'algorithme de comparaison ainsi que l'implémentation et l'évaluation de ce système expert.

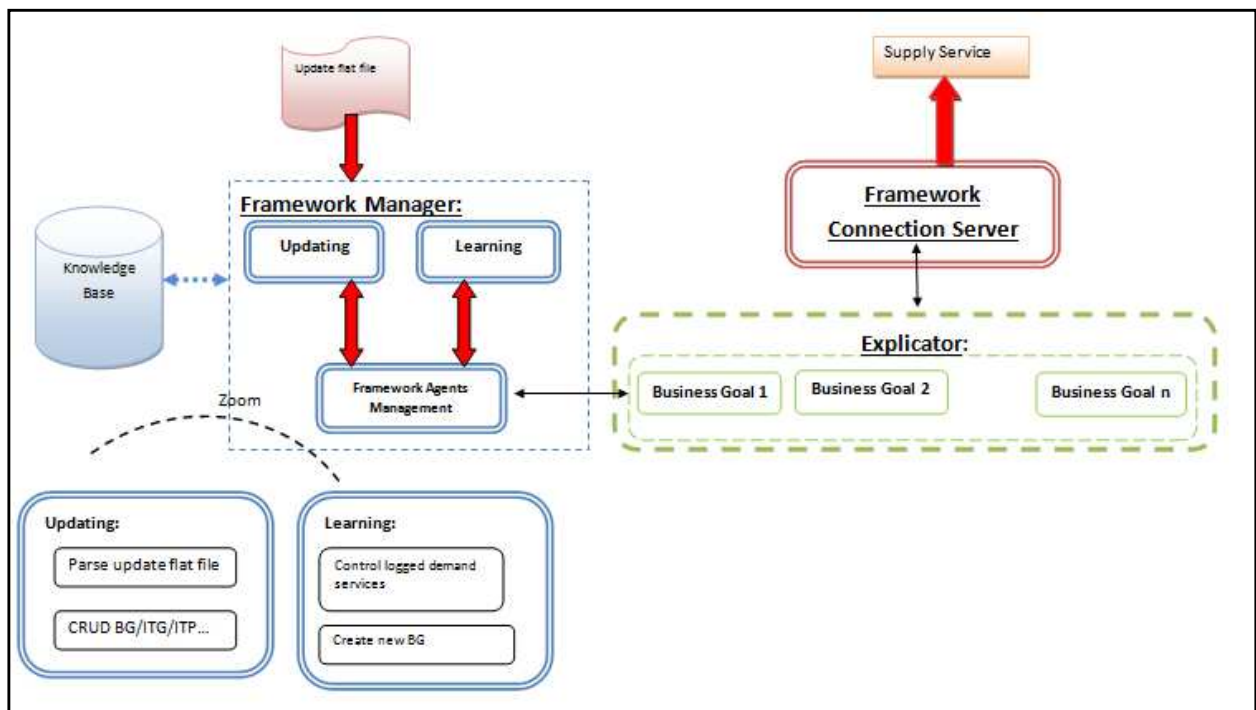


Figure 36: Couche Framework de l'architecture proposée version 3

D'autre part, parmi les limitations de la deuxième version, le fait qu'elle se base à 100% sur le référentiel COBIT. Certes COBIT est un leader des référentiels de la gouvernance des SI au niveau stratégique, cependant une personnalisation selon la maturité de l'entreprise s'impose pour une plateforme plus proche du contexte et plus efficace. En effet, nous avons pensé à une couche de référence ayant un noyau COBIT, vu sa puissance en GTI, auquel nous avons ajouté d'autres

objectifs métiers, d'autre processus IT et d'autres métriques provenant de deux sources différentes :

1. Mise à jour du référentiel (nouvelle version COBIT ou fusion avec d'autres référentiels du niveau stratégique)
2. Objectifs métiers et processus IT provenant du savoir faire de l'entreprise via la solution sujet de ce travail ou proposée par la direction des systèmes d'informations.

Pour concrétiser cet aspect, nous avons remplacé les agents COBIT par des agents objectifs métiers et l'agent Framework Manager par un système multi-agents Framework Management, constitué de trois agents :

- **Agent Manager** qui garde les fonctionnalités de gestion des agents objectifs métiers
- **Agent Update** responsable de la mise à jour du référentiel utilisé. nous rappelons que dans l'itération $t=0$, nous avons le référentiel COBIT dans sa version 4.1.
- **Agent Learning** qui se charge d'apprendre et de journaliser les demandes traitées pour enrichir le noyau et avec possibilité d'ajout de nouveaux objectifs et processus par la DSI.

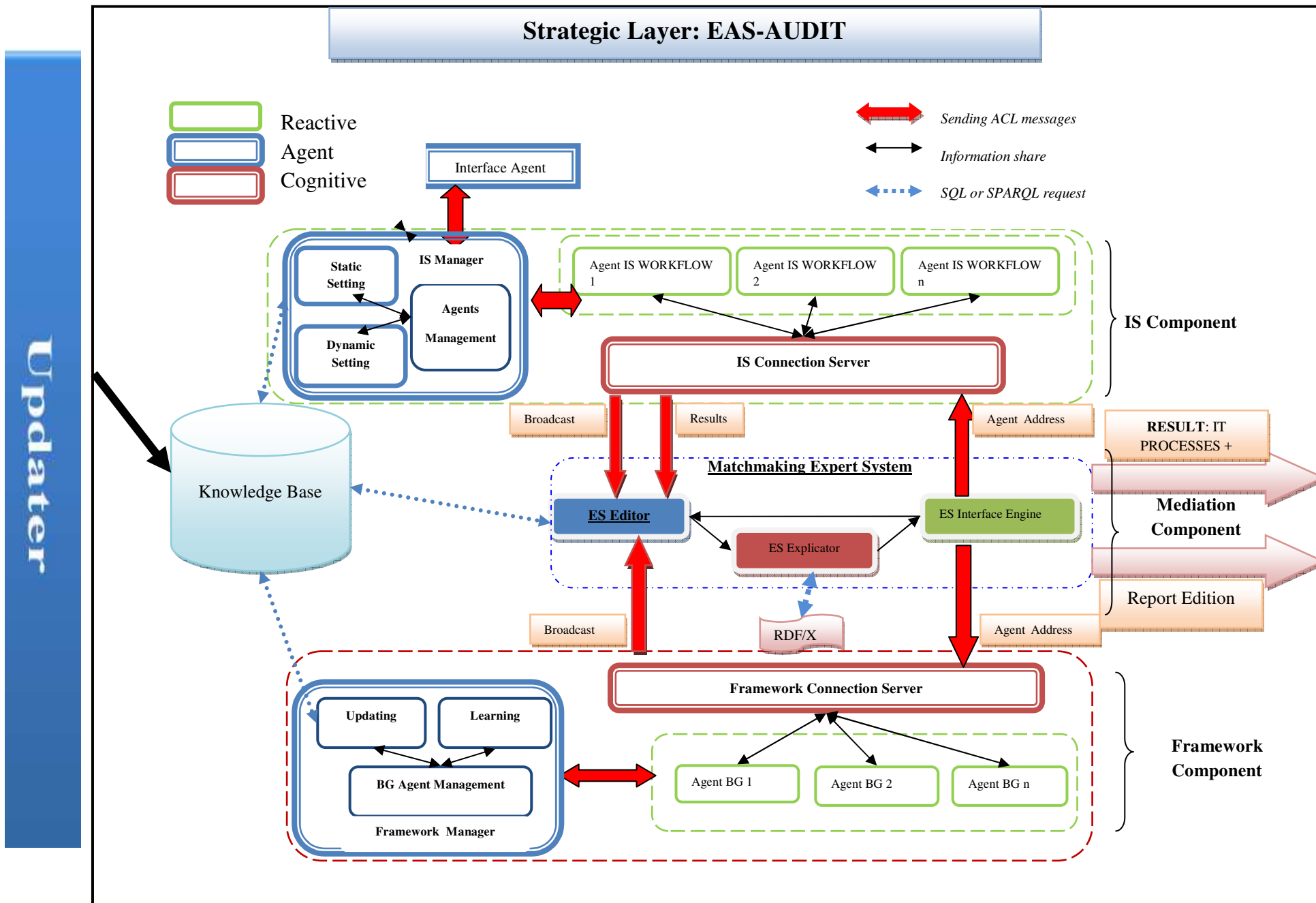


Figure 37: Architecture proposée version 3

III.6. Conclusion

Dans ce chapitre, nous avons proposé une architecture d'une plateforme de gouvernance des systèmes d'information basée sur un workflow inter-organisationnel et des systèmes multi-agent. Cette solution représente la couche stratégique de la plateforme globale de la Gouvernance Risque Conformité IT de l'équipe EAS.

Parmi les avantages de cette architecture, nous citons :

- L'adaptation à un SI quelconque grâce à son aspect inter-organisationnel lâche,
- La compréhension des demandes de toutes les parties prenantes du SI sans pré-requis en matière de Gouvernance IT, grâce au moteur sémantique proposé,
- L'interprétation de ces demandes en langages compréhensibles par l'ensemble des référentiels de gouvernance risque et conformité IT grâce au système expert de médiation disposant d'un noyau COBIT,
- L'apprentissage des demandes passées et la mise à jour du référentiel grâce à des agents intelligents dédiés.

CHAPITRE 4 :
NOUVELLE ONTOLOGIE DE
GOUVERNANCE SI
« ITGovernanceOntology »

IV. Nouvelle Ontologie de gouvernance IT « ITGovernanceOntology ».

IV.1. Introduction

Dans ce chapitre, nous proposons une ontologie de gouvernance des SI pour l'interprétation des offres et des demandes au niveau de la couche médiation du WIO. Pour se faire, nous définissons les ontologies et le web sémantiques et nous présentons les types, les langages de représentation et les outils de construction dans la première partie. Dans la deuxième partie, nous proposons une méthode de conception de l'ontologie GSI et nous décrivons ses différentes phases, à savoir la spécification, l'acquisition de connaissance, la conceptualisation, l'implémentation et l'évaluation.

IV.2. Web Sémantique et Ontologies

L'ontologie désigne la science décrivant les types d'entités dans le monde et la façon dont elles sont reliées (Peter, Schneider, Hayes, & Horrocks, 2004). Elle est souvent réalisée à des fins de raisonnement avec des avantages d'ingénierie portant sur la communication des parties prenantes, l'interopérabilité des systèmes et la réutilisation. La méta-modélisation ontologique est de nos jours, une nécessité pour chaque discipline pour partager des concepts et des termes et annoter des informations. Elle clarifie la structure des connaissances pour une meilleure compréhension du vocabulaire du domaine de ses entités et de ses relations.

Dans l'absence d'un support académique et d'un fondement scientifique en matière de gouvernance des systèmes d'information, nous proposons la conception et le développement d'une ontologie du domaine de la tâche (Hernandez, 2005) basée sur le cadre de référence COBIT nommée: "IT Governance Ontology", utilisée dans la solution de gouvernance IT sujet de ce travail de recherche.

Nous avons proposé un processus générique de construction d'ontologie (Chergui, Sayouti, & Medromi, 2015), spécifique au domaine de la gouvernance risque et conformité des systèmes d'information et qui peut être appliqué à d'autres référentiels. En effet, jusqu'à présent, il n'y a pas d'ontologie propre à la gouvernance des systèmes d'information, malgré la richesse de ce domaine et son utilisation de plus en plus courante.

Les technologies de l'information ont un impact sur le chiffre d'affaires des entreprises et font des différences sur leur fonction d'évolution. Le système d'information étant un investissement important face à l'agilité du marché mondial, il représente un atout permettant à l'entreprise d'atteindre ses objectifs métiers en maîtrisant les coûts et les délais. Cependant, cet investissement n'est pas toujours un succès en l'absence d'une bonne gouvernance. Les experts du système d'information ont mis à la disposition des entreprises les meilleures pratiques et cadres de références, capables de contrôler leurs décisions à propos des projets informatiques. La GSI est une tâche complexe sur le plan pratique qui dépend du contexte et du métier de l'entreprise. Le défi est de comprendre ces référentiels et être en mesure de les adapter aux besoins courants.

En conséquence, nous avons besoin d'une méta-modélisation de la structure logique des référentiels GSI pour une intégration efficace. D'où la nécessité de construction d'une ontologie dans ce domaine, permettant de donner un haut niveau d'abstraction aux composantes des référentiels. Cette démarche est à l'instar des autres domaines qui ont eu recours aux ontologies pour les mêmes raisons de représentation des connaissances, telles que : SNEMOD (Stearns, Price, Spackman, & Wang, 2001) dans le domaine médical, UNSPSC pour la terminologie des produits et services (unspsc, 2014).

Le web sémantique, quant à lui, est un environnement créé par le *World Wide Web Consortium* en 1988, ayant pour objectif l'échange des données et des connaissances à travers le web. Il permet de comprendre les pages web et les données stockées pour répondre aux demandes des humains sans l'assistance d'une personne physique. Les ontologies représentent le noyau incontournable du web sémantique (BOURHIS, 2010), dont la finalité est d'explicitier les données aux agents logiciels via des descriptions standardisées pour comprendre le contenu sémantique du contexte.

Nous construisons dans ce chapitre l'ontologie « ITGovernanceOntology » en utilisant le référentiel COBIT.

IV.2.1. Ontologie

La littérature a donné de nombreuses définitions à l'ontologie. La plus pertinente et à proximité de notre contexte de travail est celle de Gruber qui définit l'ontologie comme «une spécification de conceptualisation explicite ».

Une ontologie est aussi définie comme une spécification formelle et implicite d'une conceptualisation partagée" (Zemmouri, Behja, Marzak, & Trousse, 2011). En effet, il s'agit d'une représentation des connaissances permettant la structuration du vocabulaire et son exploitation autour d'un consensus bien défini.

Une ontologie avec des instances de classes individuelles représente une base de connaissances, la classe est l'objet principale de toute ontologie, une classe peut avoir des sous-classes pour détailler le concept de la superclasse. La relation entre la sous-classe et la superclasse est une relation de « subsomption » relation ou « est-une » relation. Il existe d'autres types de relations définies dans le contexte de la conception d'une ontologie. La classe de l'ontologie a également des propriétés qui la caractérisent, ces propriétés sont soit des propriétés de données ou des propriétés de l'objet.

Les ontologies en informatique sont construites en ingénierie de connaissance, en intelligence artificielle, en gestion des systèmes d'information ou en web sémantique, et dans tous ses volets une conceptualisation est indispensable.

IV.2.2. Types d'ontologies

Il existe de nombreux types d'ontologies dans la littérature du web sémantique, les plus courantes sont notamment (Charlet, Bachimont, & Troncy, 2004) :

- L'ontologie générique: décrit des concepts très généraux tels que l'espace, le temps, la matière, les objets, les événements et les actions, qui sont indépendants d'un problème ou d'un domaine d'application particulier.

- L'ontologie de domaine: son vocabulaire est lié à un domaine en spécialisant les concepts présentés dans l'ontologie générique telle que l'électronique et l'automobile.
- L'ontologie de la tâche: décrit le vocabulaire lié à une tâche ou une activité spécifique.
- L'ontologie de méthode: explicite le rôle joué par chaque concept en détaillant son argument.
- L'ontologie de l'application ou du domaine de la tâche: décrit le concept d'un domaine et d'une tâche particulière, qui sont souvent deux spécialisations des ontologies connexes.

Nous pouvons éventuellement représenter les relations hiérarchiques entre les différents types d'ontologies dans la figure ci-dessous:

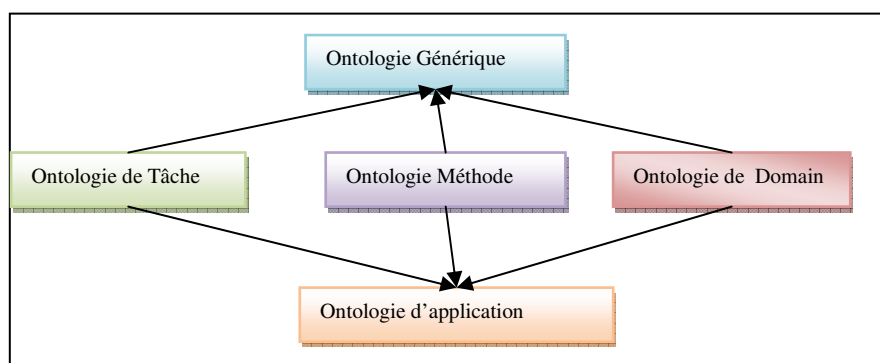


Figure 38: Relation entre les types d'ontologies

IV.2.3. Langages de représentation et outils de construction

IV.2.3.1. Langages de représentation des ontologies

L'implémentation d'une ontologie repose sur la construction d'un modèle formel représenté dans un langage ontologique. Il existe principalement deux classes de langage de représentation d'ontologie, langages classique et langages à balisage.

Pour la première classe, nous citons à titre d'exemple :

- Ontolingua: pour ontologies portables, définit les classes, relations et fonctions sur KIF (Knowledge Interchange Form) formalisme (BOURHIS, 2010) et est capable de traduire les ontologies génériques au métier. C'est un langage déclaratif qui s'appuie sur le premier calcul des prédicats.
- Loom: un langage de programmation de haut niveau et de l'environnement destiné à être utilisé dans la construction de systèmes experts et d'autres programmes d'application intelligents. Basé sur l'enchaînement, dans l'unification sémantique et les technologies orientées objet pour fournir un soutien déductif.
- OIL: couche d'inférence ontologique est un langage ontologique pour la représentation en combinant la sémantique formelle et les méthodes descriptive de raisonnement logique. OIL est largement utilisé pour le web, basé sur RDF / RDFS et XML formalisme (Klein, Fensel, van Harmelen, & Horrocks, 2000).
- SHOE : Simple extensions HTML pour les ontologies qui permet aux auteurs de pages Web de générer les annotations de leurs documents pour être compris par une machine. Ce langage peut être utilisé par des agents (Luke & Heflin, 1997).

Pour la deuxième classe, il ya beaucoup de langages et de formalismes utilisés pour représenter les ontologies, le W3C a proposé la pyramide de langages de représentation ci-dessous :

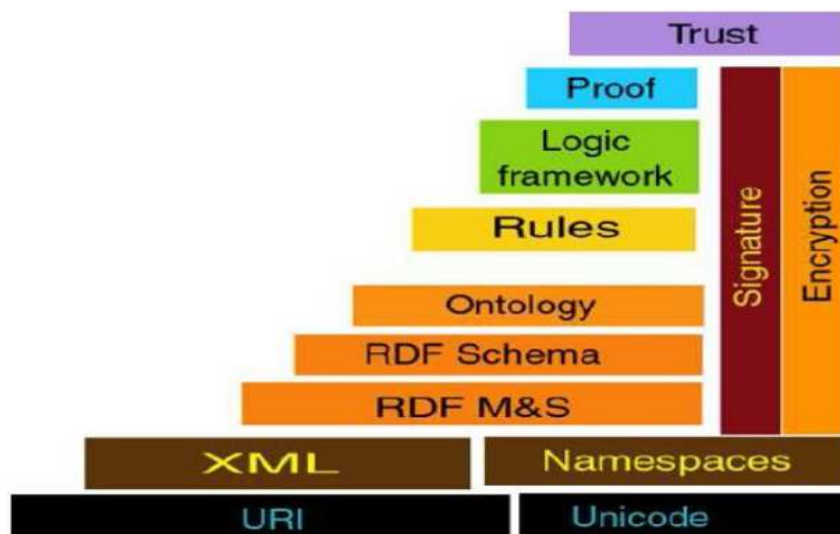


Figure 39: Langages de représentation d'ontologie W3C

Cependant, seuls certains ont réussi à représenter efficacement les ontologies dans le web, à savoir :

XML : langage de balisage proposant une syntaxe de description de la structure d'un document, mais aussi de sa création et sa manipulation. Il utilise un espace de nommage (Namespace) permettant d'identifier des balises utilisées. Le document est validé et définit par un schéma. Cependant, XML n'impose aucune contrainte sémantique à la signification de ces documents. XML ne suffit donc pas pour assurer aux logiciels la compréhension du contenu des données incluses dans le document.

RDF (Resource Description Framework) : Langage de description sémantique complémentaire au XML développé par W3C. C'est un mécanisme normalisé et interopérable pour décrire les ressources sans hypothèse sur la structure du document ni son contexte. Il contient principalement les ressources ou entités repérés par une adresse www, les propriétés décrivant ces entités et les objets attribuant une valeur à la propriété pour une ressource donnée.

OWL (Web Ontology Language) : langage de représentation de connaissances spécifique aux ontologies web, basé sur le modèle de données RDF, sa deuxième version est la recommandation du W3C, il dispose de trois sous-langages plus expressifs: OWL Lite, OWL DL et OWL Full (Charlet, Bachimont, & Troncy, 2004). Il dispose également d'un langage de requêtes adéquats SPARQL . Il est reconnu par ses forces de structuration de XML et d'une sémantique formelle compréhensible par les machines par rapport aux contenus du web.

Pour l'implémentation de l'ontologie IT Governance Ontolgy nous avons opté pour OWL vu sa richesse, son expressivité mais également vu son interopérabilité avec les autres langages de représentations d'ontologie.

IV.2.3.2. Outils de construction des ontologies

Il existe deux grandes catégories d'outils de constructions :

1. Des outils classiques qui se focalisent sur le niveau symbolique de l'ontologie (exemple serveur Ontlingua), ces outils imposent le langage et sa structure au créateur de l'ontologie.
2. Des outils qui se focalisent sur le niveau des connaissances et permettent de créer l'ontologie indépendamment du langage. L'opérationnalisation se fait automatiquement.

A titre d'exemple, nous citons les outils les plus utilisés :

- Protégé (Noy, et al., 2003) est un environnement graphique, pour le développement des ontologies, basé sur le modèle de connaissance hiérarchique (classes attributs, propriétés). Il est l'un des éditeurs les plus utilisés regroupant une large communauté d'utilisateurs. Il a une compatibilité avec la référence OWL, la gestion de la base de connaissances, ontologies visualisations, alimentation et fusion.
- OILED (Bechhofer, Horrocks, Goble, & Stevens, 2001) est une plateforme basée sur la hiérarchie des classes, elle fournit des spécialisations de rôles, et teste les définitions, mais elle est restreinte à la construction d'ontologies OIL.
- OntoEDit (Sure, Erdmann, Angele, Staab, Studer, & Wenke, 2002) : est une solution propriétaire basée sur des concepts hiérarchiques, capable d'exprimer les axiomes. Mais elle est limitée à une comparaison lexicale des termes.

IV.2.3.3. Méthodes de conception d'ontologies

Dans la littérature du web sémantique, il n'existe pas de cycle de vie unifié des méthodologies et des techniques pour construire des ontologies. Mais les auteurs examinent trois types d'approches pour concevoir une ontologie formelle (Van Der Vet & Mars, 1998) : de bas en haut; de haut en bas; approche moyenne (Uschold & Gruninger, 1996).

Plus tard, Gomez-Perez (Fernández-López, Gómez-Pérez, & Juristo, 1997) propose une méthode appelée METHONTOLOGY, qui passe par les étapes suivantes: la spécification, l'acquisition de connaissance, la conceptualisation, l'intégration, l'implémentation, l'évaluation, la maintenance et la documentation.

Il y a d'autres méthodes de conception d'ontologie telles que TOVE (Gruninger, Schlenoff, Knutilla, & Ray, 1997), mais cette approche est spécialisée dans l'évaluation des ontologies, après l'établissement de compétences de scénarios existants. D'autres auteurs ont simplement fait des critères d'acceptation (Lozano-Tello & Gómez-Pérez, 2004) ou des difficultés de construction.

IV.2.4. Synthèse

Pour la conception et le développement de l'ontologie « ITGovernanceOntology » que nous utilisons au niveau du moteur d'inférence du système expert de médiation, nous avons opté pour la méthode METHONTOLOGY que nous avons adapté au contexte de la gouvernance SI. Nous

avons ainsi proposé une approche générique de modélisation et de développement d'ontologie de gouvernance risque et conformité SI.

Pour le développement, nous avons opté pour le langage OWL-S et l'outil Protégé 4.3 avec son éditeur OWL VIZ.

IV.3. Modélisation « ITGovernanceOntology »

IV.3.1. Introduction

ITGovernanceOntology est une ontologie de domaine de la tâche autour de la gouvernance des systèmes d'information.

ITGovernanceOntology est le résultat de consolidation de nombreuses sources d'information. En l'absence d'une ontologie semi-formelle de gouvernance des TI ou IT ; qui fait dans la plupart des cas le point de départ de mise en place des ontologies formelles, quelque soit leur type ; nous construisons ITGovernanceOntology à partir de/de l'/de la:

1. Indexation de COBIT: nous choisissons MAUI indexeur, un projet open source avec la licence GNU, efficace pour l'indexation de texte, en mesure de résoudre les mots-clés et thésaurus principal dans de nombreux domaines (Medelyan, 2009) ;
2. Méta-modèles existants du référentiel COBIT, essentiellement basés sur la méthodologie entité / relation, à savoir, le modèle officiel COBIT architecture (Gerrard, 2009) ;
3. Description des processus IT de COBIT pour la description détaillée des relations et des cardinalités et éventuellement pour la validation pratique.

Nous avons opté pour METHONTOLOGY comme méthodologie de conception et nous avons ajouté successivement les étapes précédentes (1-2 et 3) entre l'acquisition des connaissances et la conception. Nous avons donc comme étapes de construction d'ontologie de gouvernance SI (Chergui, Sayouti, & Medroumi, 2015) :

- | | |
|--|-----------------------|
| 1. Spécification, | 6. Conceptualisation, |
| 2. Acquisition de connaissances, | 7. Intégration, |
| 3. Indexation du référentiel, | 8. Implémentation, |
| 4. Consolidation des méta-modèles du référentiel, | 9. Evaluation, |
| 5. Description de l'entité élémentaire du référentiel, | 10. Maintenance, |
| | 11. Documentation. |

Ces étapes peuvent être utilisées pour construire les ontologies de tous les frameworks IT GRC. Elles représentent une modélisation formelle ou une semi "Ontologisation" qui sera suivie d'une traduction en langage officiel ("opérationnalisation") (Luke & Heflin, 1997).

Il est à noter que « ITGovernanceOntology » est ouverte à l'enrichissement, car elle est développée en OWL.

IV.3.2. Spécifications

Dans la phase de spécification, nous définissons les objectifs de l'ontologie, son type, son niveau de formalité et son champ d'application.

ITGovernanceOntology est une ontologie formelle de tâche dont le domaine est la gouvernance des technologies de l'information, son principal objectif est de guider les utilisateurs des systèmes d'information pour évaluer leurs objectifs métiers grâce au référentiel COBIT. Ceci pour obtenir:

- les mesures des processus IT ;
- les responsabilités ;
- les activités clés ;
- les ressources informatiques sollicitées ;
- les contrôles et tests.

ITGovernanceOntology décrit le vocabulaire de la gouvernance des TI, en plus des fonctions et des méthodes effectuées par les composantes de COBIT, indépendamment du contexte de l'entreprise et de son SI.

Cette ontologie répond à des questions telles que :

- Quel est le niveau de maturité atteint par l'entreprise?
- Que devrions-nous faire pour l'améliorer?
- Qui est le responsable pour un objectif IT ou un processus IT d'une entreprise donnée?
- Quel processus IT est convenable pour un objectif métier donné?

En bref, ITGovernanceOntology soutient les opérations et les méthodes de tous les composantes de COBIT pour mesurer l'alignement stratégique du SI avec le métier, exiger des valeurs et bien gérer les ressources humaines et matérielles.

IV.3.3. Acquisition de Connaissances

Dans cette étape, nous procédons à la consolidation des connaissances pour effectuer la conceptualisation de l'ontologie, nous débutons l'acquisition des connaissances à l'étape de spécification et nous l'étalons à l'étape de conceptualisation.

Etape 1 : Nous examinons une liste de mots-clés et vocabulaire produite par un générateur de mots clé appliqué aux référentiels COBIT dans sa version 4.1. En effet, notre choix s'est porté sur MAUI indexer comme algorithme open source réputé par la qualité des mots clés générés pour un input de taille moyenne (Seifert, 2011), cet outil est présent sous forme de web service permettant l'/le:

- Affectation à long terme avec un vocabulaire contrôlé (ou thésaurus) ;
- Indexation par sujet ;
- Indexation d'un sujet à partir des termes de Wikipedia ;
- Extraction des phrases clés ;
- Extraction de terminologie ;
- Tagging automatique (Wiederhold, 1992);

- Extraction de la terminologie et indexation semi-automatique du sujet.

Les résultats obtenus sont les suivants:

	Concepts	Actions
IT ressource	activities	Evaluate
level of maturity	applications	Acquire and Implement
management and control	business	changes
management guidelines	business goals	practices
management process	business requirements	Acquire
maturity	COBIT	Manage
procedures	control	include
program	control objectives	provides
RACI chart	development	Ensure
requirements	development process	define
resources	enterprise	Define
responsibilities	framework	Monitor and Evaluate
monitoring	goals	measure
risks	governance	monitor
security	information	control
performance	information criteria	Evaluate
service levels	internal control	Acquire and Implement
solutions	investment	changes
standards	tactical plans	practices
strategic		Acquire

Tableau 4.1: Résultat de l'indexeur MAUI appliqué à COBIT 4.1

Etape 2 : il est à signaler qu'il y a un manque de fondement théorique en ce qui concerne les référentiels de gouvernance des systèmes d'informations, malgré leur forte application en pratique. Nous avons donc étudié les méta-modèles existants et réexaminer la documentation de COBIT pour soulever toutes les questions auxquelles cette ontologie peut répondre afin d'acquérir toutes les connaissances à mettre en jeu (Chergui, Sayouti, & Medromi, 2015).

Dans le domaine des SI, nous utilisons des modèles afin de faire abstraction de phénomènes et des objets du monde réel. La représentation de modèle est généralement la première étape d'élaboration d'un système d'application ou d'un logiciel.

Si les sujets de recherche sont des modèles, et non le monde réel, nous créons les modèles de modèles. Habituellement, un «modèle d'un modèle », qui est un niveau d'abstraction plus élevé, est appelé « méta-modèle ».

Afin de décrire la syntaxe abstraite d'un domaine, habituellement les méta-modèles linguistiques sont appliqués. Ils définissent les éléments de langage disponibles et les relations entre eux ainsi que leur signification et leurs règles génératives (Ferstl, 2006). L'un des méta-modèles les plus répandus est le modèle entité association, il se caractérise par son application étendue dans des domaines différents et aussi par sa simplicité et sa cohérence.

Le référentiel COBIT étant la matière première de notre ontologie, contient des concepts et des entités et schématise des relations entre eux. Le diagramme (Figure 35) représente la liaison entre les concepts de base du référentiel COBIT.

En plus de ce qui précède, les avis et le savoir faire des experts en gouvernance des systèmes d'information nous permettent de reformuler les questions de compétences auxquelles l'ontologie doit répondre.

Parmi les questions auxquels l'ontologie peut répondre :

1. Quels objectifs métiers COBIT correspondent à cette demande?
2. Quels sont les objectifs informatiques liés à un objectif métier?
3. Quels processus IT devrions-nous déployer?
4. Quel est le domaine contenant un processus IT donné ?
5. Quel domaine de gouvernance IT est primaire par rapport à un processus IT et qui est secondaire ?

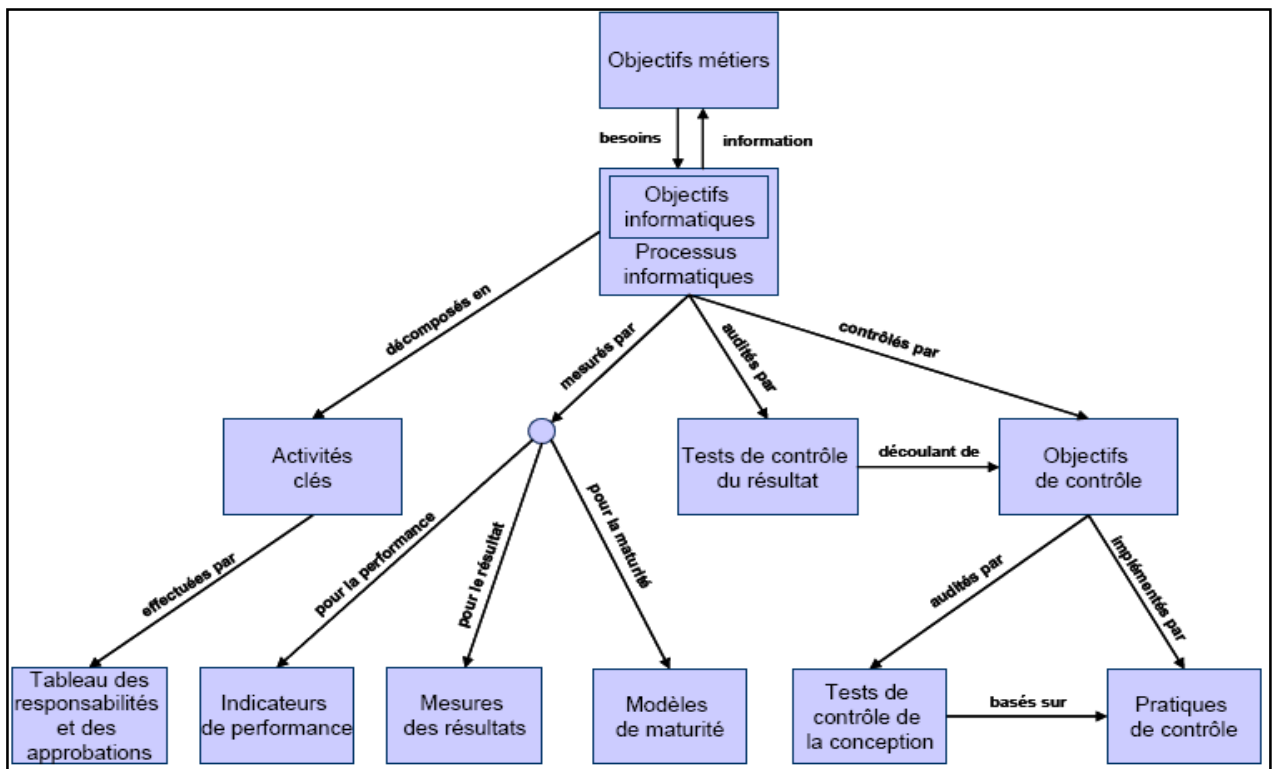


Figure 40: Relation entre les composants de COBIT

6. Quel contrôle devrions-nous appliquer à chaque processus IT?
7. Quelles mesures sont appropriées pour chaque commande?
8. Quelles sont les activités clés que nous devrions exécuter pour un processus -IT? Et qui est le responsable de ces activités?
9. Quel est le niveau et le modèle de maturité atteint par un processus-IT donné?

IV.3.4. Conceptualisation

Dans cette étape, un diagramme de classes du cadre de référence doit être établi ; dans le cas de COBIT nous avons fait le schéma Figure 36.

D'après les étapes précédentes, nous étions en mesure de définir les concepts de base suivants :

- **Domaine informatique:** c'est un regroupement cohérent de processus. COBIT contient quatre domaines.
- **Processus IT:** c'est un ensemble d'activités corrélées qui transforme des éléments entrants en éléments sortants. COBIT contient trente quatre processus IT, qui appartiennent aux 4 domaines.
- **Activité:** tout processus informatique est divisé en de nombreuses activités, ayant une vue de la gestion.
- **Contrôle IT:** c'est une vue opérationnelle pour les auditeurs pour s'assurer de l'exécution des objectifs métiers.
- **Objectif:** c'est un indicateur-clé qui définit par un processus informatique, mesuré par une métrique.
- **Modèle de maturité:** c'est la déclinaison spécifique capable de mesurer tous les processus IT et leur amélioration.
- **Niveau de maturité:** c'est est une valeur numérique entre 0 et 5 pour mesurer le processus par rapport au modèle de maturité.
- **Critères d'information:** c'est la segmentation des informations selon 7 critères: efficacité, l'efficacité, la confidentialité, l'intégrité, la disponibilité, la conformité et la fiabilité.
- **Resource IT:** c'est l'une des ressources de l'entreprise qui est soit une application, soit une infrastructure, soit une information, soit une personne.
- **Secteur de gouvernance TI:** c'est l'une des cinq zones de commande de focalisation du COSO: alignement stratégique, livraison de la valeur, gestion des risques, gestion des ressources et mesure du rendement.
- **Résultat:** c'est un ensemble de processus IT produit par un autre processus IT.
- **Rôle:** c'est l'acteur responsable de la mise en œuvre d'une activité clé.

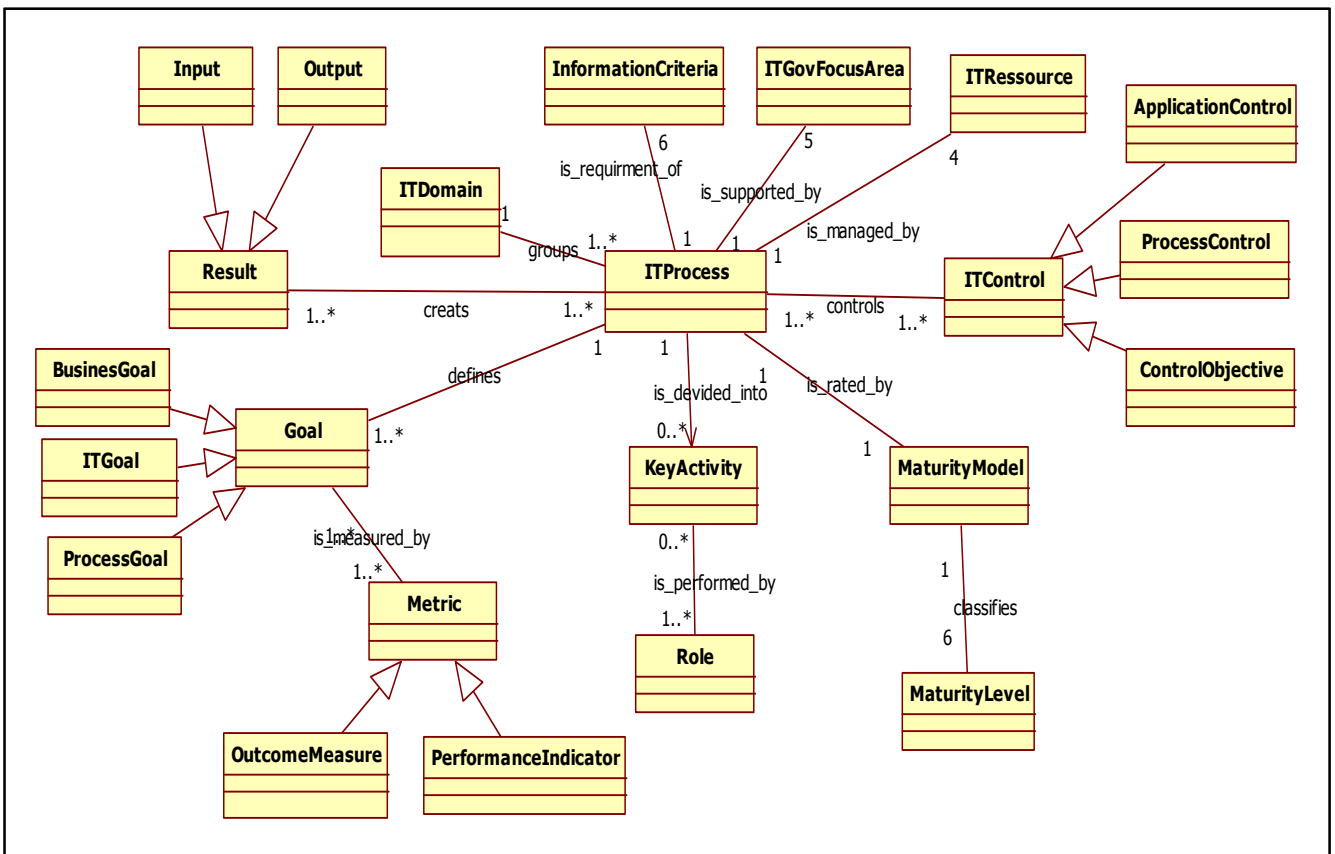


Figure 41: Le diagramme de classes des concepts COBIT

Quant aux propriétés, il existe deux types :

- des propriétés de données.
- des propriétés de l'objet.

Les propriétés de données sont des propriétés de description des concepts: exemple :

- ITProcess (String processCode, String processDescription).
- ITDomain (String domainCode, String domainDescription).

Les propriétés de l'objet sont les relations qui relient les objets. Parmi les principales propriétés de l'objet, nous citons:

- **Est un:** relation de généralisation ;
- **Est vérifié avec:** relation entre le contrôle IT et le test de contrôle ;
- **Est classé par:** relation entre le modèle de maturité et le niveau de maturité ;
- **Est contrôlé par:** relation entre processus IT et Contrôle IT ;
- **Est mesurée par:** relation entre objectif IT et métrique ;
- **Est classé par:** relation entre processus IT et modèle de maturité ;
- **Est soutenu par:** relation entre processus IT et secteur de gouvernance TI.
-

IV.3.5. Implémentation

Pour mettre en œuvre ITGovernanceOntology, nous choisissons OWL-DL comme langue d'ontologie web et Protégé 4.3 en tant qu'éditeur d'ontologie. Nous avons procédé comme suit :

Etape 1: Définition de l'ontologie active et sa description ;

Etape 2: Création des classes et sous-classes ;

Etape 3: Création d'objets Propriétés ;

Etape 4: Création des propriétés de données ;

Etape 5: Création des individus .

Nous obtenons les résultats suivants:

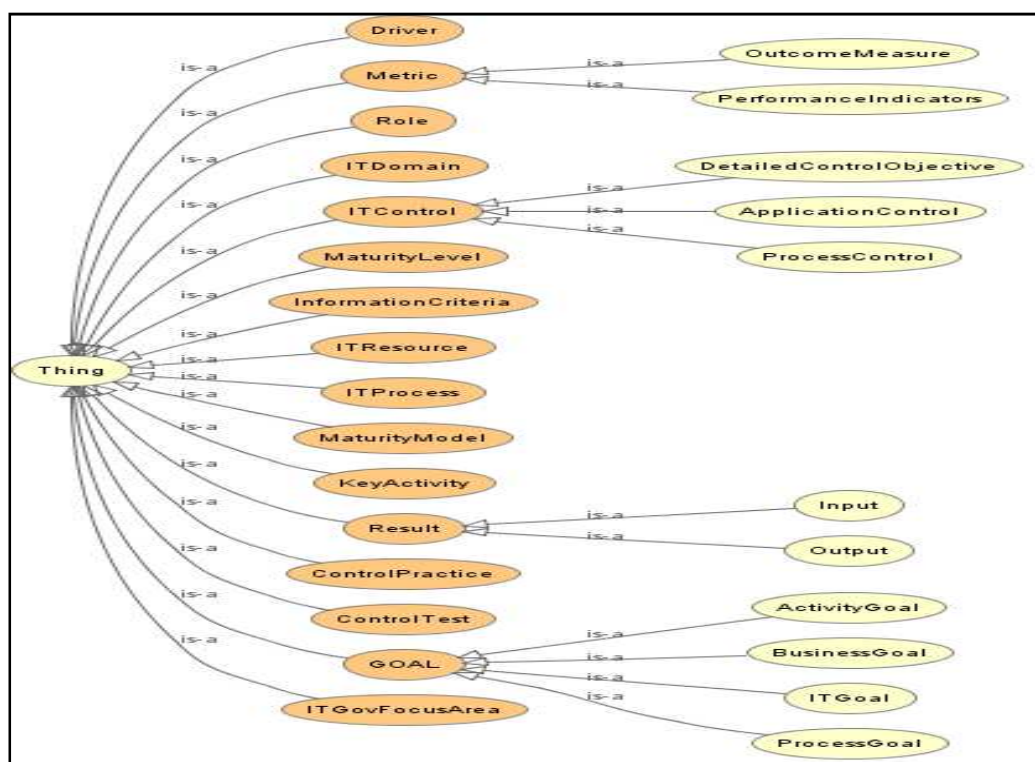


Figure 42 : Vue d'IT Governance Ontology sur OWLViz

Le fragment de code présenté sur la figure ci-dessous montre le code d'un ensemble de propriété d'objet de l'ontologie ITGovernanceOntology, en format RDF/XML,

```

<rdf:RDF xmlns="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#"
  xml:base="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  <owl:Ontology rdf:about="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1">
    <rdfs:comment>AuditOntology is a formal ontology describing Audit operation through COBIT 4.1
      IT Governance Framework, its Components relationship and methods </rdfs:comment>
    <owl:versionInfo>This is AuditOntology version 1</owl:versionInfo>
  </owl:Ontology>
  <!--
  // Object Properties
  <!-- http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#audits -->
  <owl:ObjectProperty rdf:about="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#audits">
    <rdfs:domain rdf:resource="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#ControlTest"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#ITControl"/>
    <owl:inverseOf rdf:resource="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#is_audited_with"/>
  </owl:ObjectProperty>
  <!-- http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#belongs_to -->
  <owl:ObjectProperty rdf:about="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#belongs_to">
    <rdfs:range rdf:resource="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#ITDomain"/>
    <rdfs:domain rdf:resource="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#ITProcess"/>
  </owl:ObjectProperty>
  <!-- http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#classifies -->
  <owl:ObjectProperty rdf:about="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#classifies">
    <rdfs:domain rdf:resource="http://www.semanticweb.org/cherqui/ontologies/2014/10/AuditOntologyV1#MaturityLevel"/>
  </owl:ObjectProperty>
  </rdf:RDF>

```

Figure 43: Aperçu du code OWL d'IT Governance Ontology

Dans l'annexe 3 nous présentons des captures de l'ontologie à l'aide des plugins de Protégé.

IV.3.6. Evaluation et synthèse.

Dans cette phase nous avons effectué une évaluation de l'ontologie proposée, il s'agit d'une évaluation de l'intégrité via un moteur d'inférence pour vérifier la cohérence de l'ontologie. Il existe de nombreux moteurs d'inférence comme, Jena, Ermite, Pellet et Fact++. Dans notre cas, nous choisissons Fact ++ Plug-in de Protégé pour tester les inconsistances d'ITGovernanceOntology et à travers ses propositions nous élaborons les classes présumées. La figure ci-dessous illustre la hiérarchie des classes de l'ontologie ITGovernanceOntology après vérification, ainsi que les classes d'équivalence inférées grâce à Fact++.

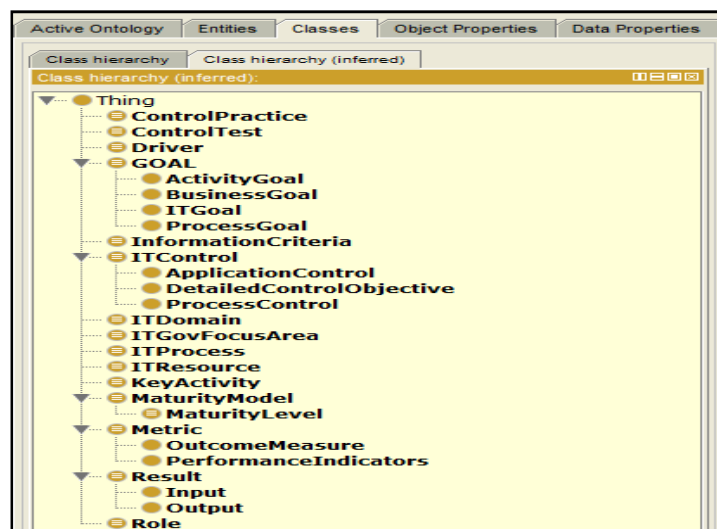
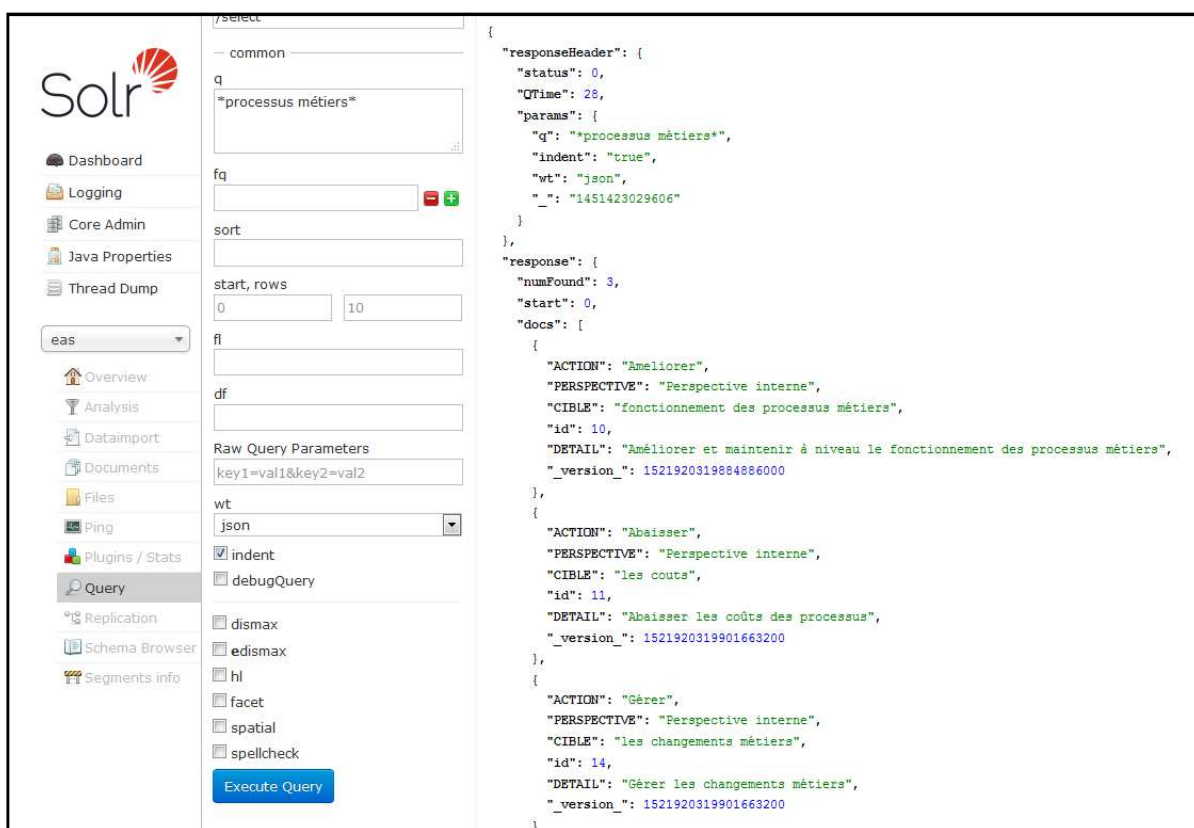


Figure 44: Classes inférées par Fact ++ d'IT Governance Ontology

Pour l'opérationnalisation d'ITGovernanceOntology et en guise de test, nous avons paramétré le serveur Solr 5, pour effectuer les requêtes sémantiques appropriées. Pour se faire, nous avons procédé comme suit :

1. Installation du serveur Solr 5 et SolRDF.
2. Création du Core « eas » lié à l'ontologie générée (fichier ITGovernanceOntology.rdf) et son paramétrage.
3. Import des individus de l'ontologie « ITGovernanceOntology » sur le serveur Solr.
4. Lancement des requêtes.

Ci-dessous les résultats des requêtes :



The screenshot displays the Solr Admin interface. On the left is a navigation menu with options like Dashboard, Logging, Core Admin, Java Properties, Thread Dump, Overview, Analysis, Dataimport, Documents, Files, Ping, Plugins / Stats, Query, Replication, Schema Browser, and Segments info. The main area shows a query configuration for the 'eas' core. The query is 'processus métiers*' with 'indent' checked. The response is a JSON array of three documents, each with fields for ACTION, PERSPECTIVE, CIBLE, id, and _version_.

```
q=*processus métiers*
indent
wt=json
response={
  "numFound": 3,
  "start": 0,
  "docs": [
    {
      "ACTION": "Améliorer",
      "PERSPECTIVE": "Perspective interne",
      "CIBLE": "fonctionnement des processus métiers",
      "id": 10,
      "DETAIL": "Améliorer et maintenir à niveau le fonctionnement des processus métiers",
      "_version_": 1521920319884886000
    },
    {
      "ACTION": "Abaisser",
      "PERSPECTIVE": "Perspective interne",
      "CIBLE": "les coûts",
      "id": 11,
      "DETAIL": "Abaisser les coûts des processus",
      "_version_": 1521920319901663200
    },
    {
      "ACTION": "Gérer",
      "PERSPECTIVE": "Perspective interne",
      "CIBLE": "les changements métiers",
      "id": 14,
      "DETAIL": "Gérer les changements métiers",
      "_version_": 1521920319901663200
    }
  ]
}
```

Figure 45: résultats requêtes ITGovernanceOntology sur Solr

Nous détaillons dans le chapitre suivant, l'exploitation effective de l'ontologie « ITGovernanceOntology » dans la plateforme de gouvernance IT sujet de cette thèse.

IV.4. Conclusion

L'objectif de ce chapitre est la modélisation d'une ontologie du domaine de la tâche de la gouvernance des systèmes d'informations, capable de cerner le vocabulaire de ce domaine et les actions qui tournent autour. Dans la première partie de ce chapitre, nous avons défini les ontologies et le web sémantique. Nous avons aussi présenté les langages de représentation des ontologies, leurs outils de construction et leurs méthodes de conception.

Dans la deuxième partie, nous avons modélisé « ITGovernanceOntology » en se basant sur la méthode METHONOTOLOGY qui consiste en la spécification, l'acquisition des connaissances, la conceptualisation, l'implémentation et l'évaluation.

CHAPITRE 5:
SYSTEME EXPERT DE
MEDIATION SEMANTIQUE

V. Système Expert de Médiation sémantique.

V.1. Introduction

Dans ce chapitre, nous proposons l'architecture du système expert de médiation sémantique de la plateforme, où nous avons déployé l'ontologie ITGovernanceOntology.

Pour se faire, nous présentons dans la première partie les systèmes de médiation, en l'occurrence les agents médiateurs. Nous citons les différents algorithmes de matchmaking sémantique existants et nous définissons les systèmes experts comme étant un cas particulier de système multi-agents.

Dans la deuxième partie, nous proposons une architecture du système expert de médiation de notre solution et nous détaillons ses différents composants. Nous l'appliquons ensuite au contexte de la gouvernance SI en explicitant ses requêtes et son algorithme de comparaison.

V.2. Contexte

L'entreprise d'aujourd'hui, exige un système d'information de plus en plus distribué avec des accès aux données optimisés. Le couplage faible est une exigence de modélisation très demandées par les clients du web. Les choix statiques et limités sont dépassés par l'architecture orientée services et l'évolution des systèmes experts.

L'architecture de médiation est l'une des formes de l'architecture en flot de données distribuées proposées par GIO Wiederhold en 1992 (Ives, Hamilton, & Davis, 1980). Il a été conçu pour résoudre la problématique de l'accès et l'intégrité des informations par l'introduction d'une entité de médiation. Le but principal de cette entité est de décoder des informations de connaissances offertes et de contribuer à la couche supérieure de la création de valeur des applications (prise de décision, filtrage de base, traitement des données, répertoire intelligent de création de base de données).

Le matchmaking, comme une forme de médiation, permet à l'utilisateur final de trouver une réponse pratique à ses demandes au sujet d'un domaine spécifique. Les réponses ne sont pas toujours totalement équivalentes à la demande. Mais elles peuvent être incluses ou généralisées. Par conséquent, un système de matchmaking doit répondre à tous les cas possibles.

L'intégration du web sémantique est une amélioration des processus de médiation, pour permettre une meilleure compréhension des transactions avec des pré-conditions et des post-conditions.

L'ontologie étant "une spécification formelle d'une conceptualisation partagée", représente toujours la connaissance autour d'un domaine spécial, en définissant les concepts et les propriétés pour éviter les ambiguïtés taxonomiques (Alavi & Carlson, 1992). De nombreux travaux ont été développés en matchmaking sémantique. Mais, la plupart d'entre eux concerne les services web acheminés autour d'un agent de médiation et se concentre sur la relation «subsumption».

Nous proposons un système expert avec des agents intelligents et un moteur d'inférence sémantique avec différents niveaux de matchmaking (comparaison et priorités).

La conception dudit système a pour objectif la résolution du problème de matchmaking des services IT (offre et demande) dans le domaine de la gouvernance des systèmes d'information. La solution utilise les systèmes multi-agents pour profiter de leur autonomie et coopération, en plus de l'utilisation des protocoles de communication de haut niveau. La spécificité de cette solution par rapport aux systèmes existants de médiation est l'utilisation d'un moteur d'inférence sémantique, faisant référence à une ontologie spécifique et offrant trois modes de comparaison: «égalité», «inclusion» et «généralisation».

L'architecture du système expert de médiation représente la couche médiation du workflow inter-organisationnel de gouvernance IT. En effet, cette couche a été développée selon les étapes suivantes :

- Médiation à l'aide d'un seul agent intelligent « Matchmaker Agent »,
- Système multi-agent de médiation, vu la nécessité de stockage des demandes et des offres en plus de la médiation,
- Système expert de médiation avec une base de connaissance dédiée et un moteur d'inférence sémantique, pour interpréter les demandes selon un référentiel de gouvernance des systèmes d'information.

Nous présentons dans ce qui suit, les concepts et algorithmes essentiels de la médiation et du matchmaking. Ensuite, nous introduisons les systèmes experts, pour enfin proposer l'architecture du système expert de matchmaking sémantique représentant la couche médiation de notre workflow de gouvernance SI, ainsi que sa base de connaissance.

V.3. Systèmes de Médiation

Le matchmaking ou la médiation est l'établissement de la connexion entre les acteurs ou entités demandant un ou plusieurs services. Il offre la capacité de la description des services de pages jaunes entre les demandeurs et les fournisseurs (Northrop, et al., 2006).

Le matchmaking sémantique est un processus de découverte et d'invocation dynamique de services non connus préalablement par le client. Il permet à l'aide d'une ontologie de décrire les services par les caractéristiques suivantes : les entrées, les sorties, les pré-conditions et les effets (Bellur & Kulkarni, 2007).

Dans la littérature, il existe trois modes de matchmaking, égalité, plug-in et subsomption. Pour l'égalité, il est la correspondance exacte, entre une demande et une réponse qui signifie que la demande est totalement répondue. Le deuxième cas est le plug-in, il signifie que la réponse inclut la demande. Dans ce cas, nous parlons d'approvisionnement suffisant et nous mesurons le niveau de satisfaction. Il en est de même dans le cas de subsomption où la réponse est incluse dans la demande ; pour la mesure de la satisfaction, nous définissons l'algorithme de choix de la «meilleure correspondance» pour la requête initiale selon le contexte. Nous ne pouvons pas nier le scénario de l'échec là ou aucune réponse n'existe pour la demande initiale.

Le processus de médiation peut être appliqué à des pré-conditions, ce qui signifie que la comparaison respective des entrées de la demande, de la réponse et la définition des critères pertinents dans l'unité de traitement, est indispensable. Elle peut également être appliquée à la fois avec des pré-conditions et des post-conditions, avec la mesure de la distance dans le contexte sémantique et la comparaison de valeur significative dans d'autres contextes.

Il existe deux catégories de médiation : médiation de schéma et médiation de contexte (Jouanot, 2000):

La médiation de schéma est inspirée du modèle fédéré fortement couplé (Hernandez, 2005), c'est une résolution statique de problème de médiation, basée sur des sources avec des schémas locaux adaptés à un problème global. Il est caractérisé par un plan d'exécution de requêtes prédéfini: elle permet la correspondance de nombreuses demandes d'une source spécifique à d'autres sources ou l'inverse, afin d'obtenir la meilleure réponse pour la requête initiale.

La médiation de contexte, quant à elle, est plus adéquate à des systèmes ouverts où l'apparition et la disparition des sources d'information est continuellement possible. Cette approche est également caractérisée par une prise en charge automatisée de la sémantique, grâce aux mécanismes d'unification ou de réconciliation des contextes. La médiation de contexte est plus liée à des requêtes qui établissent une relation dynamique des sources provenant d'environnements hétérogènes avec un traitement automatisé.

Chaque catégorie a ses avantages et ses inconvénients, comme le montre le tableau ci-dessous :

	Mediation de schéma	Médiation de contexte
Avantages	<p>Meilleur temps d'exécution</p> <p>Basée sur le modèle fédéré</p> <p>Analyse optimale du schéma globale à partir des schémas locaux</p>	<p>Relation dynamique entre les nœuds</p> <p>Aucune donnée d'intégration prérequise.</p>
Inconvénients	<p>Changement conceptuel difficile (ajout de nœud ou suppression de nœud)</p>	<p>Médiation non exacte</p>

Tableau 5.1 : Comparaison des deux modes de médiation

Pour synthétiser, la médiation de schéma offre un meilleur calendrier d'exécution de l'opération de médiation. Mais elle a besoin de changements conceptuels à chaque fois qu'une source est ajoutée ou supprimée. La médiation de contexte n'a pas besoin de changements conceptuels. Elle permet d'établir des liens dynamiques en fonction de la requête initiale. Sa limitation est plutôt la médiation non exacte, qui peut être non conforme à la requête.

Les systèmes de matchmaking sont utilisés dans de nombreux domaines, à savoir:

- La communication en réseau (sans fil) ;
- La gestion du portefeuille d'affaires ;
- La gestion de la chaîne logistique.

Dans cette thèse, nous avons appliqué des systèmes de matchmaking sémantique au contexte de la gouvernance de système d'information, pour l'alignement stratégique des objectifs métiers de l'entreprise avec les objectifs de référence, puis avec les objectifs et les investissements IT.

V.3.1. Agents Médiateurs

Dans les systèmes de matchmaking, nous parlons souvent d'agent intermédiaire comme l'une des implémentations les plus fréquentes ; il s'agit de l'utilisation des systèmes multi-agents dans le processus de matchmaking. Nous entendons par agent intermédiaire, une entité intelligente responsable de l'interconnexion de la demande avec l'offre. Nous définissons de nombreux types d'agents intermédiaires en fonction de leurs comportements (Fimbel, 2001) selon le tableau ci-dessous:

Agent Médiateur	Description de la médiation
Matchmaker	Service des Page Jaunes permettant d'échanger les adresses des agents demandeurs et fournisseurs
Broker	Échange des transactions du service demandé, communication et négociation entre demandeur et fournisseur qui passent obligatoirement par le broker.
Facilitator	Collecte des requêtes et leur répartition, conseil des demandeurs par rapport aux fournisseurs.

Tableau 5.2 : Comparaison des agents médiateurs.

Nous proposons un système de matchmaking autour d'un agent matchmaker, en mesure de publier des services de demande et d'offre et les faire correspondre sémantiquement. Le choix du Matchmaker est convenable au contexte de gouvernance SI où la négociation des services se fait d'une manière synchrone entre les acteurs concernés, mais une identification préalable est indispensable.

V.4. Algorithmes de matchmaking sémantique

Il existe plusieurs algorithmes de matchmaking dans chaque catégorie de médiation (contexte et de schéma). Certains d'entre eux sont basés sur la composition des services par les entrées et les sorties annotations (Coraux, 2007), d'autres se concentrent sur la description des capacités (entrées / sorties / effets) avec fonction de comparaison (Themistocleous, Irani, O'Keefe, & Paul, 2001). D'autres algorithmes sont modélisés selon les résultats de catégorisation (Perrey & Lycett, 2003), les plus utilisés parmi ces algorithmes sont :

1. **Approche de Greedy** : la correspondance est basée sur des termes de l'entrée et de la sortie. Chaque concept de l'entrée ou la requête correspond à l'un des concepts de la sortie

ou la réponse. Le traitement commence par tous les concepts de sortie de la requête (appeler la liste des candidats) et supprime un concept de la liste d'entrée dès qu'il est lié à un concept de réponse (Payne, Paolucci, Kawmura, & Sycara, 2002).

2. **Graphe bipartie** : la procédure de recherche prend une requête en entrée et fait la correspondance de ses concepts de sortie avec les concepts d'entrée de chaque offre. S'il existe une correspondance entre les concepts d'entrée et de sortie, les résultats sont publiés. Pour faire correspondre les entrées, nous invoquons l'algorithme hongrois (Kuhn, 1955) sur un graphique créé pour calculer une adaptation optimale du graphique. Le degré de correspondance est défini par le poids maximum du sommet lors de la correspondance. À la fin, une liste des annonces classées sur la base des concepts d'entrée et sortie est renvoyée.
3. **Matchmaking via des ontologies hétérogènes** : L'algorithme fournit un moyen de matchmaking sémantique via les descriptions avec des ontologies hétérogènes. L'algorithme utilise une fonction de similitude des concepts pour la correspondance qui a comme argument la somme pondérée de l'ensemble de synonymes, le voisinage sémantique et les caractéristiques de distinction (Jiajin Le ruiqiang & Dehua, 2005).
4. **La récupération de l'instance classée** : cet algorithme prend en compte la préférence des concepts fournis par l'utilisateur (Matthias & Burkhard, 2006). Il utilise le concept d'un arbre de classement pour faire correspondre et comparer les diverses réponses d'une requête particulière.
5. **QoS** : l'algorithme prend en considération la **Qualité du Service** que va offrir la réponse à la requête QoS. Cette valeur est variable selon le contexte et semble décisive en ce qui concerne l'adéquation de la demande, le retour de l'utilisateur est sollicité (Le-Hung Vu, Hauswirth, & Aberer, 2006).

Chaque algorithme présente des limitations selon son contexte d'utilisation, à titre d'exemple :

- Grand nombre d'itérations au cours du processus de matchmaking (approche de Greedy) ;
- Dépendance des résultats de la requête avec l'ordre des termes (Quality of Service) ;
- Absence de résultats du classement (l'algorithme hongrois) ;
- Absence des préférences de l'utilisateur (Matchmaking via des ontologies hétérogènes).

Dans notre cas, nous combinons le graphe biparti et la récupération de l'instance classée, pour proposer un algorithme de matchmaking sémantique adéquat à la médiation des objectifs-métiers SI et des objectifs métiers du référentiel.

Cependant, un système de médiation sémantique avec des entrées variables tel que le nôtre, est un problème d'IA pur que nous résolvons classiquement avec la notion de Système Expert à laquelle on va imbriquer des améliorations de contexte.

V.5. System Expert

V.5.1. Terminologie

Plusieurs définitions ont été données à un système expert (SE), la plus courante est celle de Petter Denning en 1986 qui le définit comme un programme conçu pour simuler le comportement d'un humain qui est un spécialiste ou un expert dans un domaine très restreint".

Il permet d'assembler et de déployer des connaissances formelles et des décisions de l'expert (forme heuristiques), tout en suivant un raisonnement conduisant aux réponses des questions initiales.

Dans un domaine spécifique, le SE est capable d'ajouter une nouvelle information ou un nouveau concept à l'existant, d'une manière incrémentale, en séparant les connaissances des contrôles (moyens d'utiliser des connaissances).

Un SE "Dialoguant" (systèmes évolués) peut communiquer avec l'utilisateur dans un langage compréhensible par ce dernier (langue naturelle, "jargon" du domaine, etc.)

La notion de systèmes Expert est apparue en 1970 avec le système MYCIN (Shortliffe, 2012) dans le domaine médical contenant une base initiale de 200 règles. Aujourd'hui les SE sont des systèmes à base de connaissances capable de modéliser les connaissances, et le raisonnement pour atteindre des résultats comparables à ceux donnés par un expert dans un contexte déterminé. Les systèmes experts sont fondés sur des stratégies heuristiques, pour résoudre des problèmes à l'opposé de raisonnement cognitif.

L'architecture générique d'un système expert (Figure 46) (Defarges & Bertucci, 2003) contient des modules de traitement interconnectés, à savoir:

- **Interface utilisateur:** facilite la communication avec l'utilisateur final ou les acteurs intermédiaires avec des formulaires ou des questions et réponses.
- **Base de connaissances:** contient des informations et des connaissances sur le contexte du problème, ainsi que les règles et les faits.
- **Moteur d'inférence:** utilise une stratégie de résolution et l'applique aux connaissances existantes dans le but de trouver de nouvelles informations.
- **Module d'explication:** permet au système expert de justifier son raisonnement.
- **Editeur:** édite les données et les connaissances dans la base.

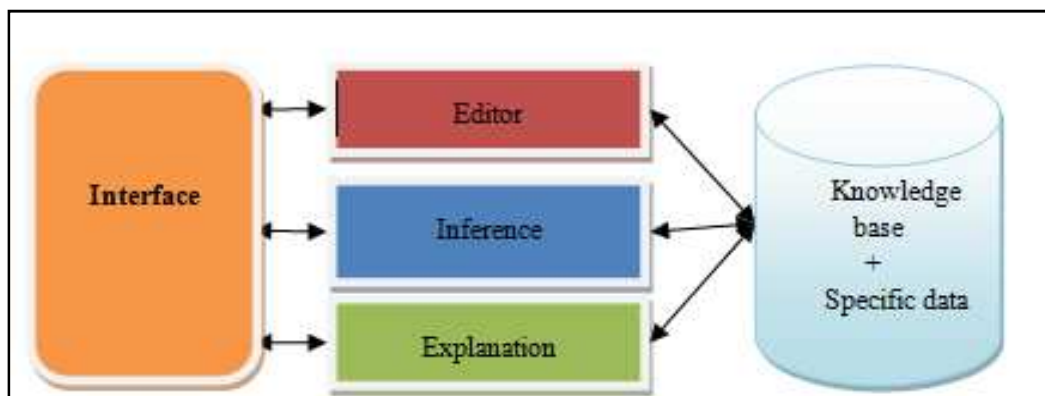


Figure 46: Architecture d'un système expert

Les principales architectures d'un SE sont basés sur :

1. La séparation du moteur d'inférence de la base de connaissances ;
2. L'utilisation du mode de représentation uniforme ;
3. La simplification du moteur d'inférence ;
4. L'exploitation de la redondance.

V.5.2. Types des Systèmes Expert

Il existe deux grandes familles de SE, les systèmes superficiels et les systèmes profonds (Defarges & Bertucci, 2003) :

1. Systèmes superficiels : Ils sont conçus pour s'exécuter rapidement (conclusions en un petit nombre d'étapes), en se basant sur des connaissances principalement empiriques ou compilées. Ils donnent de bons résultats dans leur domaine propre, et de mauvais résultats en dehors, exemple : MYCIN.
2. Systèmes profonds : Ils se basent sur des conclusions dérivant des principes premiers d'un modèle des phénomènes et du domaine, en donnant des solutions généralement longues à obtenir (résultats souvent non-triviaux aux observateurs) Exemple : SOPHIE de J. S. Brown & de J. Kleer (la physique qualitative).

V.5.3. Limite des Systèmes Expert

Les systèmes experts, solution clé de l'IA, malgré leurs succès et leurs bienfaits dans de nombreux domaines, présentent de nombreuses limites de spécialisation, de capacité d'apprentissage et de mise à jour. En effet, un SE est plutôt performant pour la résolution de problème précis dans un domaine de connaissance donné. Il ne l'est pas en cas de base de connaissances étendue et solutions nécessitant une prise de décision subjective. D'autres limitations de SE sont aussi à signaler, telles que :

- Le coût élevé de mise en place ;
- La difficulté de développement ;
- Le coût élevé des ressources humaines (ingénieurs de connaissances) matérielles et logicielles ;
- La mise à jour non automatisée avec absence d'apprentissage.

C'est dans cette perspective, que l'Intelligence Artificielle Distribuée (IAD) a vu le jour pour remédier à ces problèmes et étendre l'utilisation des SE.

V.5.4. Systèmes Expert et Systèmes multi-agents

Il est légitime de dire que les systèmes multi-agents ont vu le jour avec l'intelligence artificielle distribuée (IAD). En effet, l'IAD ne s'est intéressé au début qu'à la coopération entre plusieurs SE, pour atteindre un but commun. Pour y parvenir, nous divisons un problème quelconque en sous problèmes, et nous attribuons à chaque SE un sous-problème. L'ensemble des SE sont amenés à coopérer pour élaborer des solutions partielles qui vont finalement être synthétisées en une réponse globale au problème initial. Ainsi, l'IAD s'est intéressée au "problème à résoudre" par un ensemble d'entités intelligentes. Elle a été suivie par les agents et les systèmes multi-agents comme nous l'avons détaillé dans le chapitre 2.

Les agents d'un SMA sont (entre autres) autonomes, possiblement préexistants et généralement hétérogènes. On s'intéresse plutôt à « comment les agents vont-ils coordonner leurs connaissances, buts et plans pour agir et résoudre des problèmes ? ». Autrement ce sont les limitations des SE que nous sommes en train de dépasser. D'autant plus qu'avec la variété des

types d'agent et de modèle multi-agents, l'expertise est encapsulée dans un agent cognitif qui est soutenu par des agents réactifs et hybrides aidant ce dernier à réussir la résolution du problème.

V.6. Architecture de médiation proposée

La solution que nous proposons dans ce travail est un système expert de matchmaking de contexte, qui contient des agents cognitifs et un moteur d'inférence sémantique.

L'architecture de matchmaking proposée prend les avantages de système expert pour l'autonomie et l'expertise de résolution du problème. Elle déploie d'autant plus, les atouts de coopération des systèmes multi-agents et leurs protocoles de communication de haut niveau ; en particulier avec les autres agents du workflow inter-organisation de gouvernance SI, pour une meilleure orchestration.

Le système contient une entité de traitement sémantique pour comprendre les services via une ontologie du domaine de la tâche avant leur matchmaking. Deux algorithmes sont développés à ce niveau à savoir : algorithme de comparaison et algorithme de priorisation. Nous détaillons plus tard chacun d'entre eux.

L'architecture est basée sur la structure de référence d'un système expert à savoir: un explicateur, un moteur d'inférence et une base de connaissances, liée à un éditeur. Ces composants sont « agentifiés ».

Le matchmaking sémantique est assuré par le moteur d'inférence lié à l'explicateur qui effectue préalablement une analyse par rapport à l'ontologie et envoie les services indexées. Une comparaison des réponses avec les demandes est effectuée afin d'hierarchiser les meilleures offres.

L'ontologie est exprimée en Ontology Web Language (OWL), et analysée par l'entité de traitement sémantique qui envoie les services indexés au matchmaker pour effectuer la correspondance. Les résultats sont ensuite renvoyés à l'éditeur pour les enregistrer sur la base de connaissances selon les règles programmées.

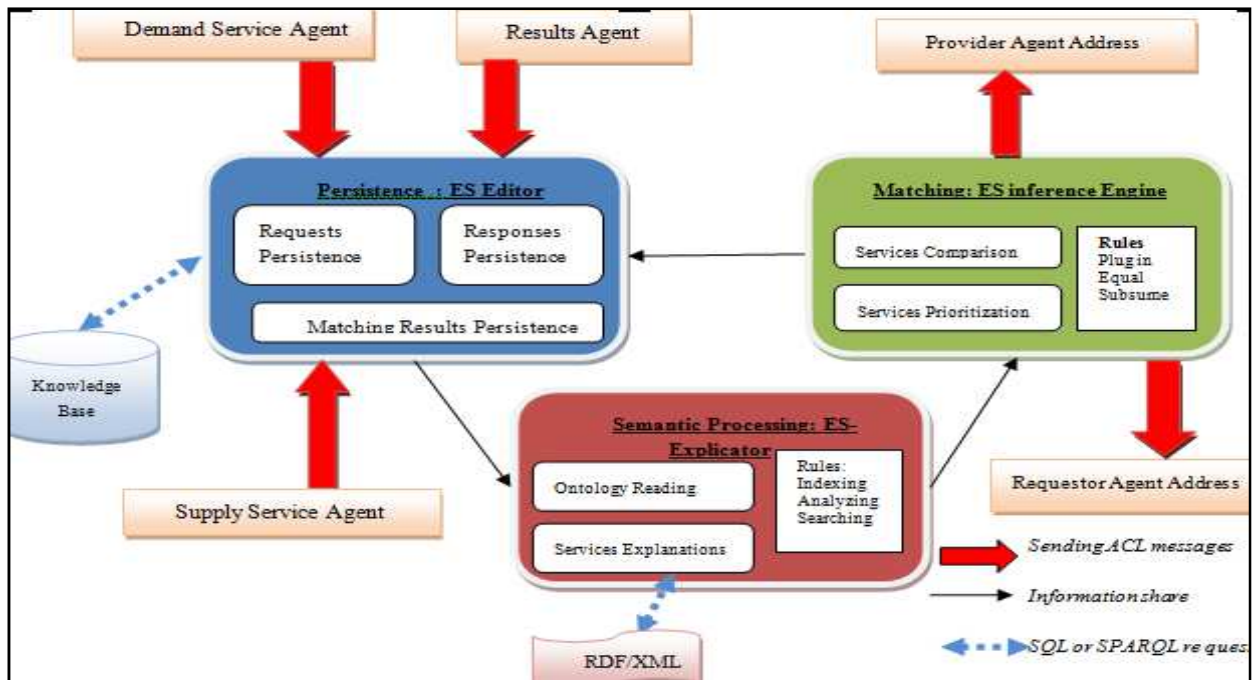


Figure 47: Architecture du système expert de médiation sémantique

V.6.1. Entité de persistance

Comme son nom l'indique, l'entité de persistance est responsable du stockage des demandes, des réponses et des résultats dans la base de connaissances (BC), elle est aussi responsable de l'actualisation de toutes les données et connaissances spécifiques autour du domaine, de l'environnement, des règles et des contraintes au niveau de la BC.

Sur le plan conceptuel, il s'agit d'un agent cognitif ayant comme tâche la persistance des services. Il est lancé une fois que une notification est envoyée de l'extérieur (interface utilisateur ou d'autres systèmes de notifications).

La structure du service à persister est différente d'un contexte à un autre. Ce service peut même être un service web car il n'y a pas de contrainte sur la structure interne du service dans les autres composants du système expert. Cependant, la description du service est nécessaire pour démarrer le traitement sémantique dans le premier niveau et le matchmaking sémantique dans le deuxième niveau.

V.6.2. Entité de traitement sémantique

C'est l'explicateur du système expert ; il contient essentiellement deux niveaux: lecture d'ontologie et explication des services. L'entité de traitement sémantique est liée au fichier RDF contenant l'ontologie : dans notre cas « ITGovernanceOntlogy ».

Dans le système expert de matchmaking, l'entité de traitement sémantique est un agent ayant la tâche de l'analyse de l'ontologie et la lecture des demandes et leurs réponses. Le traitement consiste à lire les offres et les demandes et les comparer à l'ontologie selon les préférences utilisateurs et ensuite donner une valeur de comparaison par attribut puis une valeur globale comme la somme des valeurs de comparaison des attributs.

Nous utilisons essentiellement la médiation de contexte avec des agents cognitifs communiquant avec ACL-requêtes pour envoyer les résultats entre eux. Pour l'algorithme de matchmaking nous avons combiné le graphe biparti et la récupération de l'instance classée.

Dans cette étape, le conflit des valeurs sémantiques est géré par l'ontologie de synonyme de manière à améliorer les performances du système (langue française dans notre cas)

O est l'ontologie avec des propriétés de données **Di** avec **i** et **j** qui sont des nombres entiers (**i** est compteur des propriétés de données et **j** compteur des requêtes)

Nous supposons que v est la fonction de comparaison donnant la distance sémantique de l'attribut A_{ij} d'une requête Q_i à propos d'un service donné, par rapport à la propriété des données **Di** de l'ontologie **O**.

$$V(Q_j) = \sum v(A_{ij}) = \sum A_{ij} \cdot D_i.$$

$V(Q_j)$ est la valeur de comparaison entre une requête indexée j : **Qj** et l'ontologie **O**. De la même manière $V(R_j)$ est la valeur de comparaison d'une réponse **Rj** par rapport à l'ontologie **O**.

V.6.3. Entité de médiation

C'est l'entité principale du système expert où les services compris par l'entité de traitement sémantique sont comparés et priorisés. En fait, l'agent matchmaker compare les demandes et les réponses en tenant compte de la valeur donnée par l'entité sémantique pour donner la meilleure réponse à la requête, en calculant M en tant que valeur de matchmaking:

$$M = \min (V(R_j) - V(Q_j))$$

Quatre cas sont possibles: l'égalité $M = 0$, Subsumption $-1 < M < 0$, Inclusion $0 < M < 1$ et

Échec $M > 1$.

Quant à la subsumption et l'inclusion un algorithme de priorisation est exécuté de manière à donner des réponses adéquates avec la description de la priorité.

V.7. Application au contexte de la Gouvernance IT

V.7.1. Introduction

Dans cette partie nous détaillons la façon dont l'approche est appliquée pour effectuer le matchmaking sémantique des services de gouvernance des systèmes d'information. Nous supposons que :

- Demandes: objectifs métier stratégique de l'entreprise
- Réponses: objectifs-métier de référence.

V.7.2. Modélisation de la requête

Nous proposons un modèle de requête spécifique à la gouvernance SI à consommer au niveau du système expert de médiation. La conception de la requête prend la forme d'un objectif COBIT et qui est comme suit :

Query = Objectif stratégique de l'entreprise

Objectif stratégique de l'entreprise = Perspective U action U cible U Détail.

En ce qui concerne les contraintes du système de matchmaking, un objectif stratégique d'entreprise peut avoir de nombreuses actions et de nombreux objectifs, mais une seule perspective.

Nous supposons que les détails de deux objectifs métiers sont approximativement égaux si près de 1/3 de mots sont égaux sans compter les mots de liaison.

Ainsi

$Q(x) = (P(x), A(x), T(x), D(x))$. **P** : perspective ; **A** : action ; **T** :target ; **D** : details.

Avec x qui est une requête de l'utilisateur à un moment donné à propos d'un composant du système d'information.

La requête $Q(P(x), A(x), T(x), D(x))$

La réponse comme un objectif métier $BO(P(x'), A(x'), T(x'), D(x'))$.

Nous définissons les quatres cas de comparaison comme suit:

1. Règle 1: Q est égal à BO si: $P(x) = P(x')$ et $A(x) = A(x')$ et $T(x) = T(x')$ et $D(x) \approx D(x')$.
2. Règle 2: Q est différent de BO si: $P(x) \# P(x')$ et $A(x) \# A(x')$
3. Règle 3: Q est incluse en BO si: $P(x) = P(x')$ et $A(x) \subset A(x')$ et $T(x) \subset T(x')$ et $D(x) \approx D(x')$.
4. Règle 4: Q est en subsomption avec BO si: $P(x) = P(x')$ et $A(x) \supset A(x')$ et $T(x) \supset T(x')$ et $D(x) \approx D(x')$.

Il est à noter que:

- Les opérateurs « U », « c » et « \supset » signifient l'union et l'inclusion de l'algèbre relationnelle.

V.7.3. Algorithme de Comparaison

```
Fonction Compare (R : chaîne, Q : chaîne) : entier
% 1 si R englobe Q
% 0 si Q englobe R
% R est un élément de l'attribut détail de l'offre R
% Q est un élément de l'attribut détail de la demande Q
% 0 représente l'ontologie (représentation en graphe)
% fonctions utilisées :
```

```

% Sup(E) : l'élément supérieur de R ou Q dans O
% Origine(O) : la racine de O
% Perspective(E) : perspective de R ou Q
% Cible (E) : cible de R ou Q
%Action (E) : action de R ou Q
% include( A,B) : inclusion syntaxique
Variables
SCourant : Sommet de O en cours d'examen
antécédentQ : EnsembledeSommets % antécédent à Q
antécédentR : EnsembledeSommets % antécédent à R

Début
antécédentQ <-- ∅
antécédentR <-- ∅

Si Q = racine(O) Alors
% Q n'a pas d'antécédent
antécédentQ <-- ∅
Sinon
Sommet <-- Sup(Q)
antécédentQ <-- Sup(Q)
Tant Que (Sommet <> Origine (O)) Faire
SCourant <-- Sup(SCourant)
% «+» désigne l'ajout d'un nouvel élément dans l'ensemble antécédentQ
antécédentQ <-- antécédentQ + SCourant
Fin Tant Que
Fin Si
Si ((Perspective(Q) = Perspective(R)) et (include (action(Q), action (R)) et include (
cible (R), cible (Q))
Alors
Compare<-- (R ∈ antécédentQ)
Sinon
Fin

```

V.7.4. Priorisation

Nous entendons par priorisation dans le contexte particulier des demandes et offres de la gouvernance des systèmes d'information, la déduction des processus IT adéquat à un objectif métier donné. En effet, un objectif métier stratégique de l'entreprise peut correspondre à un ou plusieurs objectifs métiers du référentiel et chaque objectif métier du référentiel correspond à plusieurs processus IT. Cependant il n'est pas convenable de déployer l'ensemble des processus évoqué, il faut les prioriser selon les préférences de l'utilisateur et selon la requête initiale.

Dans ce qui suit, nous proposons la fonction de priorisation suivante :

$$\text{Priorite}(x) = \text{Priorite}(\text{OM}(x)) + r(x)$$

Avec : x : processus IT donné

$\text{OM}(x)$: Objectif métier qui est appelé x dans la réponse à la requête initiale

$r(x)$: nombre d'apparitions de x dans la réponse à la requête initiale

NB : $\text{Priorite}(\text{OM}(x))$: est une préférence de l'utilisateur final.

V.7.5. Exemple

- 1) Pour tester l'architecture proposée, nous utilisons à titre d'exemple le but stratégique d'entreprise:
- 2) x = "obtenir des informations pertinentes et prendre des décisions stratégiques à propos des risques".

Perspective = "interne"

Actions = "Obtenir", "prendre"

Cibles = "informations pertinentes", "décision stratégique", "risque"

En appliquant les équations ci-dessus, nous trouvons les meilleures réponses suivantes

1. BG9: "obtenir des informations fiables et utiles"
2. BG2: «gérer le risque d'entreprise lié à l'IT».

Après hiérarchisation et traitement sémantique nous avons trouvé les résultats suivants:

<i>Processus IT</i>	<i>Priorité</i>
PO5 : Communiquer buts et l'orientation de gestion	5
DS7 : Former les utilisateurs	3
DS5 : Assurer la sécurité des systèmes	2
AI7 : Installer et accréditer les solutions et les changements	4

Tableau 5.3: Résultats du matchmaking.

Pour évaluer la solution, nous comparons les résultats du système expert de matchmaking avec une recherche sémantique simple pour le même exemple :

Requête	Système expert de matchmaking	Comparaison syntaxique
Obtenir des informations pertinentes et prendre des décisions stratégiques à propos des risques	PO5, DS7, DS5, AI7	PO2, PO9

Tableau 5.4: Comparaison des résultats du matchmaking et recherche sémantique

Nous pouvons en déduire que la comparaison syntaxique simple ne suffit pas pour les objectifs métiers de la gouvernance SI. Beaucoup de considérations sont nécessaires, c'est la raison pour laquelle la solution présentée définit des attributs spécifiques de la requête ainsi que des règles et des contraintes.

V.8. Conclusion

Dans ce chapitre, nous avons proposé une architecture de matchmaking composée d'un système expert de gouvernance des SI doté d'un moteur d'inférence sémantique et des systèmes multi-agent. Dans la première partie, nous avons présenté brièvement l'état de l'art des systèmes de matchmaking et des systèmes experts. Ensuite, nous avons proposé l'architecture du Système expert de matchmaking sémantique dans le contexte particulier de la gouvernance des systèmes d'information, en détaillant ses entités et les relations entre elles.

Dans la dernière partie, nous avons présenté l'algorithme de comparaison proposé et donné un exemple de mise en œuvre pour une requête ITG donnée.

Partie III:
Conception et Réalisation

CHAPITRE 6:
CONCEPTION ET REALISATION
DE LA PLATEFORME

VI. Conception et Réalisation de la plateforme.

VI.1. Introduction

Dans ce chapitre, nous présentons dans la première partie, le paradigme de modélisation orienté agent face à l'orienté objet ainsi que les particularités de chacun. Dans la deuxième partie, nous présentons les phases de conception et de réalisation de la plateforme EAS IT-Stratégie, à savoir la description des cas d'utilisation et des scénarii, l'analyse, la conception et la réalisation. Nous consacrons une dernière partie aux tests de la plateforme ainsi que son évaluation qualitative et quantitative.

VI.2. Contexte

EAS IT-Stratégie est une solution distribuée basée sur un workflow inter-organisationnel de gouvernance SI et des systèmes multi-agents. Sa conception doit donc être faite avec le paradigme orienté-agent. Cependant, l'environnement de ces agents contient un ensemble d'objets mis à leur disposition comme ressources. Donc le fait que l'environnement soit hétérogène fait d'objets et d'agents nécessite une modélisation objet en plus de la modélisation des agents.

Avant de détailler les deux approches de modélisation utilisées, nous donnons un aperçu sur ce qu'est un langage de modélisation pour un système informatique :

Un langage de modélisation (Ferber & Gutknecht, 1998) doit définir :

- Une sémantique des concepts ;
- Une notation pour la représentation de concepts ;
- Des règles de construction et d'utilisation des concepts
- Des langages à différents niveaux de formalisation tels que Z, B, VDM ayant une représentation mathématiques formelle des spécifications, ou semi-formels tels que MERISE et UML ayant une représentation graphiques, plus proche à la pratique de mise en œuvre.

L'industrie du logiciel dispose de nombreux langages de modélisation selon le système à mettre en place :

- Adaptés aux systèmes procéduraux : (MERISE...) ;
- Adaptés aux systèmes temps réel (ROOM, SADT...) ;
- Adaptés aux systèmes à objets (OMT, Booch, UML...).

Les mieux adaptées à notre contexte sont : UML pour la modélisation Objet et AUML pour la modalisation Agent :

UML (Unified Modeling language) est le langage de modélisation le plus répandu pour la conception orienté objet, il permet éventuellement de structurer les phases essentielles de développement du produit logiciel à savoir l'analyse des besoins, la conception, l'implémentation, les testes unitaires et les tests d'intégration. Une extension orienté agent de ce langage a été mise en place nommée AUML pour la modélisation graphique des systèmes multi-agents. AUML

hérite des représentations proposées par UML et comporte dix types de diagrammes symbolisant les différentes vues pour représenter des concepts particuliers d'un système d'information.

Nous optons pour AUML afin de modéliser cette plateforme vu son adéquation au contexte et sa richesse en ingénierie orientée agent.

VI.3. Modélisation Orienté Objet

UML (Unified Modeling Language) est un langage de modélisation unifié ayant pour objectif:

- ❖ Modéliser un système informatique de bout en bout (des concepts vers l'exécutable), en utilisant le paradigme orienté objet ;
- ❖ Simplifier et standardiser la modélisation;
- ❖ Mettre la modélisation à la portée de l'homme comme à celle de la machine;
- ❖ Représenter le système graphiquement avec des qualités formelles suffisantes pour le traduire automatiquement en code source.

En ce qui concerne les références de la modélisation UML :

- UML 1.0 : Booch, OOSE et OMT (1994).
- UML 2.0, guide de référence James Rumbaugh, Ivar Jacobson, Grady Booch(2005).
- UML 2.0 Benoît Charoux, Aomar Osmani, Yann Thierry-Mieg (2008).
- UML 2.0 Superstructure et UML 2.0 Infrastructure par OMG (Object Management Group) en (2004).

UML propose plusieurs diagrammes qui permettent de décrire les phases du projet de développement :

Diagrammes structurels ou diagrammes statiques :

- Diagramme de classes (Class diagram),
- Diagramme d'objets (Object diagram),
- Diagramme de paquetages (Package diagram),
- Diagramme de composants (Component diagram) ,
- Diagramme de déploiement (Deployment diagram).

Diagrammes comportementaux ou Diagrammes dynamiques :

- Diagramme de séquence (Sequence diagram),
- Diagramme d'activités (Activity diagram),
- Diagramme de cas d'utilisation (Use case diagram),
- Diagramme de collaboration (Collaboration diagram),
- Diagramme d'états (Statechart diagram).

VI.4. Modélisation Orientée Agent

AUML « Agent UML » est un projet de Bauer, Odell et al en 1999 (Graja, 2009), ses premiers travaux de spécification ont été menés au début de 2003 (Albert, 2005). Ce projet avait pour objectif de proposer des notations mieux adaptées au paradigme multi-agent, en étendant le langage UML.

AUML est développé par l'équipe FIPA appelée FIPA Modeling Technical Committee (Modeling TC) (Albert, 2005) qui a :

- Élaboré un état de l'art de la modélisation orientée agent,
- Recommandé une technologie pour l'adoption d'une sémantique, un méta-modèle et une syntaxe abstraite communes pour l'analyse et la conception des méthodologies basées sur les agents
- Été en adéquation avec les spécifications existantes de FIPA (Foundation for Intelligent Physical Agents) et OMG (Object Management Group)
- Présenté un aperçu du déroulement des protocoles d'interaction du modèle FIPA-ACL.
- Représenté le comportement interne d'un agent.

AUML adopte les approches suivantes :

- Niveau1 qui représente le protocole global (diagramme de séquence, package, classe).
- Niveau2 qui représente les interactions entre les agents (diagramme de séquence, activités, collaboration, état).
- Niveau3 qui représente l'architecture interne des agents (diagramme d'activité, et d'état).

Vu les caractéristiques diverses et les contextes variés des systèmes multi-agents, plusieurs améliorations ont été soulignées par rapport aux diagrammes UML.

- **Diagramme de classe :**

En effet, les diagrammes de classes servent à représenter l'aspect statique des entités de la plateforme. Ainsi, le diagramme de classes dans AUML a pour but de représenter un système entier d'agents, et non simplement des agents individuels. En effet, une classe agent est une classe UML qui peut aussi spécifier à la fois, l'identificateur de l'agent, les rôles, les actions, les capacités, les services, les protocoles et les contraintes.

Il est à noter que dépendamment de l'agent à modéliser, nous spécifions la totalité ou une partie de ses caractéristiques, la forme de base est la classe objet UML à laquelle nous imbriquons les autres caractéristiques.

- **Diagramme de séquence :**

Il a été modifié en diagramme de protocoles et correspond à la représentation des protocoles d'interaction utilisés par les agents. Il met l'accent sur la communication entre agents.

- **Diagramme de collaboration**

Il regroupe à la fois le diagramme de classes et le diagramme de protocole pour présenter la structure du système et son comportement.

- **Diagramme d'activité**

Il traite toutes les activités du système et décrit les échanges entre ces activités.

- **Les « Statecharts »**

Il ressemble au diagramme d'activités en gérant les états et le comportement du système.

- **Diagramme des cas d'utilisation**

Il est utilisé pour définir des cas d'utilisation du système et donner une analyse de son comportement fonctionnel.

- **Diagramme d'objet**

Il représente des objets instance de classe à un instant t de l'exécution du système afin d'identifier les valeurs de chaque attribut.

- **Diagramme de composants**

Il décrit les composants nécessaires pour l'exécution du système. Un composant est un élément du système qui via une interface fournit des services.

- **Diagramme de déploiement**

Il organise les composants physiques du système lors de leurs déploiements.

VI.5. Spécifications fonctionnelles et techniques

Le Workflow de Gouvernance SI est une plateforme multiutilisateurs qui permet pour un système d'information donné de/d' :

- S'authentifier.
- Effectuer une configuration statique du système.
- Evaluer un objectif stratégique lié au SI de l'entreprise par rapport à un référentiel IT GRC.
- Concrétiser les processus IT, les activités clé, les métriques et les contrôles à déployer et répartir les responsabilités.
- Mesurer le niveau de maturité de la demande stratégique.
- Mettre à jour le référentiel de gouvernance SI de la plateforme.
- Envoyer les résultats stratégiques à la plateforme EAS-IT-GRC.

VI.5.1. Description des cas d'utilisation

Deux niveaux de description des cas d'utilisation sont présentés à savoir :

- Cas d'utilisation des utilisateurs de la plateforme.

- Cas d'utilisation des agents de la plateforme.

Les utilisateurs de la plateforme sont :

- Responsables métier et/ou l'utilisateur d'un actif du SI.
- Responsable Système d'Information.
- Super utilisateur.

Les fonctionnalités de ces utilisateurs correspondent aux actions suivantes :

- S'authentifier.
- Configurer statiquement l'environnement.
- Formuler un objectif métier de l'entreprise lié au SI pour un actif donné.
- Récupérer le rapport d'activité à propos de l'objectif métier demandé.
- Lancer le traitement spécialisé de la demande via EAS-IT GRC.
- Lancer la mise à jour du référentiel.

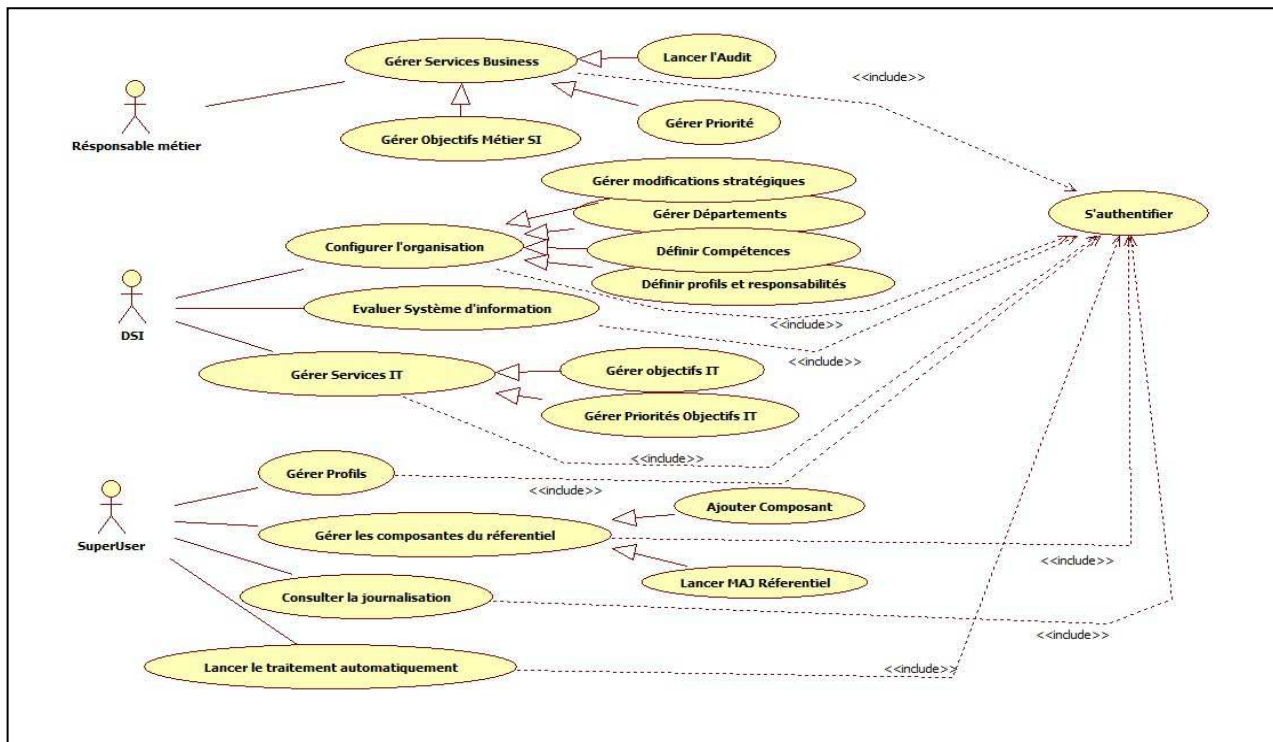


Figure 48: Diagramme cas d'utilisation utilisateurs

Les agents et système multi-agents de la plateforme sont :

- Agent workflow SI.
- Agent serveur de connexion SI.
- Agent serveur de connexion référentiel.
- Agent objectif métier.
- SMA Matchmaking.
- SMA SI Manager.

- SMA Framework Manager.

Les fonctionnalités des agents et SMA correspondent aux actions suivant :

- Encapsuler en service la demande SI à propos d'un actif.
- Gérer les agents workflow SI.
- Publier l'annuaire des services demandes et des adresses des agents demandeurs.
- Encapsuler en service l'offre proposée par le référentiel.
- Gérer les agents objectifs métier.
- Gérer les mises à jour du référentiel.
- Publier l'annuaire des services offre.
- Correspondre les offres aux demandes.
- Editer les rapports d'activités par demande traitée.
- Journaliser les demandes traitées.
- Envoyer les demandes traitées à EAS IT GRC.

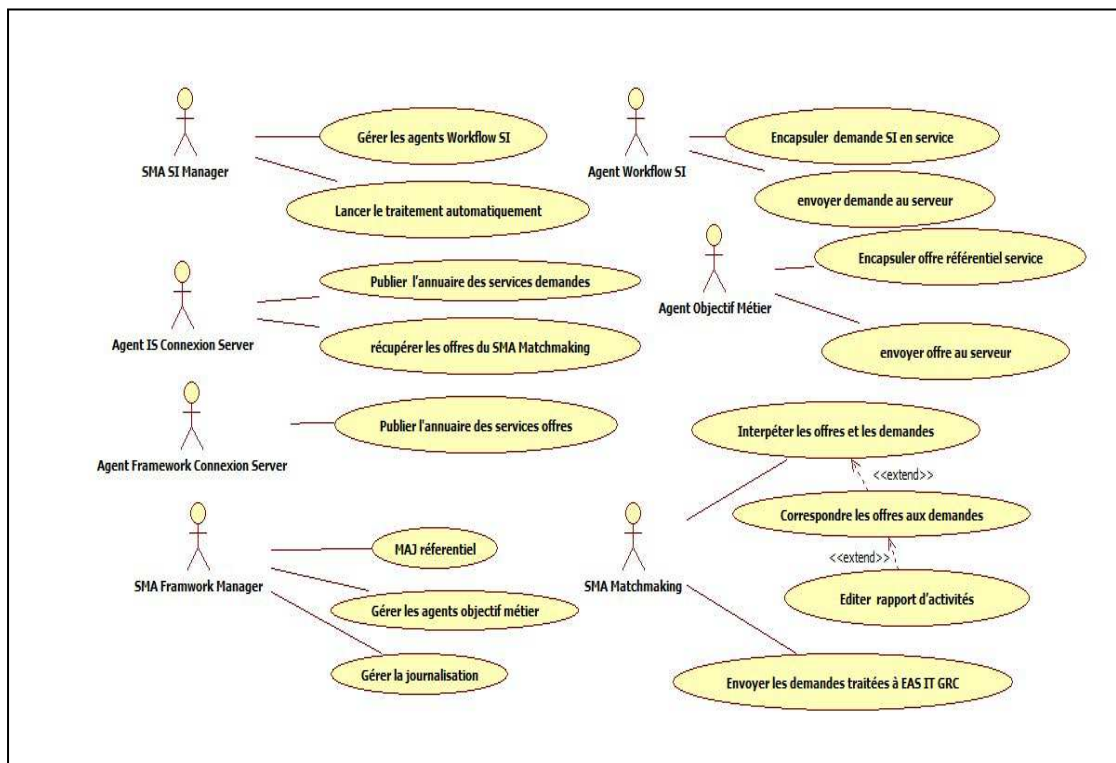


Figure 49: Diagramme cas d'utilisation Agents

VI.5.2. Description des Scénarii

Dans la même perspective de description des deux niveaux d'utilisation, en ce qui concerne les utilisateurs finaux de la plateforme :

- Le responsable SI s'authentifie et configure l'environnement par l'ensemble des informations à propos de l'entreprise à savoir ; la raison sociale, la capacité, le secteur d'activité, l'organigramme, les profils, les actifs, etc.

- Le responsable métier ou l'utilisateur d'un actif donné s'authentifie et crée une demande en formulant un objectif métier stratégique de l'entreprise à propos de l'actif en question.
- Les deux responsables peuvent récupérer le rapport d'activité autour de cette demande contenant :
 - Les objectifs métiers et informatiques de références adéquats à cette demande ;
 - Les processus IT et les activités clés à lancer ;
 - La matrice des responsabilités selon les profils de l'entreprise ;
 - Le niveau de maturité du SI dans ce propos ;
 - Les contrôles et métriques à déployer.
- Le responsable métier peut lancer le traitement spécialisé de sa demande via la plateforme globale EAS-IT GRC.
- Le responsable SI peut lancer la mise à jour du référentiel et afficher les journaux de la couche stratégique par date.
- Le responsable métier peut aussi lancer le traitement automatique à partir des anomalies évoquées par les utilisateurs potentiels et stockées dans une table des anomalies SI.

En ce qui concerne les agents et systèmes multi-agents de la plateforme :

- L'agent workflow SI récupère la demande d'un utilisateur, la transforme en service et l'envoie à l'agent serveur de connexion,
- Le SMA SI Manager gère le lancement et la suspension des agents Workflow SI.
- L'agent objectif métier présente un service offre et le publie au niveau de l'agent serveur de connexion référentiel.
- L'agent serveur de connexion référentiel publie l'annuaire des offres de la plateforme. Il est aussi actualisé par le SMA Framework Manager.
- Le SMA framework manager peut éventuellement ajouter, modifier ou supprimer une offre et gère la vie des agents objectifs métier. Il s'occupe aussi de la mise à jour du référentiel et de la journalisation.
- Le SMA Matchmaking interprète les demandes et les fait correspondre aux offres adéquates.
- SMA Matchmaking envoie les résultats à un SMA Strategie-COM de la couche communication de la plateforme globale EAS IT GRC.

VI.6. Analyse et Conception

VI.6.1. Diagrammes de séquences

Pour une vision globale du système, nous commençons par un diagramme de séquences fonctionnel représentant les échanges des utilisateurs finaux avec ce dernier pour auditer un objectif métier stratégique autour d'un ou plusieurs actifs du SI :

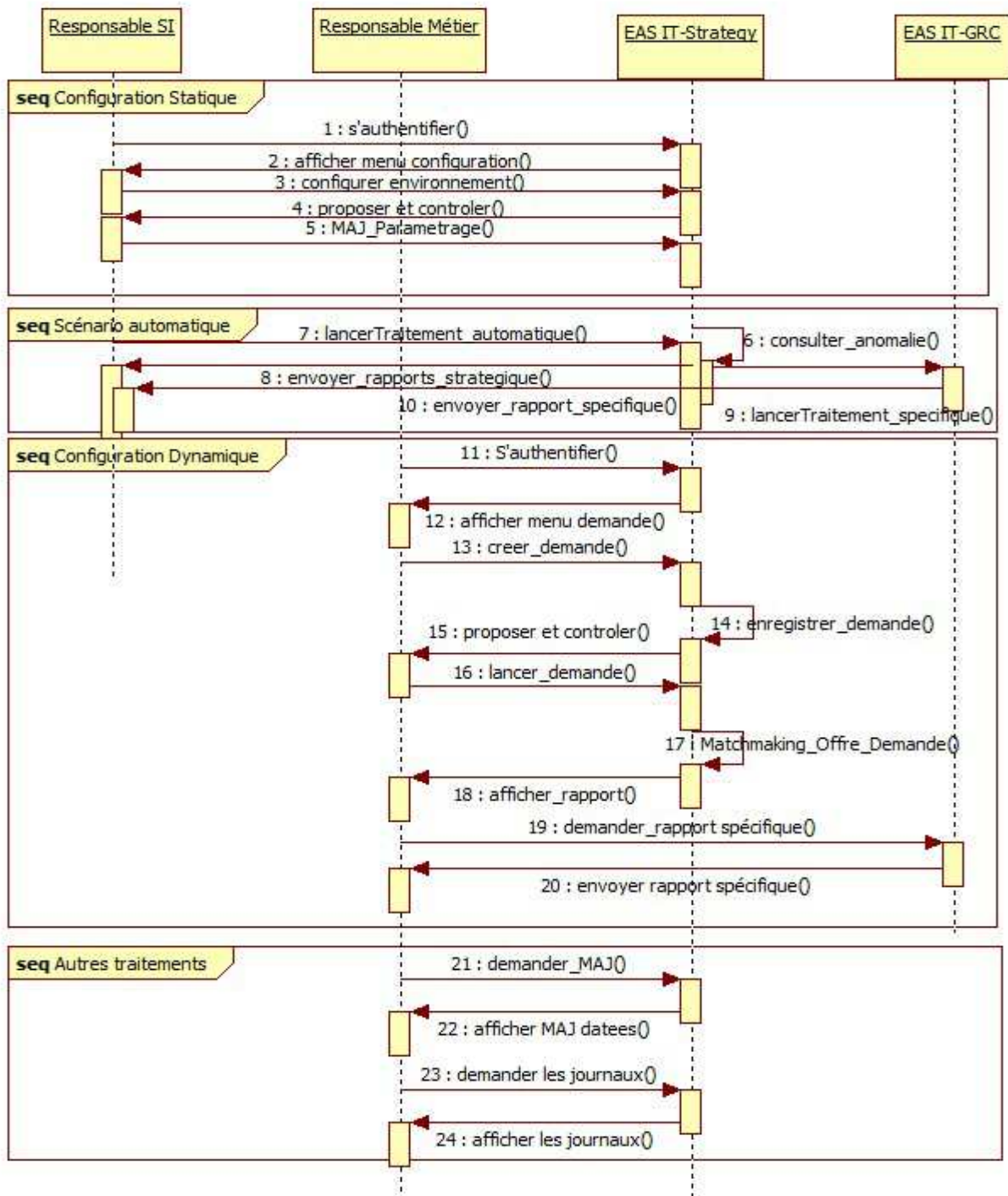


Figure 50: Diagramme de Séquence Fonctionnel

L'étape suivante est de présenter les diagrammes de séquences agents, Le premier diagramme figure 51 décrit l'interaction globale des agents et des SMA de la plateforme dans le temps à la réception d'un objectif métier stratégique de l'utilisateur :

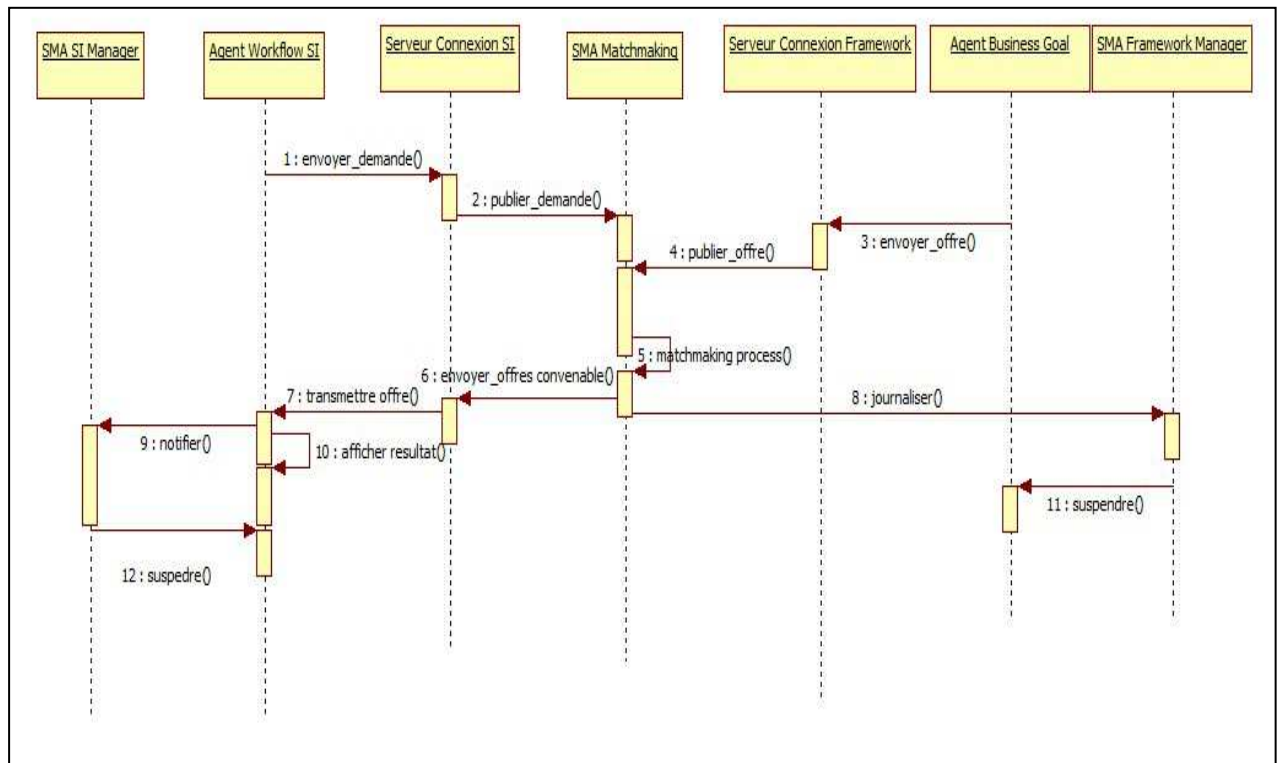


Figure 51: Diagramme de Séquence Globale Agents

Eventuellement pour chaque SMA, il y a un diagramme de séquences qui permet de décrire les échanges entre ses agents, avec les agents interfaces des autres couches, avec la base de connaissances et avec la plateforme globale EAS-IT GRC. Nous présentons donc, dans ce qui suit, trois diagrammes séquences à savoir:

- Diagramme de séquences SMA SI Manager ;
- Diagramme de séquences SMA Matchmaking ;
- Diagramme de séquences SMA Framework Manager ;

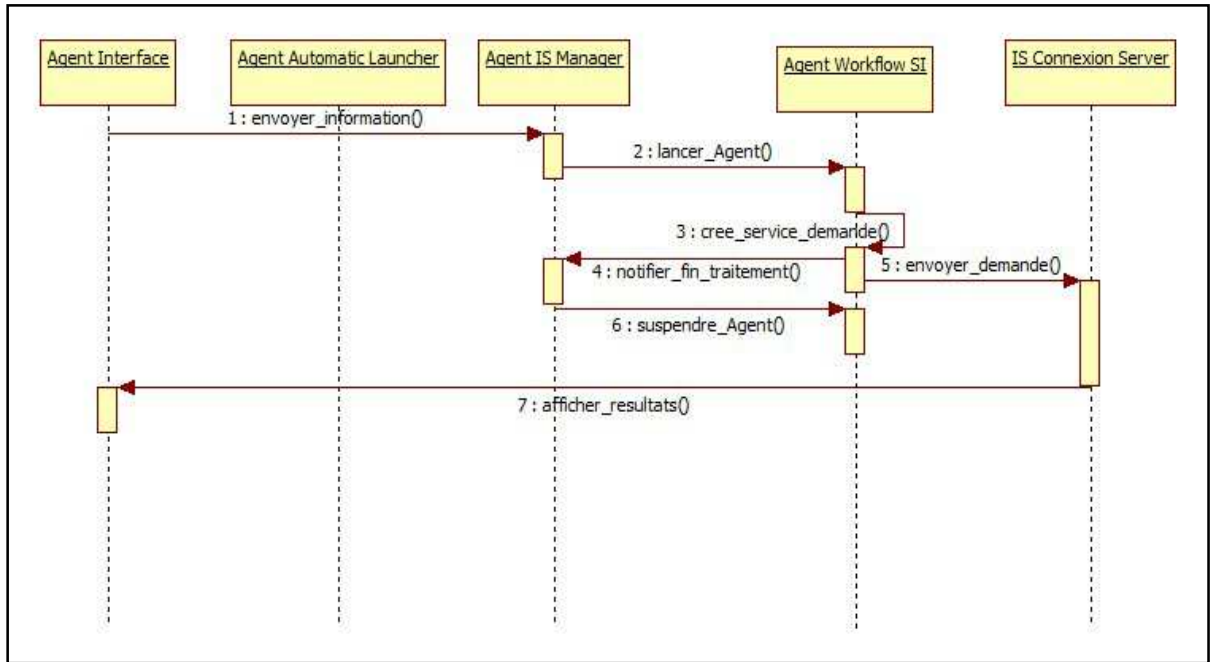


Figure 52: Diagramme de séquence SMA SI Manager et Agent Workflow SI

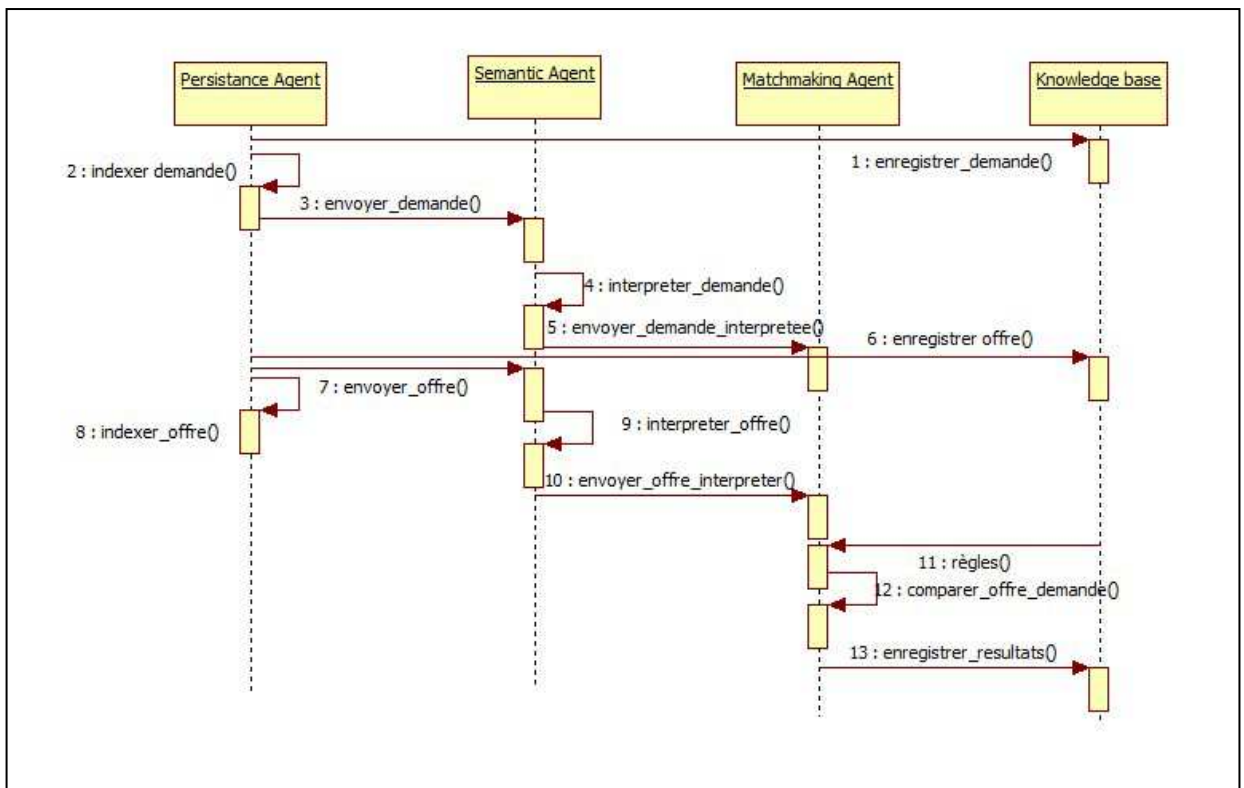


Figure 53: Diagramme de séquence SMA Matchmaking

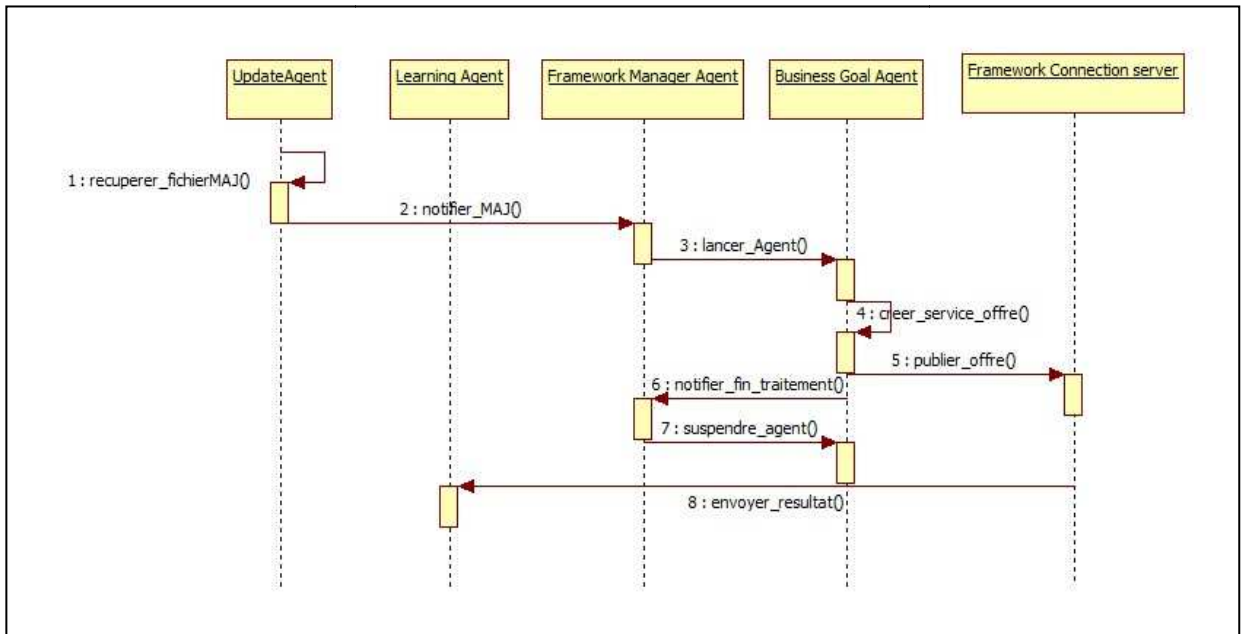


Figure 54: Diagramme de séquence Framework Manager

VI.6.2. Diagrammes de classes

Le diagramme de classes permet de définir le niveau conception et le niveau implémentation de la modélisation de la plateforme. En effet, dans notre cas, l'existence d'agents impose la représentation d'un diagramme de classe agent. Cependant, ces agents utilisent des objets comme ressources pour effectuer leurs tâches donc une coexistence des agents et des objets est proposée dans notre modélisation pour une meilleure compréhension.

En guise de simplification, nous présentons le diagramme de classes de chaque couche du Workflow de Gouvernance SI à part ; nous avons donc trois diagrammes :

- Diagramme de classe couche SI
- Diagramme de classe couche Médiation
- Diagramme de classe couche Framework

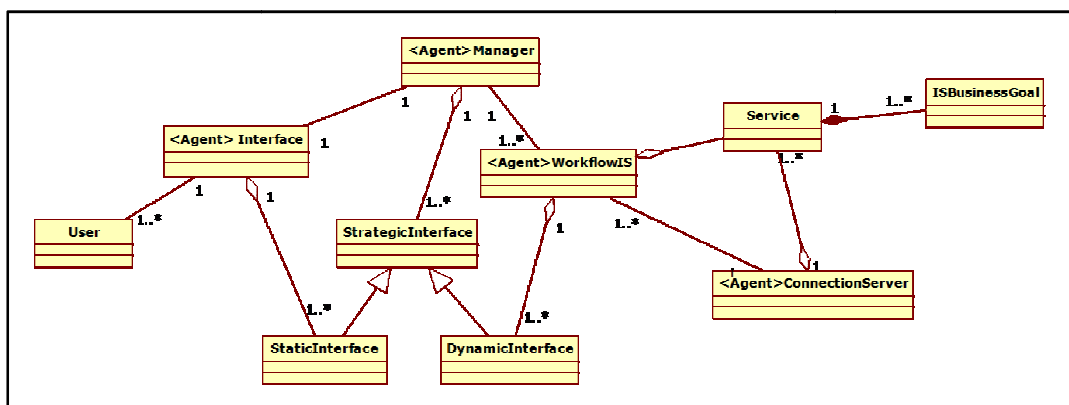


Figure 55: Diagramme de Classe Couche SI

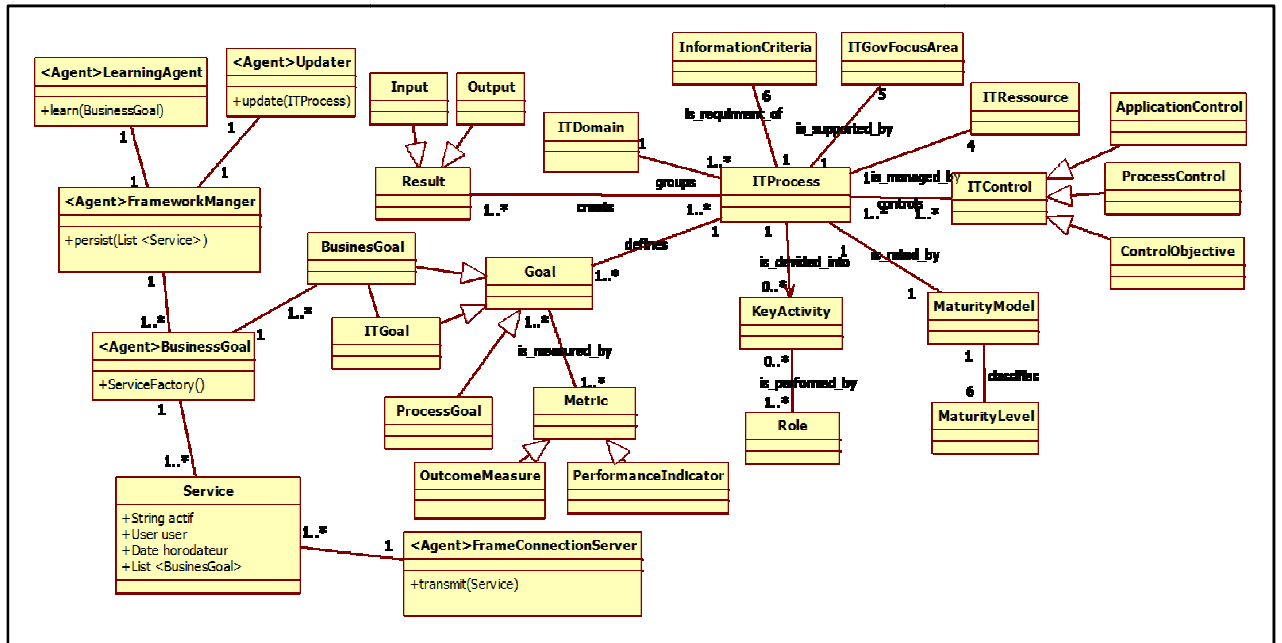


Figure 56: Diagramme de Classe Couche Framework

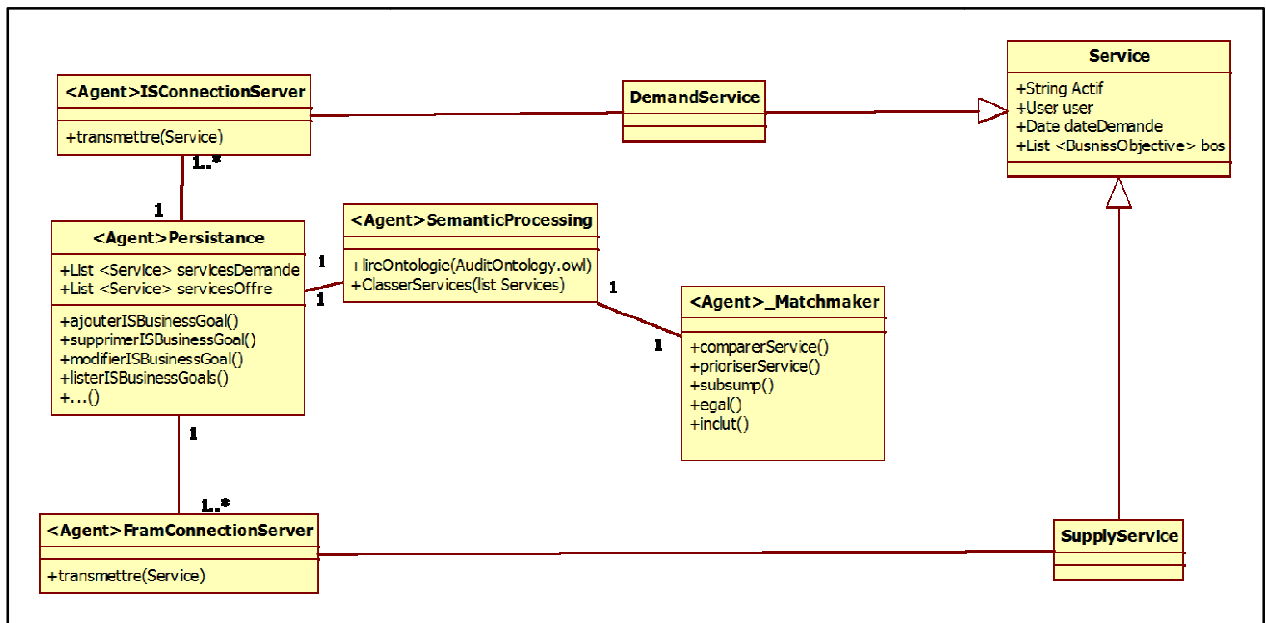


Figure 57: Diagramme de classe Couche Médiation

Au niveau implémentation, les aspects Workflow Inter-organisations et Systèmes Multi-agents ont impliqué la structure interne de la classe agent. En effet, pour chaque agent de la plateforme nous identifions six parties :

- Les attributs : les caractéristiques statiques de l'agent qui le décrivent dans le système global, exemples : identifiant, horodateur, intitulé, etc.
- Les groupes : les regroupements d'agents de la plateforme définis selon le modèle AGR (voir chapitre 3) à savoir : Groupe Audité, Groupe Auditeur, Groupe Recherche Auditeur, Groupe Recherche Audité et le Groupe Audit.
- Les rôles : les fonctionnalités des agents au sein des groupes selon le modèle AGR.
- Les services : l'ensemble des échanges des organisations du WIO de gouvernance SI, encapsulé dans un service qui change d'état et qui est compréhensible par les agents et les acteurs.
- Le protocole de communication : FIPA- ACL
- Les fonctions : la dimension dynamique de la classe.

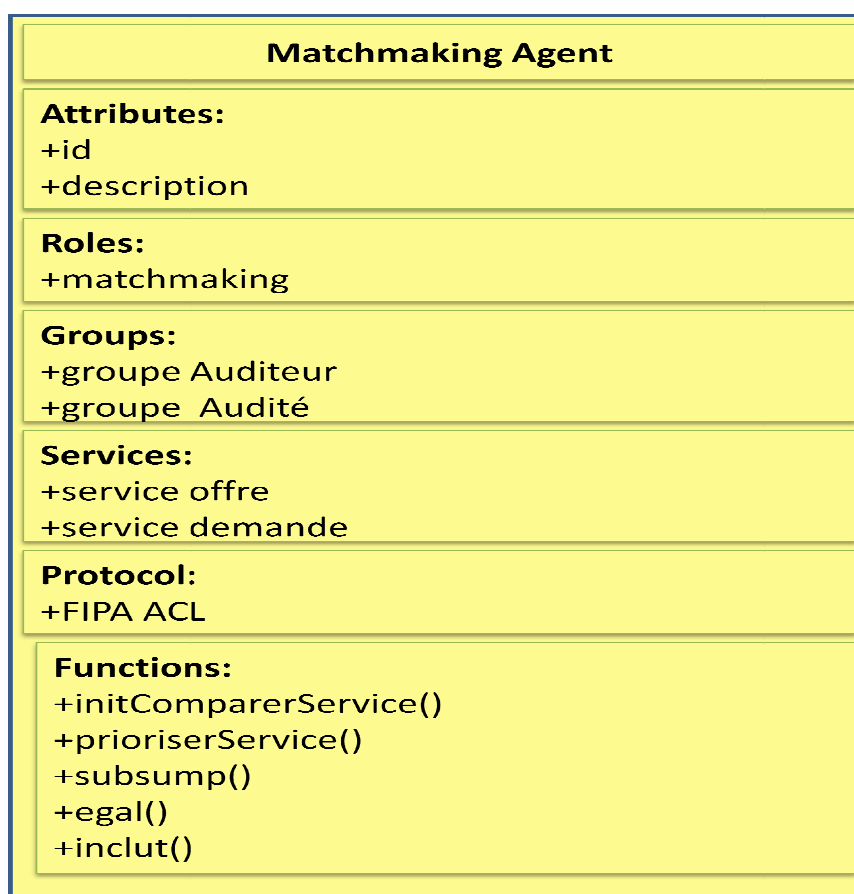


Figure 58: Exemple de classe Agent

VI.7. Réalisation

VI.7.1. Introduction

Pour valider expérimentalement l'architecture du Workflow Inter-organisationnel de type tâche de gouvernance des systèmes d'information, nous avons procédé à l'implémentation des différentes versions de la plateforme.

La première et la deuxième version étaient des simulateurs java client lourd, la troisième version est une application web qui en plus des fonctionnalités proposées communique avec la plateforme globale EAS-IT GRC.

Dans la première version, nous avons simplement validé la communication des différents agents du Workflow Inter-organisationnel avec des objectifs métiers prédéfinis de part et d'autres au niveau de la couche SI et la couche Framework et une médiation en « si sinon ».

Dans la deuxième version, nous avons ajouté à la première version, la base de connaissance où nous avons stocké les informations de bases et les différentes règles de médiation et nous avons implémenté une interface de lancement des agents pour simuler les échanges. Nous avons effectué des testes pour valider

Dans la troisième version nous avons implémenté la quasi-totalité des fonctionnalités proposées avec :

- les interfaces des utilisateurs potentiels ;
- l'ontologie « ITGovernanceOntology » de gouvernance des systèmes d'information que nous avons déployé dans la couche de médiation pour l'interprétation des offres et des demandes ;
- l'algorithme global de comparaison pour améliorer le système expert de médiation ;
- la mise à jour du référentiel avec la version 5 de Cobit ;
- la journalisation des demandes et l'apprentissage pour développer les offres proposées.

VI.7.2. Environnement de Développement

La plateforme dans sa version 3 est une solution web développée en Java /J2EE en utilisant les Frameworks JPA, EJB, JSF2.2 avec une base de données Mysql. L'IDE utilisé est Netbeans 7.4.

NetBeans : est un environnement de développement - un outil pour les programmeurs pour écrire, compiler, déboguer, déployer et tester des programmes. Il est écrit en Java - mais peut supporter n'importe quel langage de programmation. Il y a également un grand nombre de modules et plug-in pour étendre l'IDE NetBeans

Java (Albert, 2005) :

Le langage Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton employés de Sun Microsystems avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld. En 2007 sun est racheté par Oracle corporation.

Le langage Java assure la portabilité sur plusieurs systèmes d'exploitation tels qu'Unix, Linux, Microsoft Windows ou Mac OS avec des pré-requis minimaux (version JDK et JRE). Il permet de développer des applications client-serveur, mobiles, web. Côté client il supporte des applets, servlets et des portlets.

J2EE (Gosling, 2000) :

C'est une spécification pour la technique Java d'Oracle plus particulièrement destinée aux applications d'entreprise. Ses applications sont considérées dans une approche d'implémentation multi-niveaux pour la création d'application distribuée de l'entreprise. Elle contient des bibliothèques supplémentaires en plus de SE et des conteneurs lourds tels que JBOSS, Apache Tomcat, Glassfish, etc..

J2EE dispose des frameworks facilitant le travail du développeur en lui déléguant uniquement son développement spécifique en prenant en charge des tâches telles que :

- l'architecture de base de l'application ;
- l'accès aux données ;
- l'internationalisation ;
- la journalisation des événements ;
- la sécurité (authentification et gestion des rôles) ;
- le paramétrage de l'application.

Les frameworks J2EE utilisés dans cette solution sont :

JPA : est une interface de programmation Java permettant d'organiser des données relationnelles dans des applications. (Correspondance objet-relation)

EJB: est une architecture de composants logiciels côté serveur pour créer des composants distribués, de proposer des services avec ou sans conservation d'état entre les appels (EJB Session) ou encore d'accomplir des tâches de manière asynchrone (EJB Message).

JSF: c'est un framework java pour le développement de la couche présentation d'une application web. Il est basé sur la notion de composant qui est enregistré lors du rendu de la page, pour être ensuite restauré au retour de la requête.

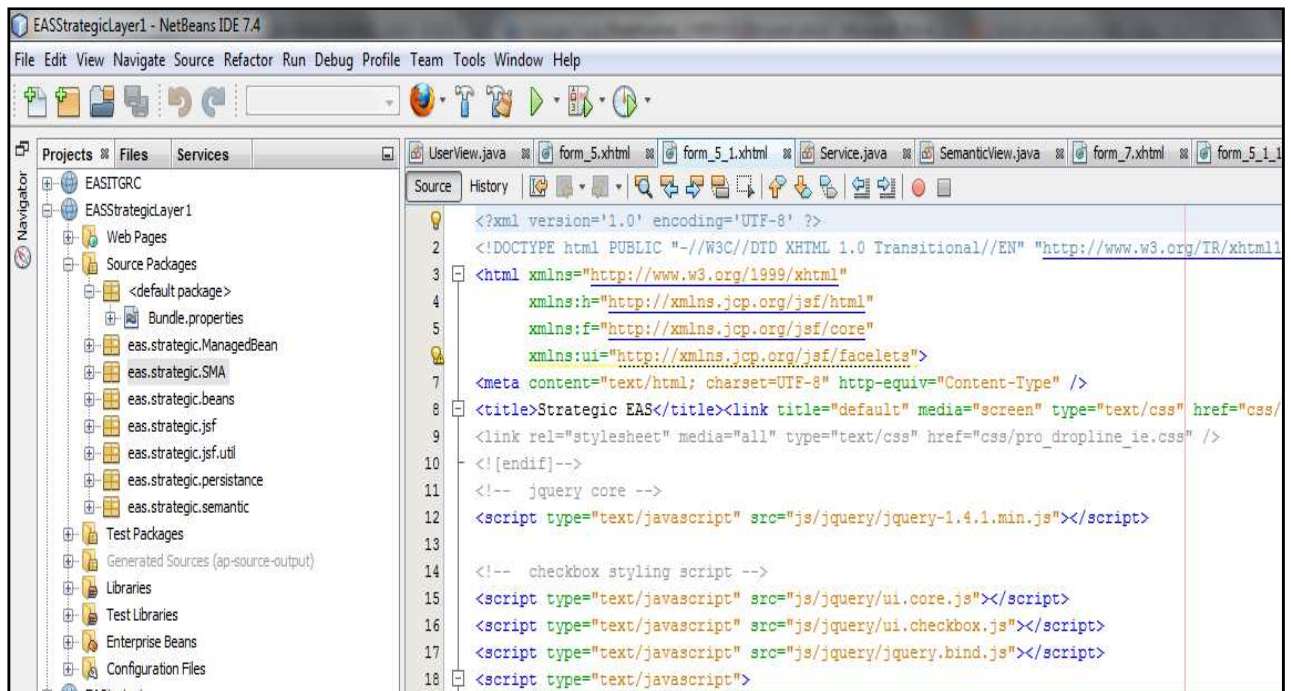


Figure 59: Structure du Projet dans l'IDE

La figure 59 montre la répartition du projet web avec le modèle de conception MVC et la couche des systèmes multi-agents sur l'IDE NetBeans.

Pour l'implémentation des agents nous avons utilisé l'API **Madkit 5**, ce sont les bibliothèques sources de la plateforme des systèmes multi-agents Madkit importés en bibliothèque externe au niveau de l'environnement Netbeans pour l'application.

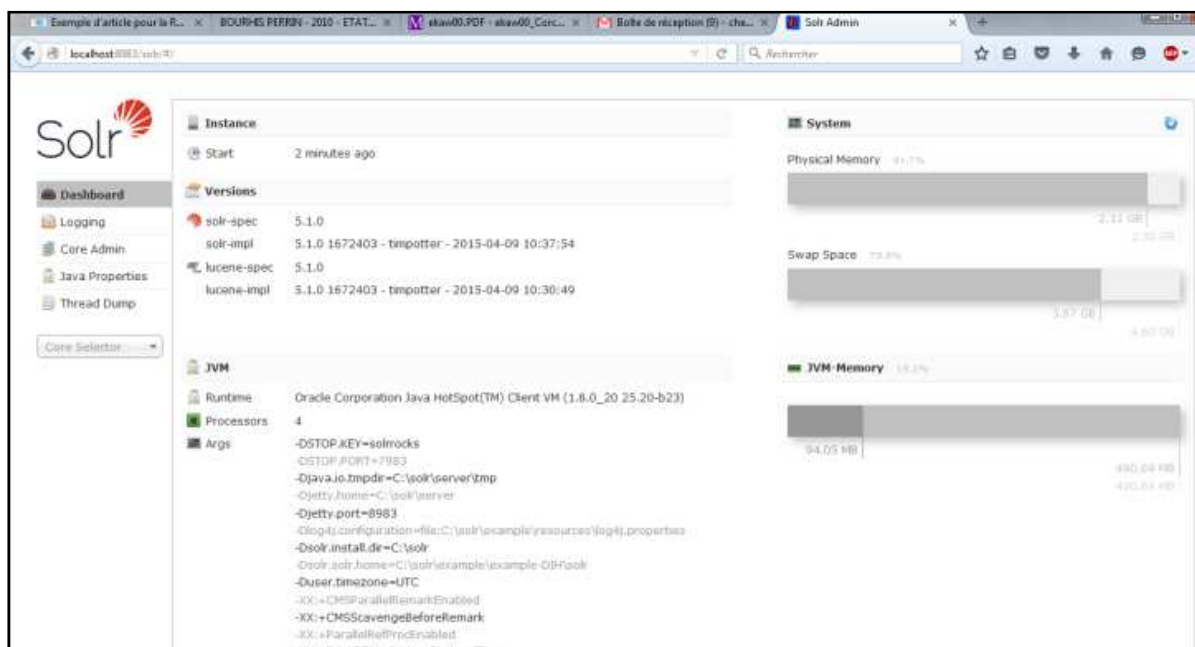


Figure 60: Environnement d'analyse sémantique

Pour l'analyse sémantique nous avons utilisé le serveur **Apache Solr version 5.1.1** (voir Figure 60) paramétré avec l'ontologie « IT Governance Ontology ». Sa valeur ajoutée en plus des testes de l'efficacité sémantique de cette dernière et de prendre en charge les synonymes, les règles syntaxiques et grammaticales de la langue française pour une amélioration de la performance. Nous avons également fait appel à la bibliothèque **Solr API** pour l'interfaçage avec les traitements de la plateforme.

Pour l'ontologie, et comme cité précédemment nous avons utilisé le langage **OWL-S** sous l'éditeur Protégé 4.3 avec le compilateur **Fact++** pour la validation.

La figure 61 montre l'environnement de développement de l'ontologie sous Protégé 3.4. Nous avons suivis plusieurs étapes pour sa mise en œuvre, telles que la création des entités et des classes, la définition des objets et des propriétés de données et l'instanciation des individus.

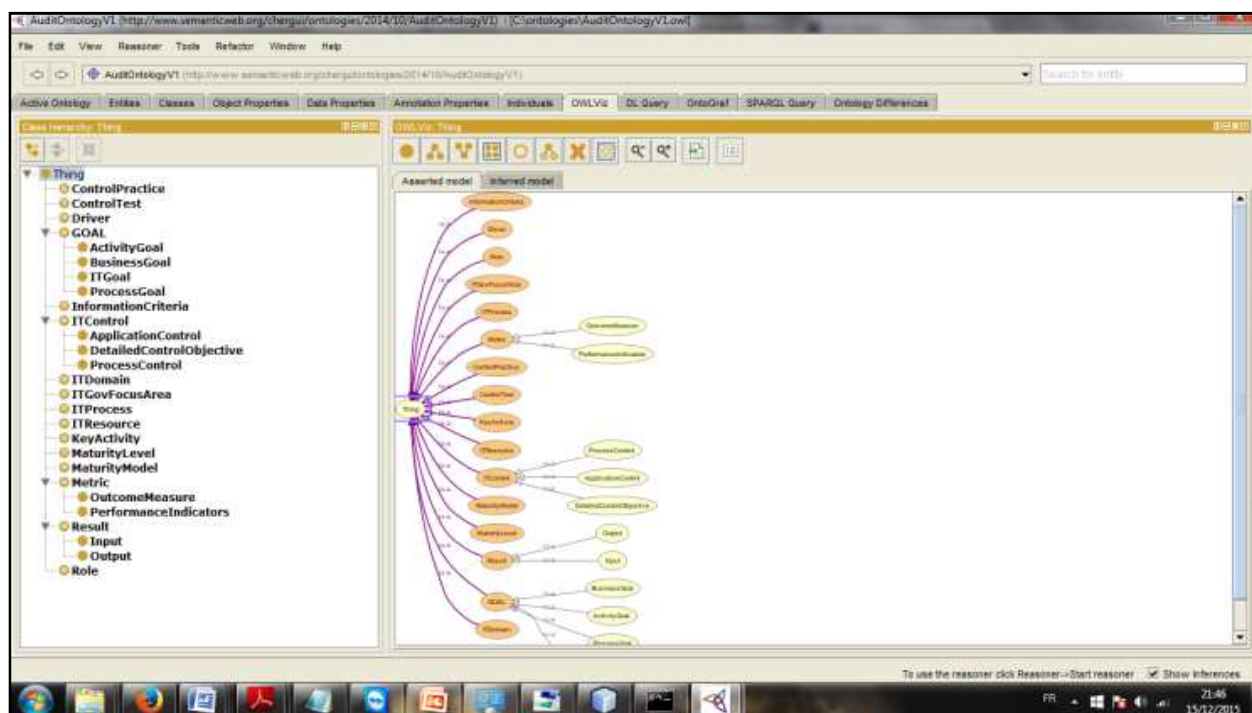


Figure 61: Environnement de développement de l'ontologie

VI.7.3. Phases de développement

Comme décrit précédemment, la dernière version de l'architecture proposée vise à assurer la bonne gouvernance d'un système d'information complexe. Il est basé sur trois éléments essentiels à savoir:

1. Workflow Inter-organisationnel de GSI ;
2. Système Expert de matchmaking avec un moteur d'inférence sémantique ;
3. Système multi-agents Framework GSI.

Pour mettre en œuvre cette architecture, nous avons proposé une solution Web multiutilisateurs liés à une base de connaissances ; des agents communicants déployés pour:

- Capture des besoins utilisateurs ;
- Interprétation des demandes en objectifs métiers de références ;

- Proposition pratique de processus informatiques à exécuter face aux demandes des utilisateurs ;
- Mise à jour du cadre de référence utilisé.

Pour évaluer les résultats de la plate-forme, nous les comparons à celles d'un expert de gouvernance IT, puisque l'un des principaux objectifs de ce travail est d'informatiser la mission d'audit d'un système d'information via des entités intelligentes remplaçant l'auditeur.

VI.7.4. Description des interfaces

Comme présenté au paravant, les fonctionnalités principales de la plateforme sont :

- Authentification ;
- Configuration statique de l'environnement ;
- Création dynamique de la requête stratégique autour d'un actif du SI ;
- Traitement intelligent de la requête et visualisation des résultats ;
- Edition de rapport d'audit ;
- Journalisation système ;
- Transfert des résultats à EAS IT-GRC.

Les écrans ci-dessous illustrent ses fonctionnalités :



Figure 62: Authentification

Plusieurs profils s'authentifient au niveau de la plateforme à savoir :

- Les responsables métiers de chaque département (et/ ou utilisateurs potentiel d'un actif du SI) ;
- Le responsable SI ;

- Le super utilisateur ;

The screenshot shows the 'Configuration Générale de l'organisation' interface. At the top, there is a navigation bar with 'Configuration Générale', 'Départements', 'Actifs', 'Organigramme', and 'Autres'. Below this, the main title is 'Configuration Générale de l'organisation'. A progress indicator shows five steps: 1. Informations Générale (active), 2. Départements, 3. Actifs, 4. Organigramme, and 5. Autres. The form contains the following fields:

- N° de Patente: RE56789
- Raison Social: ENSEM
- Capital: 10000
- Effectifs: 500
- Secteur d'activité: RECHERCHE & DEVELOPPEMENT
- Adresse 1: ROUTE ELJADIDA, KM 7, BP 8118, OIASIS, CASABLANCA MAROC
- Adresse 2: (empty)

At the bottom, there are two buttons: 'Envoyer' (green) and 'Annuler' (grey).

Figure 63: Gabarit de l'application et le Menu Principal

La Figure 63 donne un aperçu global sur le gabarit et le menu principal de l'application; la figure montre aussi la première étape de configuration de l'environnement à savoir la configuration générale de l'organisation qui s'étale sur 5 étapes, ci-dessus l'étape 1 « Identification générale ».

La figure 64 montre une autre étape de configuration statique, à savoir « l'organigramme », elle permet de définir les rôles et la hiérarchie de l'organisation.

The screenshot shows the 'Configuration Générale de l'organisation' interface at the fourth step, 'Organigramme'. The navigation bar at the top is the same as in Figure 63. The progress indicator shows five steps: 1. Informations Générale, 2. Départements, 3. Actifs, 4. Organigramme (active), and 5. Autres. The form contains the following fields:

- Nom: CHERGUI
- Prénom: MERIYEM
- Diplôme: INGENIEUR
- Certificat: ITIL
- Département: IT GRC (dropdown)
- Rôle: Conception et developpement (dropdown)

At the bottom, there are two buttons: 'Envoyer' (green) and 'Annuler' (grey), followed by a link '>>SUIVANT'.

Figure 64: Configuration statique de l'environnement

Il est à noter que ce paramétrage statique permet de situer la requête de l'utilisateur et de la traiter selon les spécificités de l'organisation en question.

Accueil Configuration de l'organisation **Stratégie** Mise à jour Mon compte

Bienvenue

Evaluation de la Stratégie

1 Définition Demande 2 **Ajout Objectifs métiers** 3 Confirmation des résultats

Intrusé Demande -> investissement

Action	Perspective	cible	Detail	Priorite
améliorer	Perspective Interne	▼ investissement	améliorer les investissements de l'entreprise	1 ▼

Envoyer

Figure 65: Création de la requête stratégique

La figure 65 concerne à la fois le responsable métier et le responsable SI et comment ils peuvent exprimer leurs demandes à propos d'une application dans le système d'information. Un ordre de priorité de cette demande est défini par l'utilisateur concerné (valeur de 1 à 5) Après le lancement du traitement, les résultats sont détaillés comme le montre la figure 66, à propos des objectifs métiers recommandés, des objectifs informatiques et des processus IT à mettre en place. Un rapport est généré avec d'autres détails tels que les contrôles et les métriques, les modèles de maturité, les activités clés et la matrice des responsabilités.

Comme le montre la figure 66 ci-après, l'application présente le détail de la demande et les offres correspondantes avant de permettre à l'utilisateur d'éditer le rapport, cette synthèse primaire lui permet de décider s'il est nécessaire de lancer le traitement de sa demande via EAS-IT GRC ou se contenter des directives d'ordre stratégique.

Evaluation de la Stratégie

1 Définition Demande 2 Ajout Objectifs métiers 3 **Confirmation résultats**

objectif métier: améliorer les investissements de l'entreprise Rechercher

Export Business Export ITGoals Export ITProcess

Objectifs métiers:

ID	ACTION	CIBLE	DETAIL	Perspective
1	Obtenir	retour sur investissement	Obtenir un bon retour sur investissement des investissements informatiques pour les métiers	Perspective financière

Objectifs IT :

ID	ITGDESCRIPTION
24	Améliorer la rentabilité de l'informatique et sa contribution à la profitabilité de l'entreprise

Processus IT :

Iditp	CODE	DESCRIPTION
5	PO5	Gérer les investissements informatiques
23	DS6	Identifier et imputer les coûts
15	AI5	Acqu岸 des ressources informatiques

Journalisation

Imprimer la page

[Vers IT GRC](#)

Figure 66: Résultats de la Demande

VI.7.5. Test et Intégration

Plusieurs tests fonctionnels ont été fait par la plateforme: une longue liste de processus IT est donnée pour chaque demande (10 processus IT prioritaires maximum). La priorité du processus IT dépend de la fonction de priorité du système expert de Matchmaking qui est dans notre cas linéaire $f(x) = x$. Elle dépend aussi de l'attribut de la requête que nous choisissons comme critères de recherche.

Dans le présent exemple, nous comparons les résultats de la plate-forme avec le point de vue d'un expert (Auditeur SI):

Requête: "obtenir des informations pertinentes et prendre des décisions stratégiques par rapport aux risques" priorité = 3.

Les résultats de la plate-forme sont présentés comme suit :

La demande correspond à plusieurs Objectifs métiers (OB). Chaque OB est détaillé sur un ou plusieurs objectifs informatiques et chaque objectif informatique est expliqué avec plusieurs processus IT.

Pour la demande exemple, nous obtenons:

Objectifs métiers	Priorité
Obtenir des informations fiables et utiles à la prise de décisions stratégiques	3
Fournir la conformité avec les lois externes, règlements et contrats	3
Fournir la conformité aux politiques internes	3
Gérer les risques métiers liés au SI	3

Tableau 6.1 : Offres Adéquates à la Requête

Objectifs informatiques	priorité
Assurer la transparence et la compréhension des coûts informatiques, les prestations, la stratégie, des politiques et des niveaux de services.12	3+2=5
Assurer une utilisation correcte et la performance des applications et des solutions technologiques 13	3+2=5
Veiller à ce que les services informatiques et les infrastructures puissent bien résister et à se remettre de l'échec dû à une erreur attaque délibérée ou d'une catastrophe 21	3+3=6
Veiller à ce que les transactions commerciales et les échanges automatisés d'information puissent faire confiance 22	3+3=6
IT Process	Priority
Communiquer objectifs et les orientations de gestion PO6	2+6=8
Éduquer et former les utilisateurs DS7	2+6=8
Assurer la sécurité du système DS5	3+9=12
Installez et accréditer les solutions et les changements AI7	3+3=6

Tableau 6.2 : Objectifs Informatiques Priorisés

Résultats de comparaison:

En comparant les résultats des experts et ceux de la plate-forme, nous pouvons en déduire que les 2/3 des processus informatiques experts sont proposés par la solution. Mais il y a une différence dans les priorités entre les deux résultats.

La différence entre les deux résultats est due au calcul de la priorité; PO2 proposé par l'expert figure dans la solution donnée par la plate-forme, mais avec une valeur de priorité inférieure.

En perspective, nous améliorons la fonction de priorité et son intégration dans le système expert de Matchmaking.

VI.7.6. Evaluation de la plateforme

Dans cette partie nous effectuons une évaluation de la plateforme pour juger de sa performance et son efficacité pour les utilisateurs potentiels. Nous présentons, dans un premier temps une évaluation quantitative de la couche médiation de la plateforme, à travers ses temps de réponse par rapport à des requêtes de différentes directions. Nous présentons, ensuite l'évaluation qualitative de la plateforme par rapport aux points définis à la proposition de la solution, à savoir: l'hétérogénéité des composants en question, leur distribution et leur autonomie.

VI.7.6.1. Evaluation quantitative

L'évaluation du temps de réponse de la couche de médiation n'est pas une finalité dans la mesure où elle ne remet pas en question l'efficacité de la solution. Cependant, elle permet de valoriser ses performances du moment que l'utilisation de cette plateforme se fait plutôt dans un contexte temps réel pour donner aux responsables métiers des solutions d'expert aux éventuels problèmes.

L'évaluation est faite sur un pc doté du système d'exploitation Windows 7 avec un processeur Intel Core i3, 2,53 GHz et 2,3Go de mémoire. Nous avons varié le mode de comparaison à savoir l'égalité, le Plug In et la subsumption et aussi le nombre de demande pour évaluer le comportement du système face à des charges différentes et nous avons défini les intervalles [1, 5] [5,10] [10,20] et >20.

Une autre variante peut éventuellement être testée à savoir le nombre de résultats à proposer par demande. Ce scénario permet d'évaluer l'impact des nouvelles versions du référentiel sur l'opération d'audit du SI.

Nous obtenons les résultats ci-après :

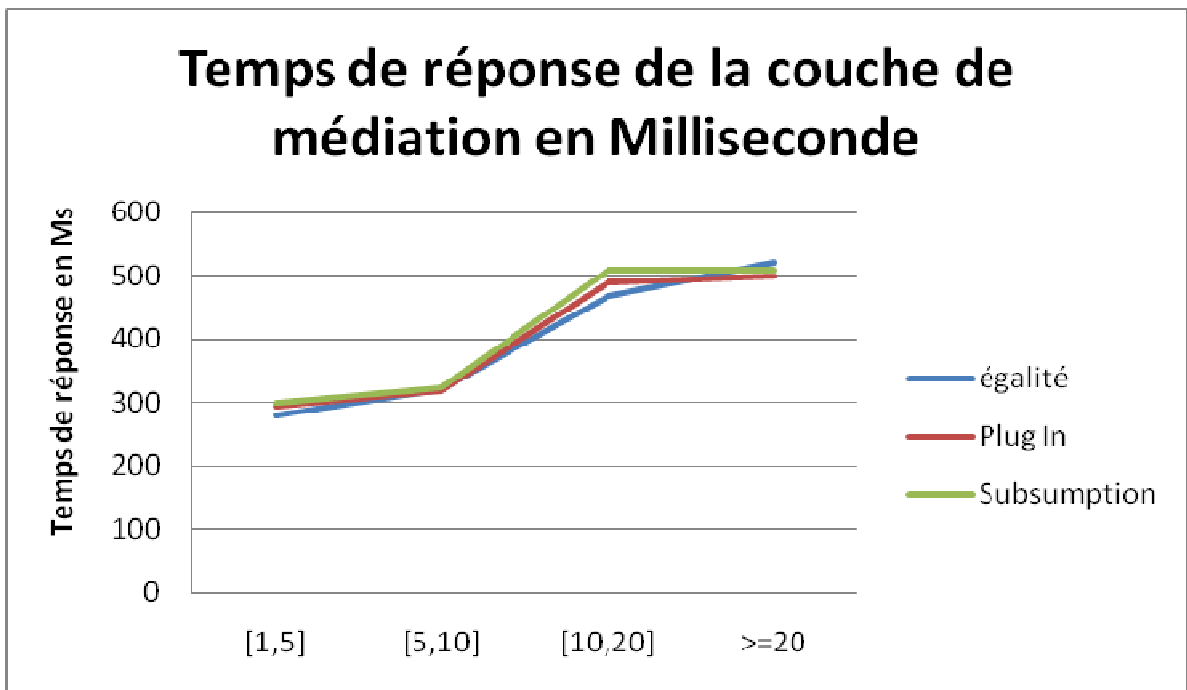


Figure 67: Temps de Réponse de la Couche de Médiation

Les résultats montrent que la plateforme est performante dans l'ensemble puisque les résultats sont en millisecondes, d'autant plus que la différence est minimale entre les 3 mondes ce qui montre une indifférence à la complexité des requêtes traitées. Un troisième point est que l'évolution du temps de réponse par rapport au nombre de demandes est linéaire par partie ce qui reflète une complexité raisonnable des algorithmes avec l'augmentation du nombre d'itérations.

VI.7.6.2. Evaluation qualitative

Dans cette partie, nous évaluons qualitativement la plateforme EAS IT-Stratégie selon les trois critères qualitatifs suivants: l'hétérogénéité, la distribution et l'autonomie.

a) Hétérogénéité :

La plateforme permet plusieurs hétérogénéités à savoir :

- L'hétérogénéité des composants du SI et des agents interrogés pour chacune des couches autorisées par le Workflow inter-organisations mis en place. En effet, quelque soit le type de composants (Module d'ERP, Client lourd, Application Web, etc), il est encapsulé dans un agent Workflow SI qui permet de remonter les objectifs métiers à évaluer en temps réel par la recherche d'agent Framework capable de les mesurer.
- L'hétérogénéité des informations est possible grâce au système multi-agent de Matchmaking doté du moteur sémantique ; l'utilisateur peut s'exprimer librement autour des incidents, problèmes, ou perspectives métier concernant la partie du SI qu'il utilise sans se soucier du bagage linguistique. Dans le même cadre de l'hétérogénéité de l'information, la plateforme permet un échange de message.
- L'hétérogénéité des modes de communication des agents de la plateforme par envoi de message et par partage d'information. Le langage ACL utilisé permet aussi des

descriptions diversifiées de services (offre ou demande), sans contrainte sur les protocoles de communication des agents.

b) Distribution

Tout d'abord, l'utilisation des WIO avec des systèmes multi-agents assure une distribution de l'architecture, comme c'est un atout technique majeur commun entre les deux.

D'autre part, il s'agit d'une plateforme WEB en J2EE permettant le lancement de plusieurs sessions distantes en même temps pour des clients légers, d'où la distribution de la solution. D'autant plus que les agents sont déployés sur la plateforme Madkit qui utilise le protocole TCP pour gérer la distribution et les sockets dans la gestion des messages entre eux.

Nous concluons donc, la solution en question assure plusieurs aspects de distribution.

c) Autonomie

Les systèmes multi-agents sont caractérisés par leur autonomie, en matière d'exécution et de résolution de problème. Leur utilisation dans cette solution assure l'autonomie de cette dernière sur le plan technique. D'autre part, l'utilisation du WIO de type lâche assure l'autonomie des organisations mises en jeu, comme ce dernier permet d'orchestrer des solutions non préalablement définie.

VI.7.7. Conclusion

Dans la première partie de ce chapitre, nous avons traité la conception de la **plateforme EAS IT-Stratégie** en utilisant le langage AUML pour la partie multi-agent et le langage UML pour la partie objet. Cela nous a permis de modéliser les cas d'utilisations, les scénarios, de décrire le rôle des acteurs humains et logiciels en évoquant les différents diagrammes.

Dans la seconde partie, nous avons présenté les interfaces de la plateforme **EAS IT-Stratégie**, en décrivant ses différentes fonctionnalités. Ensuite, nous avons soumis cette plateforme à une série de tests effectués sur des cas réels, ainsi qu'une évaluation qualitative et une évaluation quantitative suites auxquelles nous avons constaté que **EAS IT-Stratégie** version 3, répond parfaitement aux objectifs de gouvernance SI demandés et permet de fournir à la plateforme globale **EAS ITGRC** des entrées stratégiques adéquates pour lancer les traitements de choix et d'exécution du meilleur référentiel.

CONCLUSION GENERALE

Conclusion générale

Bilan

L'objectif de cette thèse de doctorat est de développer une plateforme de gouvernance des systèmes d'information à base de Workflow Inter-organisationnel et de systèmes multi-agents. En effet, les plateformes de gouvernance SI existantes actuellement, malgré leurs points forts techniques et technologiques, présentent plusieurs limitations d'intégration et de mise à jour.

De nombreux piliers de la littérature informatiques ont été invoqués dans la solution proposée à savoir:

- Workflows Inter-organisation ;
- Système multi-agent ;
- Médiation ;
- Web sémantique et ontologies.

Le choix de chaque notion a une valeur ajoutée pour cette solution. En fait, les Workflows Inter-organisation fournissent l'orchestration des composants hétérogènes d'un SI de manière autonome. Les systèmes multi-agents assurent la dimension intelligente de la solution avec un protocole de communication de haut niveau.

La Médiation dans les SMA donne un modèle théorique adéquat de services échangés entre les entités intelligentes de l'offre et de la demande, et permet de mettre en place les mécanismes nécessaires pour une meilleure correspondance.

Les Ontologies offrent l'alignement sémantique des parties prenantes avec le vocabulaire des cadres de référence et des experts GSI, chose qui est fortement sollicitée dans le contexte de la mesure de la maturité en matière de Gouvernance de l'entreprise en générale et des systèmes d'information en particulier.

L'idée de base, qui a inspiré notre travail, consistait à mener à bien en temps réel, l'un des piliers de la Gouvernance d'un Système d'Information de taille et de type quelconque à savoir l'alignement des TIC avec la stratégie métier de l'entreprise,

L'avantage d'une telle approche est de permettre à l'ensemble des parties prenantes de l'entreprise d'intervenir par le biais de cette plateforme, d'exprimer un besoin stratégique quelque soit son ampleur et de pouvoir récupérer sa projection et son implémentation au niveau de la composante du SI concernée. De plus, le système peut anticiper les problèmes récurrents, par le biais de l'apprentissage des agents spécialisés, ainsi que tirer profit des mises à jours des référentiels de pointes en matière de GSI.

D'autre part, le moteur sémantique de notre plateforme basé sur une ontologie de gouvernance des SI conçue et implémentée spécialement pour cette thèse de doctorat est aussi un atout majeur de cette solution. Du moment qu'elle permet une large compréhension du vocabulaire de ce domaine d'actualité et aussi une interprétation intelligente des besoins stratégiques des différentes directions de l'entreprise, en requête de spécialiste GSI.

En effet, de nos jours, cette correspondance SI-métier au sein de l'entreprise, avec l'évolution des nouvelles technologies et le développement des exigences et des contraintes métiers, est de plus en plus complexe. Elle nécessite obligatoirement un recours au conseil IT ou d'audit interne pour corriger le passé, mieux gérer le quotidien et réussir le futur.

Pour finir, il est à noter que la solution proposée offre une facilité de déploiement et d'utilisation, grâce à son architecture modulaire, à la portabilité de son langage de développement, et à l'interactivité de ses agents qui assurent la communication synchrone et asynchrone aussi bien entre les différentes couches qu'avec l'utilisateur final et la plateforme globale EAS-IT GRC.

Originalité

Ce travail présente des originalités par rapport aux solutions existantes à savoir :

1. La combinaison des workflow Inter-organisationnels et des systèmes multi-agents dans le domaine de la gouvernance des SI ;
2. La conception intelligente des composantes du référentiel COBIT et leur encapsulation dans des agents autonomes et ce en profitant de son aspect orienté processus proche de la notion d'agent;
3. La traduction de la demande de l'utilisateur final en langage interprétable par la communauté IT GRC sans avoir besoin de partie tiers (auditeur ou conseillé) pour effectuer cette opération.

Limites

Le présent travail présente quelques limitations à savoir :

1. Le manque de précision de protocole de communication entre les agents en générale et les serveurs de connexions en particulier, comme nous les avons implémentés sur la plateforme Madkit. Il est par contre nécessaire de définir différemment le protocole dans l'échange entre les agents de chaque couche et inter-couches de la plateforme.
2. L'ontologie développée est en langue française, alors que les référentiels une fois mis à jour sont pendant une durée en anglais, ce qui ne permet pas de bénéficier tout de suite des avantages de la nouvelle version. Il fallait donc penser au multilinguisme de la solution.
3. Le problème d'interopérabilité sémantique (au delà des termes du domaine GSI) entre les services offres et demandes se posent face à la richesse de la langue française et la variété des synonymes et des expressions que l'utilisateur peut utiliser. Nous gérons cet aspect actuellement en enrichissant les fichiers de langue du serveur « Solr » utilisé en implémentation, mais il faut résoudre ce problème normalement au niveau architectural.
4. La solution ne permet pas encore de traiter les demandes d'information particulières sur l'entreprise, de la part de la couche traitement de la plateforme générique. Mais, nous

supposons que toutes les couches de traitement lancent une même requête sur la configuration statique, ce qui n'est pas forcément le cas.

Perspectives

Ce travail ouvre les perspectives de recherche suivantes:

- Conception d'un protocole et d'une infrastructure de négociation entre les agents Workflow SI et les Agents référentiels au sein du WIO après le matchmaking pour mieux faire correspondre l'offre et la demande.
- Définition de Services Web, à partir des demandes répondues pour une large utilisation dans le cadre de l'ouverture de l'entreprise sur le marché.
- Généralisation de l'ontologie ITGovernanceOntology pour couvrir l'ensemble des référentiels de l'IT GRC.
- L'étude de fonctions de priorisation non linéaire dans la partie matchmaking sémantique.
- La généralisation du modèle de matchmaking sémantique pour l'ensemble des référentiels IT-GRC.

Enfin, nous espérons que ce travail contribuera au développement du domaine de la gouvernance des Technologies de l'information.

Bibliographie

- Alavi, M., & Carlson, P. (1992). A review of MIS research and disciplinary development. *Journal of Management Information Systems* , 45-62.
- Albert, M. (2005). *Elaboration d'une passerelle AUMML vers Jade*. Ecole Polytechnique de l'Université de Nantes.
- Amirreza, T. (2009). *Modeling and Verification of Web Service Composition Based Interorganizational Workflows*. Thèse de doctorat. Université de Vienna.
- Ammar, N., Malik, Z., Medjahed, B., & Alodib, M. (2015). K-Anonymity Based Approach for Privacy-Preserving Web Service Selection. *In Web Services (ICWS), IEEE International Conference on (pp. 281-288)*. IEEE .
- Andonoff, E., Bouaziz, W., Hanachi, C., & Bouzguenda, L. (2009). An agent-based model for autonomic coordination of inter-organizational business processes. *Informatica*, 20(3), 323-342 .
- Bechhofer, S., Horrocks, I., Goble, C., & Stevens, R. (2001). OilEd: a reason-able ontology editor for the semantic web. *the Joint German/Austrian Conf. on Artificial Intelligence, number 2174 in LNAI*. Springer-Verlag , 396–408.
- Bellur, U., & Kulkarni, R. (2007). Improved matchmaking algorithm for semantic web services based on bipartite graph matching. *ICWS 2007. IEEE International Conference on Web Services* .
- Benhadou, S. (2010). *Réalisation d'une Plateforme de Sécurité et de Détection d'Intrusion*. Casablanca: Thèse de doctorat, ENSEM, Université Hassan II Ain Cock.
- Bernroider, E. W. (2008). IT governance for enterprise resource planning supported by the DeLone–McLean model of information systems success. *Information & Management*, 45(5) , 257-269.
- Biazzo, S. &. (2003). Process management practices and quality systems standards: risks and opportunities of the new ISO 9001 certification. *Business Process Management Journal* .
- Bingi, P. S. (1999). Critical issues affecting an ERP implementation. *implementation. IS Management*, 16(3) , 7-14.
- Boissier, O., & Demazeau, Y. (1992). A distributed artificial intelligence view on general purpose vision systems. *ACM SIGOIS Bulletin*, 13(3), 16 .
- Bordini, R. H., Dix, J., & Seghrouchni, A. (2005). Multi-Agent Programming. M. Dastani (Ed.). *Springer Science+ Business Media, Incorporated* .
- Boukhedouma, S., Alimazighi, Z., Oussalah, M., & Tamzalit, D. (2012). *Adaptability of Service Based Workflow Models: The “Chained Execution” Architecture*. *In Business Information Systems (pp. 96-107)*. Berlin Heidelberg: Springer.
- BOURHIS, P. (2010). *ETAT DE L'ART SUR LES TECHNOLOGIES DU WEB SEMANTIQUE*.

- Broussard, F. W. (2007). Configuration and Change Management for IT Compliance and Risk Management: The Tripwire Approach. *IDC* .
- Brown, C. V. (1997). Examining the emergence of hybrid IS governance solutions: Evidence from a single case site. *Information systems research*, 8(1) , 69-94.
- Bussmann, S., & Demazeau, Y. (1994). An agent model combining reactive and cognitive capabilities. In *Intelligent Robots and Systems' 94. 'Advanced Robotic Systems and the Real World', IROS'94. Proceedings of the IEEE/RSJ/GI International Conference on (Vol. 3, pp. 2095-2102)*. *IEEE* .
- Chakir, A., Chergui, M., Elhasnaoui, S., Medromi, H., & Sayouti, A. (2016). Intelligent Platform to Select The Best IT-GRC Framework For Treatment The Needs Of Stakeholders. *International Journal of Applied Engineering Research (IJAER) Volume 11, Number 12* , 7829-7835.
- Chakir, A., Chergui, M., Medromi, H., & Sayouti, A. (2016). A decision approach to select the best framework to treat an IT problem by using multi-agent system and expert systems. *Advances in Ubiquitous Networking. Springer Singapore* , 499-511.
- Charlet, J., Bachimont, B., & Troncy, R. (2004). Ontologies pour le Web sémantique. *Revue Information, Interaction, Intelligence I3* .
- Chebbi, I. (2007). *CoopFlow: une approche pour la coopération ascendante de workflows dans le cadre des entreprises virtuelles*. Thèse de doctorat , Institut National des Télécommunications, France.
- Chergui, M. S. (2015). MULTI-AGENT PLATEFORME FOR COBIT IMPLEMENTATION. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY (IJESRT)* .
- Chergui, M., Chakir, A., Sayouti, A., & Medromi, H. (2015). Matchmaking Expert system based on Intelligent Agents and semantic inference engine: Application to IT Governance Context. *International Conference on Advanced Information Technology, Services and Systems (AIT2S'15) FST Settat Morocco* .
- Chergui, M., Chakir, A., Sayouti, A., & Medromi, H. (2016). Strategic IT Governance Platform Based on Matchmaking Multi-Agent System and Loose Inter-Organizational Workflows. *International Journal of Computer Science and Information (IJCSIS) IJCSIS Volume 14 No. 5 ISSN 1947 5500* .
- Chergui, M., Medromi, H., & Sayouti, A. (2014). Intelligent audit of information systems by inter-organizational workflow. *Next Generation Networks and Services (NGNS'14), Fifth International Conference. IEEE* , 1-6.
- Chergui, M., Nahla, H., Chakir, A., Elhasnaoui, S., Sekhara, Y., & Medromi, H. (2016). Empirical Study: Moroccan Information systems specificities for better IT Governance. *International Journal of Advanced Engineering, Management and Science (IJAEMS) [Vol-2, Issue-5] Infogain Publication (Infogainpublication.com) ISSN , 2454-1311*.

- Chergui, M., Nahla, H., Chakir, A., Elhasnaoui, S., Sekhara, Y., & Medromi, H. (2016). Empirical Study: Moroccan Information systems specificities for better IT Governance”. *International Journal of Advanced Engineering, Management and Science (IJAEMS) [Vol-2, Issue-5]* .
- Chergui, M., Sayouti, A., & Medromi, H. (2014). Inter-organizational Workflow for Intelligent Audit of Information Technologies in terms of Enterprise Business Processes. *International Journal of Advanced Computer Science and Applications(IJACSA)*, 5(5) .
- Chergui, M., Sayouti, A., & Medromi, H. (2015). IT Governance Ontology Building Process Example of developing Audit Ontology. *International Journal of Computer Techniques (IJCT) V2 (1)* , 134-141.
- Chergui, M., Sayouti, A., & Medromi, H. (2013). IT Governance through an Inter-Organizational Workflow based on Multi-Agent System. *International Journal of Applied Information Systems (IJ AIS) 6(6):10-16. Published by Foundation of Computer Science, New York, USA* .
- Chergui, M., Sayouti, A., & MEDROMI, H. (2015). IT Governance Ontology Building Process: Example of developing Audit Ontology. *International Journal of Computer Techniques (IJCT) V2 (1)* . , 134-141.
- Coraux, G. (2007). Infogérance: les risques du mariage. *L'Expansion Management Review*, 127(4) , 119-129.
- Curry, L. (2004). Just what does responsible board governance mean today? *Canadian Journal of Dietetic Practice and Research* , 98-98.
- Dahlberg, T. &. (2007). IT governance maturity and IT outsourcing degree: an exploratory study. *HICSS 2007. 40th Annual Hawaii International Conference. IEEE.* , 236a-236a.
- Defarges, P. M., & Bertucci, J. Y. (2003). La gouvernance. *Presses universitaires de France* .
- Divitini, M., Hanachi, C., & Sibertin-Blanc, C. (2001). Inter-Organizational Workflow for Enterprise Coordination. *Coordination of Internet Agents, chapter 15, A. Omicini, F. Zambonelli, M. Klusch, R. Tolksdorf (Eds), Springer Verlag* .
- Ehsan, N. M. (2010). Comparative study for PMBOK & CMMI frameworks and identifying possibilities for integrating ITIL for addressing needs of IT service industry. *In Management of Innovation and Technology (ICMIT) IEEE International Conference on IEEE.* , 113-116.
- Elhasnaoui, S., Chakir, A., Chergui, M., Iguer, H., Faris, S., & Medromi, H. (2015). Building an integrated IT GRC platform based on multi agent system. *International Journal of Engineering and Innovative Technology” (IJEIT). Issue 4, Vol 8* .
- ELHASNAOUI, S., CHERGUI, M., CHAKIR, A., SEKHARA, Y., Nahla, H., & Medromi, H. (2016). Empirical Study on the Interaction and Workflow Management between Information System and Business Departments of an Organization to Integrate IT GRC Processes: Case of Moroccan organizations. *International Journal of Engineering Research and Management* .

- ESPINASSE, B. (n.d.). Communication et langages de communication dans les SMA. Aix-Marseille Université (AMU) LSIS UMR CNRS 7296 .
- Ferber, J., & Gutknecht, O. (1998). A meta-model for the analysis and design of organizations in multi-agent systems. *In Multi Agent Systems. Proceedings. International Conference on (pp. 128-135). IEEE .*
- Ferguson, I. A. (1992). Touring machines: Autonomous agents with attitudes. *Computer, 25(5) , 51-55.*
- Fernández-López, M., Gómez-Pérez, A., & Juristo, N. (1997). Methontology: from ontological art towards ontological engineering. *AAAI Technical Report SS-97-06 .*
- Ferstl, O. K. (2006). *Sinz E.J.: Grundlagen der Wirtschaftsinformatik. 5.* München: überarb. und erw. Aufl., Oldenbourg- Verlag.
- Fimbel, E. (2001). *L'externalisation des systèmes d'information: les facteurs de succès .* Reims: Doctoral dissertation.
- Finin, T., & Fritzson, R. (1994). KQML: a language and protocol for knowledge and information. *In Proceedings of the Thirteenth International Workshop on Distributed Artificial Intelligence, pages 126-136, Lake Quinalt .*
- Forrester, R. (2011). <https://www.forrester.com/marketing/about/about-us.html>.
- Garrity, J. T. (1963). Getting the Most out of your computer. A Survey of company approaches and results. *McKinsey and Company, Incorporated, Management Consultants , 270.*
- Georgeff, M., Pell, B., Pollack, M., Tambe, M., & Wooldridge, M. (1998). *The belief-desire-intention model of agency.* Berlin Heidelberg: In Intelligent Agents V: Agents Theories, Architectures, and Languages (pp. 1-10). Springer.
- Gerrard, M. (2009). *IT Governance, a Flawed Concept: It's Time for Business Change Governance.* Gartner Research.
- Gherbi, T., Borne, I., & Meslati, D. (2013). Towards an MDE Methodology to Develop Multi-Agents Systems Including Mobile Agents. *In 8th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2013) (pp. 45-55). SciTePress .*
- Gibbons, R. &. (2015). *Formal Measures in Informal Management: Can a Balanced Scorecard Change a Culture?*
- Gosling, J. (2000). *The Java language specification.* Addison-Wesley Professional.
- Graja, Z. (2009). *méthode formaad: vers une approche dirigée par les modèles pour la conception d'applications à base d'agents.* université de Sfax faculté des sciences économiques.
- Gruninger, M., Schlenoff, C., Knutilla, A., & Ray, S. (1997). Using process requirements as the basis for the creation and evaluation of process ontologies for enterprise modeling. *ACM SIGGROUP Bulletin Special Issue on Enterprise Modelling, 18(3) , 52-55.*

- Guechtouli, M. A. (2014). Quels objectifs d'élaboration des politiques de sécurité de l'information dans les organisations? 396.
- Guizani, A. (2016). *Approche multi-agents pour la conception optimale des systèmes mécatroniques*. Doctoral dissertation, Université Paris-Saclay.
- Haes, S. D. (2008). Analysing the relationship between IT governance and business/IT alignment maturity. *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual IEEE* , 428-428.
- Hernandez, N. (2005). *Ontologies de domaine pour la modélisation du contexte en recherche d'information*. Toulouse: Thèse de doctorat, Université de Toulouse.
- Hsieh, F. S., & Lin, J. B. (2016). A self-adaptation scheme for workflow management in multi-agent systems. *Journal of Intelligent Manufacturing*, 27(1), 131-148 .
- <https://www.forrester.com/marketing/about/about-us.html>. (n.d.).
- Huber, M. J. (1999). JAM: A BDI-theoretic mobile agent architecture. *In Proceedings of the third annual conference on Autonomous Agents (pp. 236-243)*. ACM .
- Huget, M. P., & Koning, J. L. (2001). *Une ingénierie des protocoles d'interaction pour les systèmes multi-agents*. (Doctoral dissertation, ANRT).
- Iguer, H., Medromi, H., Sayouti, A., & Tallal, S. (2016). Including EAS-SGR IT Risk framework in an IT GRC global framework. *Advances in Ubiquitous Networking. Springer Singapore* , 525-533.
- ISACA. (2011). CobiT Mapping : mapping of ITIL V3 with CobiT 4.1.
- ISACA. (Management briefing from ITGI and OGC). *Aligning CobiT®4.1, ITIL®V3 and ISO/IEC 27002 for Business Benefit*. 2012.
- Ives, B., Hamilton, S., & Davis, G. B. (1980). A framework for research in computer-based management information systems. *Management science*, 26(9) , 910-934.
- Jennings, N. R., Sycara, K., & Wooldridge, M. (1998). A roadmap of agent research and development. *Autonomous agents and multi-agent systems*, 1(1) , 7-38.
- Jiajin Le ruiqiang, G., & Dehua, C. (2005). Matching semantic web services across heterogenous ontologies. *CIT 05:the Fifth international conference on computer and information technology* .
- Jouanot, F. (2000). Un modèle sémantique pour l'interopérabilité de systèmes d'information. *13th Congress Inforsid* , 16-19.
- Klein, M., Fensel, D., van Harmelen, F., & Horrocks, I. (2000). The Relation between Ontologies and Schema-Languages: Translating OIL-Specifications to XML-Schema. *the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI-00, Berlin, Germany* .

- Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval Research Logistic Quarterly* , 83–97.
- Le-Hung Vu, F. P., Hauswirth, M., & Aberer, K. (2006). A search engine for qosenabled discovery of semantic web services. *International Journal of Business Process Integration and Management*. Vol. 1, No.4 , 244– 255.
- Lemos, A. L., Daniel, F., & Benatallah, B. (2015). Web Service Composition: A Survey of Techniques and Tools. *ACM Computing Surveys (CSUR)*, 48(3), 33 .
- Leymann, F., & Roller, D. (2000). *Production workflow: concepts and techniques*.
- Lieberman, H. (1997). Autonomous interface agents. *ACM SIGCHI Conference on Human factors in computing systems*. ACM , 67-74.
- Lozano-Tello, A., & Gómez-Pérez, A. (2004). Ontometric: A method to choose the appropriate ontology. *Journal of Database Management*, 2(15) , 1-18.
- Luke, S., & Heflin, J. (1997). *Proposed Specification*. Retrieved from <http://www.cs.umd.edu/projects/plus/SHOE/spec.html>
- Lunardi, G. L. (2014). The impact of adopting IT governance on financial performance: An empirical analysis among Brazilian firms. *International Journal of Accounting Information Systems*, 15(1) , 66-81.
- Lunardi, G. L. (2009). The financial impact of IT governance mechanisms' adoption: an empirical analysis with Brazilian firms. In System Sciences. *HICSS'09. 42nd Hawaii International Conference IEEE*. , 1-10.
- Matthias, B., & Burkhard, F. (2006). Semantic matchmaking using ranked instance retrieval. *SMR '06: Proceedings of the 1st International Workshop on Semantic Matchmaking and Resource Retrieval, Co-located with VLDB, 178 of CEUR Workshop Proceedings* .
- Medelyan, O. (2009). *Human-competitive automatic topic indexing*. New Zeland: PhD thesis, department of computer sciences, University of Waikato.
- Mehandjiev, N., & Grefen, P. (2010). *Dynamic business process formation for instant virtual enterprises (p. 2)*. London: Springer.
- Minsky, M. (1979). The society theory of thinking. *Artificial intelligence: an MIT perspective*, 1 , 421-450.
- Mintzberg, H. (1979). The Structuring of Organizations. *Englewoods Cliffs* .
- Moussaoui, R. (2015). *CONCEPTION ET RÉALISATION D'UNE PLATEFORME DES SYSTEMES COMMANDES EN RÉSEAU APPLIQUÉE A LA ROBOTIQUE MOBILE*. Casablanca: Rapport de thèse, ENSEM, Université Hassan II Ain Cock.
- Moutaouakkil, F. (2010). *Conception et Réalisation d'une plateforme de Contrôle autonome distribuée : Application en Robotique Mobile*. Casablanca: Thèse de doctorat, ENSEM, Université Hassan II Ain Cock.

- Müller, J. P., Pischel, M., & Thiel, M. (1994). Modeling reactive behaviour in vertically layered agent architectures. *In Intelligent Agents. Springer Berlin Heidelberg* , 261-276.
- Nahla, H., Medromi, H., Chergui, M., & Elhassnaoui, S. (2016). Analysis of interactions between Information system, Communication and Marketing in Organizations. *International Journal of Advanced Engineering ,Management and Science IJAEMS-AUG--1, Vol-2, .*
- Northrop, L., Feiler, P., Gabriel, R. P., Goode-nough, J., Linger, R., Longstaff, T., et al. (2006). Ultra-large-scale systems - the software challenge of the future. *Technical report, Software Engineering Institute, Carnegie Mellon .*
- Noy, N. F., Crubézy, M., Ferguson, R. W., Knublauch, H., Tu, S. W., Vendetti, J., et al. (2003). Protege-2000: an open-source ontology-development and knowledge-acquisition environment. *In AMIA Annu Symp Proc. Vol. 953 , 953.*
- Ouyang, C., Adams, M., Wynn, M. T., & ter Hofstede, A. H. (2015). Workflow management. *In Handbook on Business Process Management 1 (pp. 475-506). Springer Berlin Heidelberg .*
- Patil, S., & Newcomer, E. (2003). ebXML and Web services. *Internet Computing, IEEE, 7(3), 74-82 .*
- Payne, T., Paolucci, M., Kawmura, T., & Sycara, K. (2002). Semantic matching of web service capabilities. *the International Semantic Web Conference. Springer Verlag, LNCS .*
- Perrey, R., & Lycett, M. (2003). Service-oriented architecture. *In Applications and the Internet Workshops, Proceedings. 2003 Symposium. IEEE , 116-119.*
- Peter, F., Schneider, P., Hayes, P., & Horrocks, I. (2004). *La sémantique et la syntaxe abstraite du langage d'ontologie Web OWL*. Retrieved from <http://www.w3.org/TR/2004/REC-owl-semantics-20040210/>
- Peterson, R. (2004). Crafting information technology governance. *. Information Systems Management, 21(4) , 7-22.*
- PMI. (2008). A Guide to the Project Management Body of Knowledge (PMBOK Guide) (4rd ed.). *Project Management Institute .*
- Remus, U. &. (2000). The role of process-oriented enterprise modeling in designing process-oriented knowledge management systems. *In Proceedings of the AAAI Symposium on Bringing Knowledge to Business Processes. Stanford, CA, USA (pp. 30-36) .*
- Rousset, A., Herrmann, B., & Lang, C. (2014). Étude comparative des plateformes parallèles pour systèmes multi-agents. *In ComPAS 2014 : conférence en parallélisme, architecture et systèmes .*
- Sayouti, A., Moutaouakkil, F., & Medromi, H. (2012). The Interaction-Oriented Approach for Modeling and Implementing Multi-Agents Systems. *International Review on Computers and Software (I.RE.CO.S), Vol. 5, N. 2 .*
- Sayouti, A., Qrichi Aniba, F., & Medromi, H. (2009). Modeling Autonomous Mobile System with Agent Oriented Approach. *International Journal of Computer Science and Network Security (IJCSNS), Vol. 9, N. 9 , 316 à 321.*

- Seifert, V. (2011). *A comparison of algorithms for automated keyword generation*. Bachelor-Thesis (ehem. UdSSR).
- Shortliffe, E. (. (2012). Computer-based medical consultations. *MYCIN (Vol. 2)*. Elsevier.
- Stearns, M. Q., Price, C., Spackman, K. A., & Wang, A. Y. (2001). SNOMED clinical terms: overview of the development process and project status. *AMIA Symposium*. American Medical Informatics Association , 662.
- Sure, Y., Erdmann, M., Angele, J., Staab, S., Studer, R., & Wenke, D. (2002). OntoEdit: Collaborative ontology development for the semantic web. *Springer Berlin Heidelberg* , 221-235.
- Thatte, S. (2001). XLANG: Web services for business process design. *Microsoft Corporation* .
- Themistocleous, M., Irani, Z., O'Keefe, R. M., & Paul, R. (2001). ERP problems and application integration issues: An empirical survey. *the 34th Annual Hawaii International Conference on IEEE* , 10.
- unspsc. (2014). Retrieved from www.unspsc.org
- Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *The knowledge engineering review*, 11(02) , 93-136.
- Van Der Aalst, W. M., & Ter Hofstede, A. H. (2005). YAWL: yet another workflow language. *Information systems*, 30(4), 245-275 .
- van der Aalst, W., ter Hofstede, A., Kiepuszewski, B., & Barros, A. (2003). Workflow Patterns. *In the International Journal on Distributed and Parallel Databases*, 34(1) p. 5–51 .
- Van Der Vet, P. E., & Mars, N. J. (1998). Bottom-up construction of ontologies. *Knowledge and Data Engineering, IEEE Transactions on*, 10(4) , 513-526.
- Weil, P. &. (2004). IT governance. *Harvard Business School Publishing, Massachusetts* .
- Wiederhold, G. (1992). Mediators in the architecture of future information systems. *IEEE Computer Magazine*, Vol. 25, No. 3 , 38-49.
- Wooldridge, M., & Jennings, N. R. (1994). *Agent theories, architectures, and languages: a survey*. Berlin Heidelberg: In Intelligent agents (pp. 1-39). Springer.
- Wooldridge, M., Jennings, N. R., & Kinny, D. (2000). The Gaia methodology for agent-oriented analysis and design. . *Autonomous Agents and multi-agent systems* , 285-312.
- Xue, Y., Liang, H., & Boulton, W. R. (2006). Information Technology Governance in Information Technology Investment Decision Processes: The Impact of Investment Characteristics, External Environment, and Internal Context,. *Management Information Systems Quarterly*, 32(1), 67-96 .
- Zarcoski, B. (2015). Les entreprises virtuelles. *Oboulo. org Publications* .
- Zemmouri, E., Behja, H., Marzak, A., & Trousse, B. (2011). Intégration des connaissances de domaine dans un processus d'ECD multi-vues. *4èmes Journées Francophones sur les Ontologies JFO2011* , 73-85.

Zhang, S. &. (2013). An Examination of the Practicability of COBIT Framework and the Proposal of a COBIT-BSC Model. *Journal of Economics.* , 1-5.

ANNEXES

Annexe I : Correspondance Objectifs métiers, Objectifs Informatiques et Processus IT

Objectifs Informatiques et Processus IT

	Objectifs Informatiques	Processus									
		PO1	PO2	PO4	PO10	AI1	AI6	AI7	DS1	DS3	SE1
1	Réagir aux exigences métiers en accord avec la stratégie métiers	PO1	PO2	PO4	PO10	AI1	AI6	AI7	DS1	DS3	SE1
2	Réagir aux exigences de la gouvernance en accord avec les orientations du CA	PO1	PO4	PO10	SE1	SE4					
3	S'assurer de la satisfaction des utilisateurs finaux à l'égard des offres et des niveaux de services	PO8	AI4	DS1	DS2	DS7	DS8	DS10	DS13		
4	Optimiser l'utilisation de l'information	PO2	DS11								
5	Donner de l'agilité à l'informatique	PO2	PO4	PO7	AI3						
6	Déterminer comment traduire les exigences métiers de fonctionnement et de contrôle en solutions automatisées efficaces et efficientes	AI1	AI2	AI6							
7	Acquérir et maintenir fonctionnels des systèmes applicatifs intégrés et standardisés	PO3	AI2	AI5							
8	Acquérir et maintenir opérationnelle une infrastructure informatique intégrée et standardisée	AI3	AI5								
9	Se procurer et conserver les compétences nécessaires à la mise en œuvre de la stratégie informatique	PO7	AI5								
10	S'assurer de la satisfaction réciproque dans les relations avec les tiers	DS2									
11	S'assurer de l'intégration progressive des solutions informatiques aux processus métiers	PO2	AI4	AI7							
12	S'assurer de la transparence et la bonne compréhension des coûts, bénéfices, stratégie, politiques et niveaux de services des SI	PO5	PO6	DS1	DS2	DS6	SE1	SE3			
13	S'assurer d'une bonne utilisation et des bonnes performances des applications et des solutions informatiques	PO6	AI4	AI7	DS7	DS8					
14	Protéger tous les actifs informatiques et en être comptable	PO9	DS5	DS9	DS12	SE2					
15	Optimiser l'infrastructure, les ressources et les capacités informatiques	PO3	AI3	DS3	DS7	DS9					
16	Réduire le nombre de défauts et de retraitements touchant la fourniture de solutions et de services	PO8	AI4	AI6	AI7	DS10					
17	Protéger l'atteinte des objectifs informatiques	PO9	DS10	SE2							
18	Montrer clairement les conséquences pour l'entreprise des risques liés aux objectifs et aux ressources informatiques	PO9									
19	S'assurer que l'information critique et confidentielle n'est pas accessible à ceux qui ne doivent pas y accéder	PO6	DS5	DS11	DS12						
20	S'assurer que les transactions métiers automatisées et les échanges d'informations sont fiables	PO6	AI7	DS5							
21	S'assurer que les services et l'infrastructure informatique peuvent résister/se rétablir convenablement en cas de panne due à une erreur, à une attaque délibérée ou à un sinistre	PO6	AI7	DS4	DS5	DS12	DS13	SE2			
22	S'assurer qu'un incident ou une modification dans la fourniture d'un service informatique n'ait qu'un impact minimum sur l'activité	PO6	AI6	DS4	DS12						
23	S'assurer que les services informatiques sont disponibles dans les conditions requises	DS3	DS4	DS8	DS13						
24	Améliorer la rentabilité de l'informatique et sa contribution à la profitabilité de l'entreprise	PO5	DS6								
25	Livrer les projets en temps et dans les limites budgétaires en respectant les standards de qualité	PO8	PO10								
26	Maintenir l'intégrité de l'information et de l'infrastructure de traitement	AI6	DS5								
27	Assurer la conformité de l'informatique aux lois et règlements	DS11	SE2	SE3	SE4						
28	S'assurer que l'informatique fait preuve d'une qualité de service efficiente en matière de coûts, d'amélioration continue et de capacité à s'adapter à des changements futurs	PO5	DS6	SE1	SE3						

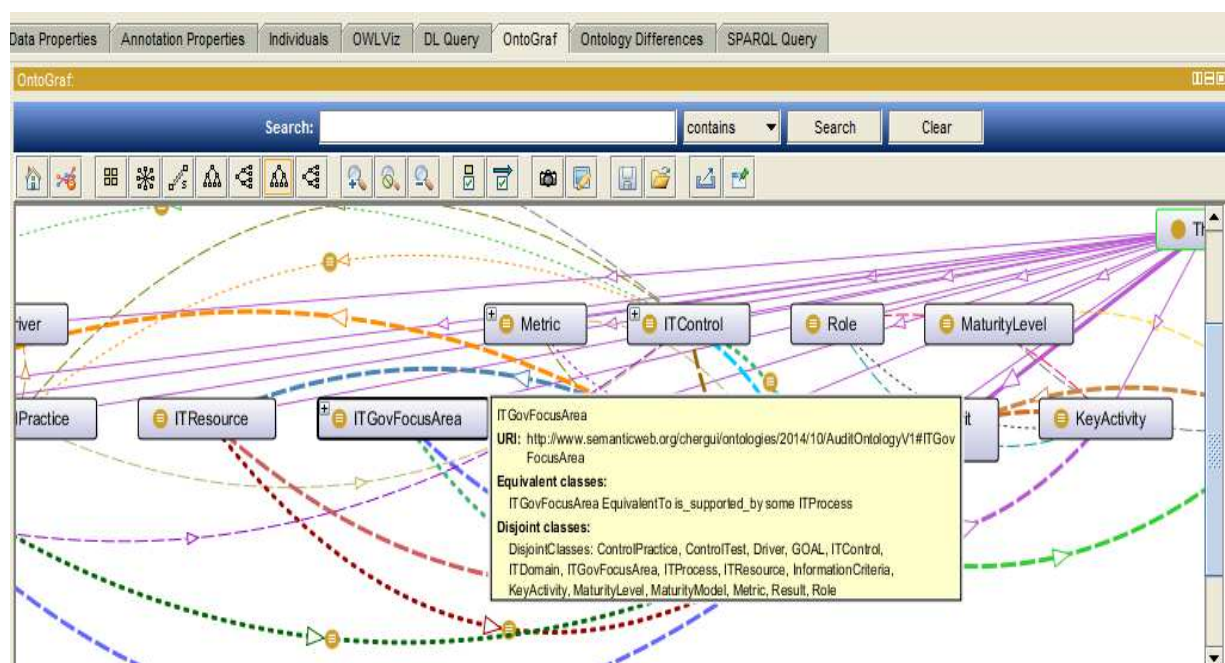
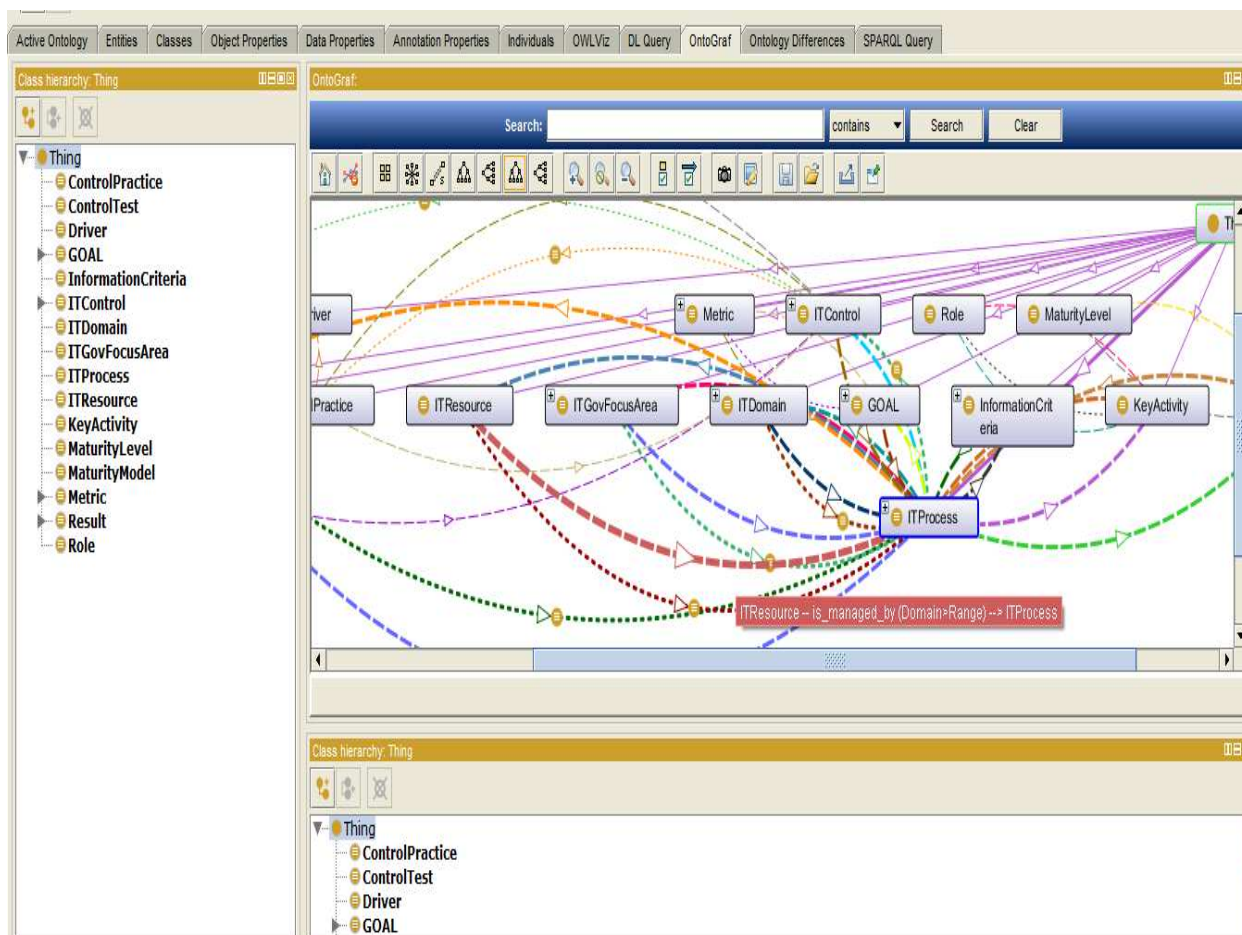
Objectifs métiers et Objectifs Informatiques

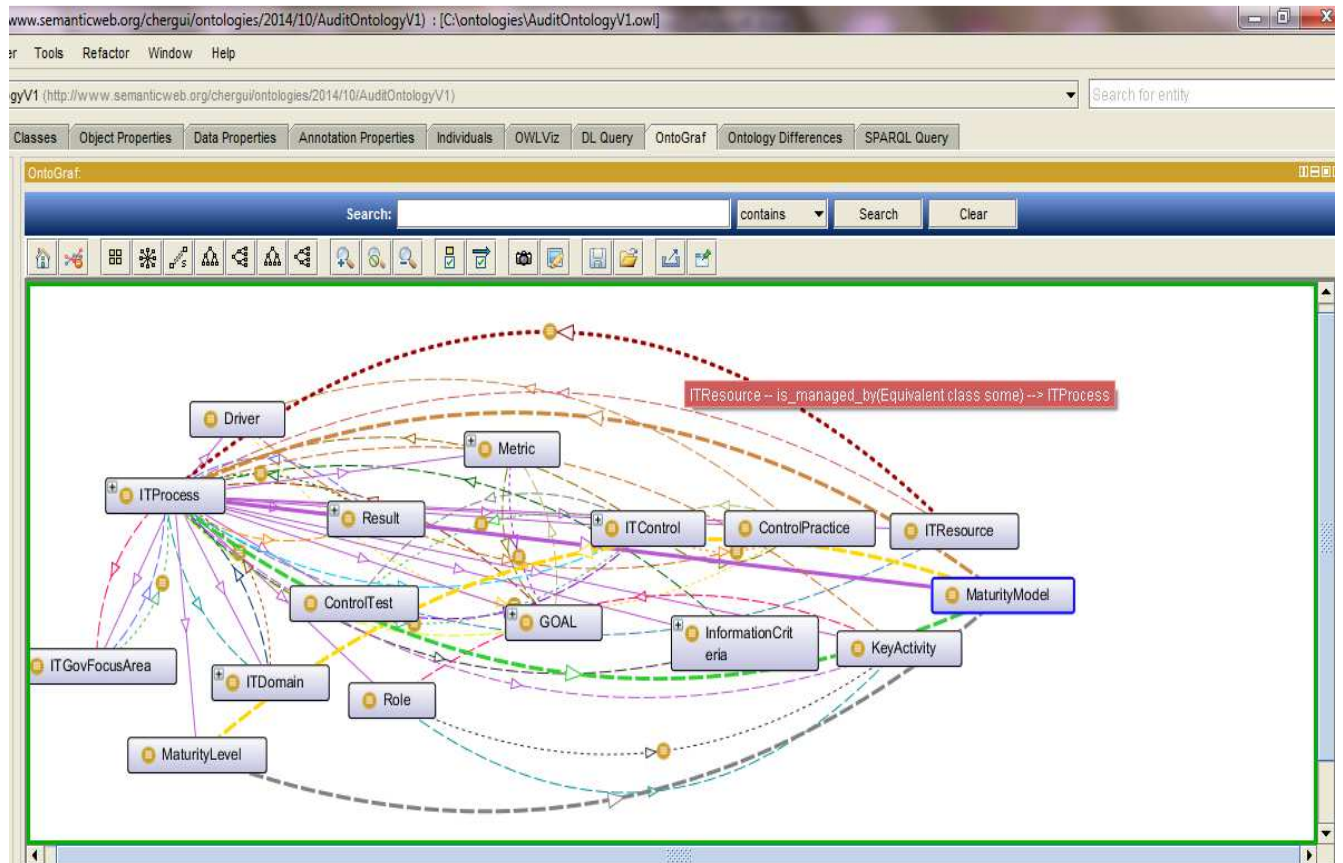
	Objectifs métiers	Objectifs Informatiques								
<i>Perspective financière</i>	1	Obtenir un bon retour sur investissement des investissements informatiques pour les métiers	24							
	2	Gérer les risques métiers liés aux SI	2	14	17	18	19	20	21	22
	3	Améliorer la gouvernance de l'entreprise et la transparence	2	18						
<i>Perspective client</i>	4	Améliorer l'orientation client et le service client	3	23						
	5	Offrir des produits et des services compétitifs	5	24						
	6	Assurer la continuité et la disponibilité des services	10	16	22	23				
	7	Développer l'agilité pour s'adapter aux modifications des exigences des métiers	1	5	25					
	8	Réussir à optimiser les coûts de la fourniture de services	7	8	10	24				
	9	Obtenir de l'information fiable et utile pour prendre des décisions stratégiques	2	4	12	20	26			
<i>Perspective interne</i>	10	Améliorer et maintenir à niveau le fonctionnement des processus métiers	6	7	11					
	11	Abaisser les coûts des processus	7	8	13	15	24			
	12	Assurer la conformité aux lois, réglementations et contrats externes	2	19	20	21	22	26	27	
	13	Assurer la conformité aux politiques internes	2	13						
	14	Gérer les changements métiers	1	5	6	11	28			
	15	Améliorer et maintenir la productivité opérationnelle et celle du personnel	7	8	11	13				
<i>Perspective apprentissage et croissance</i>	16	Gérer l'innovation produit et métiers	5	11	28					
	17	Se procurer et conserver un personnel compétent et motivé	9							

Annexe II : Questionnaire de la Gouvernance des systèmes d'information

Annexe III : Mise en œuvre de « ITGovernanceOntolgy » sur Protégé

- Vue globale de l'ontologie via OntoGraf :

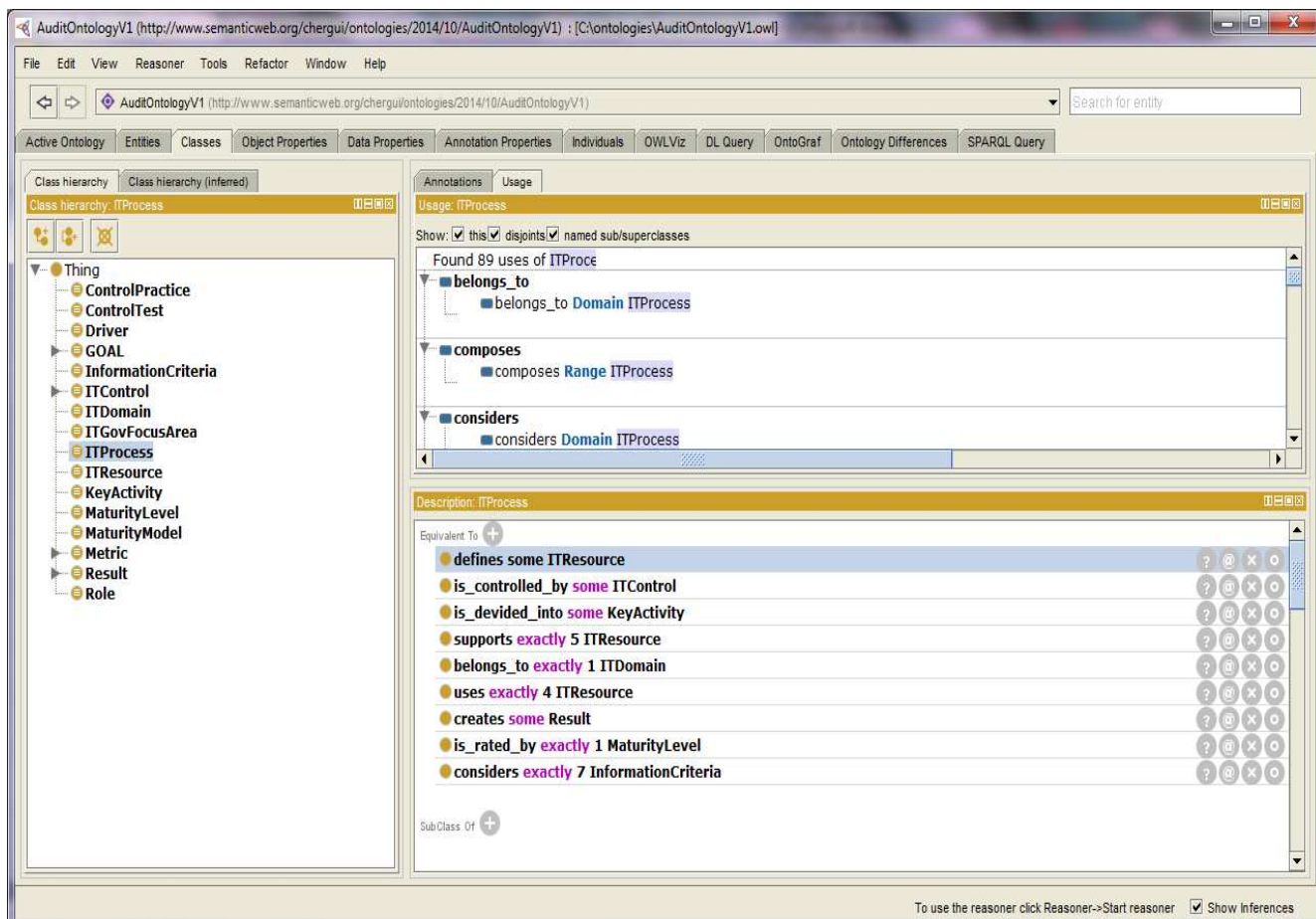
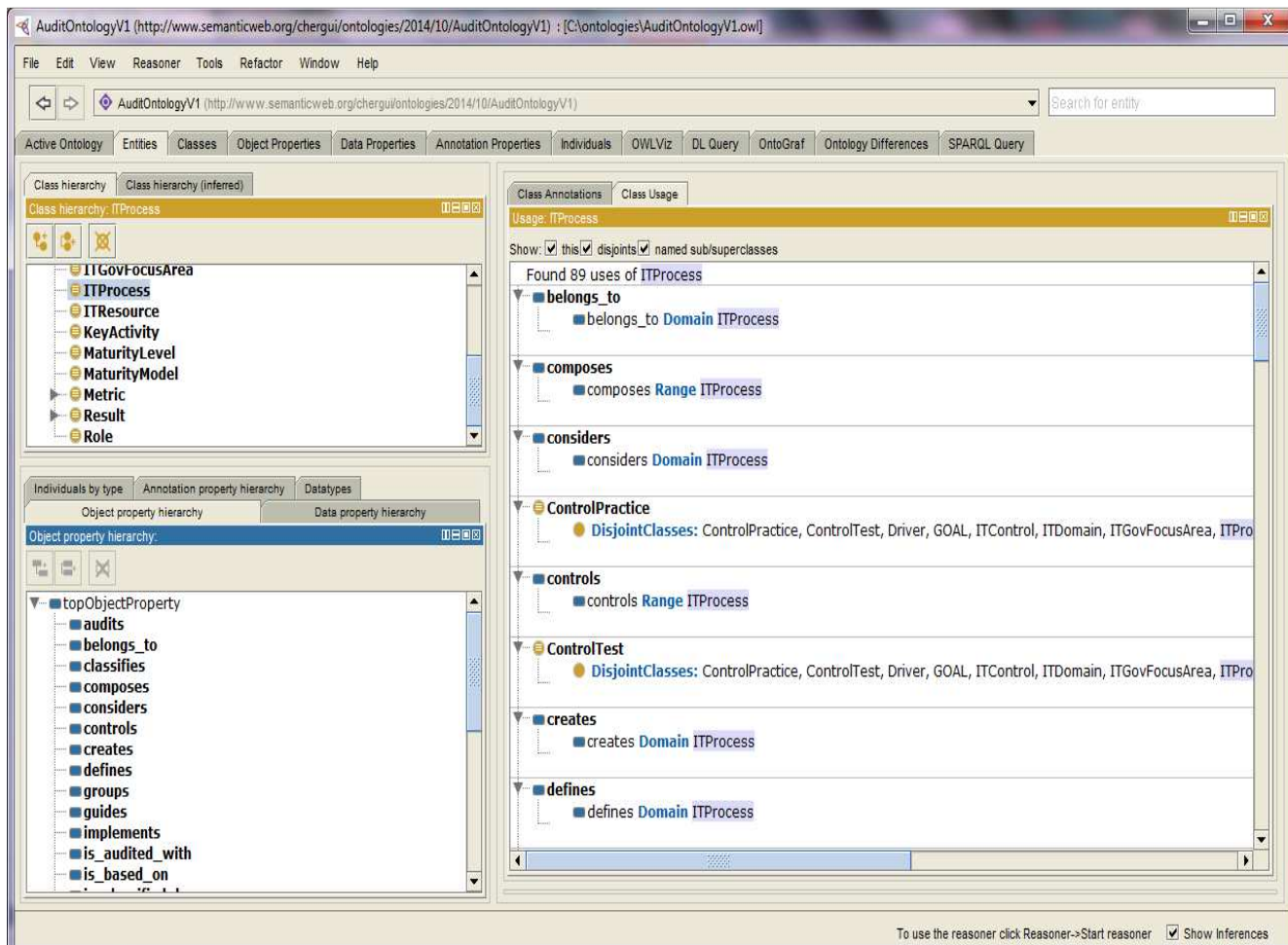




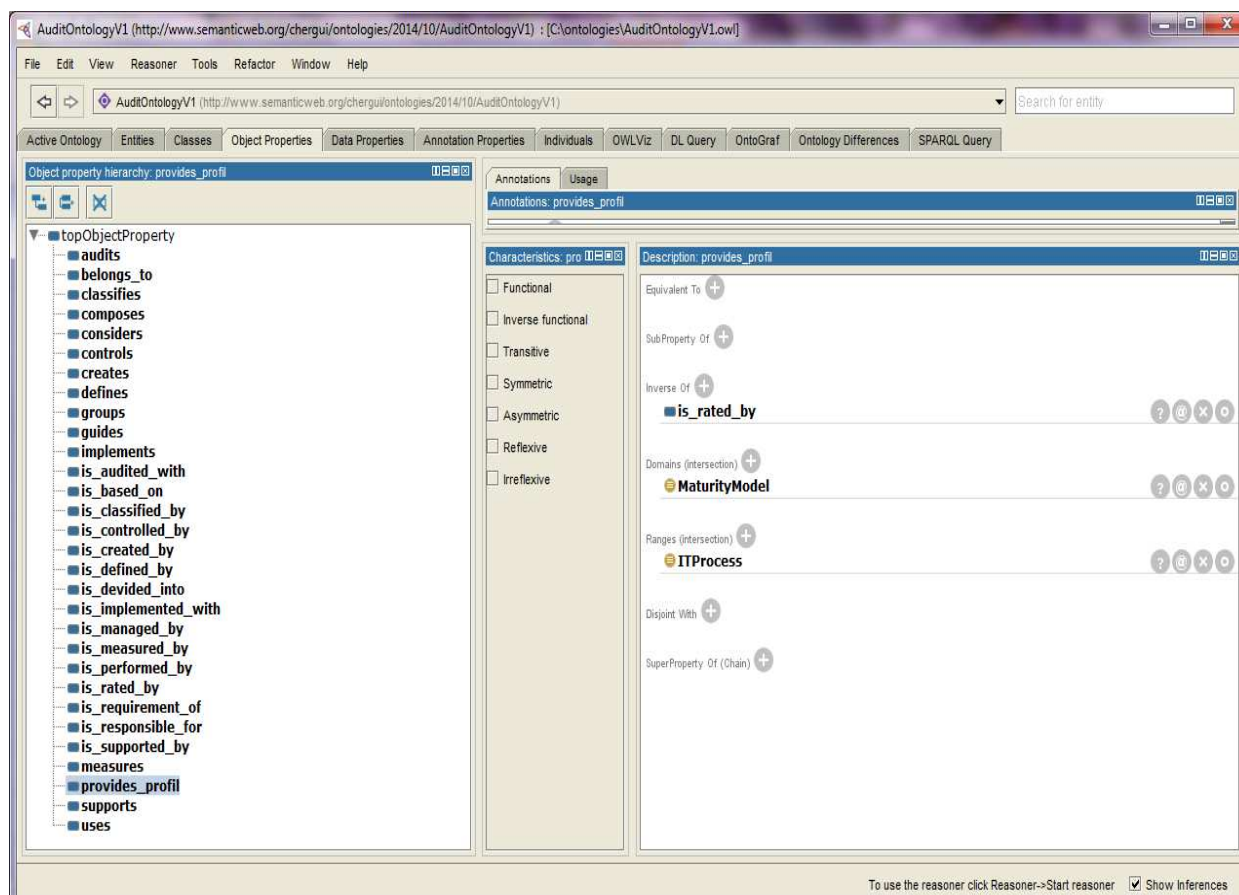
Définition des entités :

The screenshot displays the AuditOntologyV1 interface with the 'Active Ontology' tab selected. The left sidebar shows the 'Class hierarchy' for 'ITControl', listing subclasses: InformationCriteria, ITControl, ITDomain, ITGovFocusArea, ITProcess, ITResource, KeyActivity, and MaturityLevel. The right pane shows the 'Description: ITControl' with the following properties:

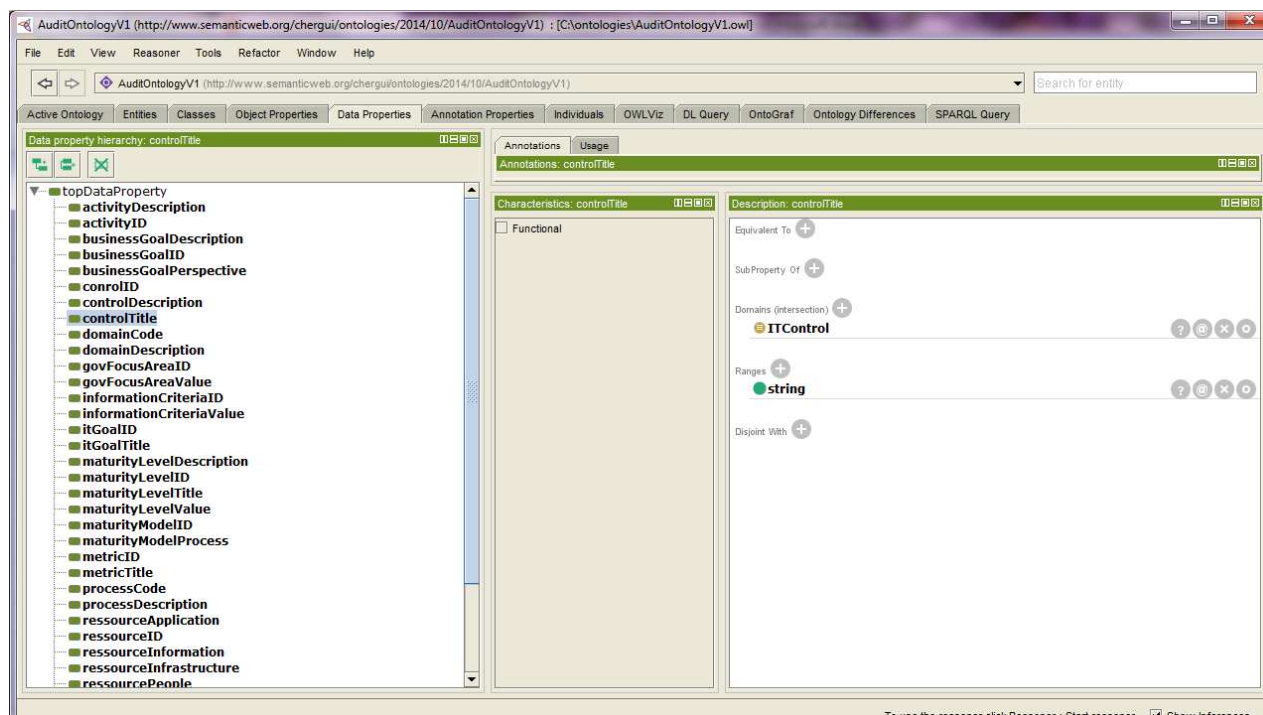
- Equivalent To:
 - is_audited_with some ControlTest
 - controls exactly 1 ITProcess
 - implements some ControlPractice
- Sub-Class Of:
 - (Anonymous Ancestor)
- Members: (empty)
- Target for Key: (empty)
- Disjoint With:
 - Driver, ControlTest, Result, Metric, ITResource, ControlPractice, ITProcess, ITGovFocusArea, ITDomain, Role, GOAL, KeyActivity, InformationCriteria, MaturityModel, MaturityLevel
- Disjoint Union Of: (empty)



Propriétés des objets :



Propriétés des données :



Annexe IV : Publications effectuées dans le cadre de la thèse

Articles dans des revues indexées internationales :

- M Chergui, A Sayouti and H Medromi. Article: **“IT Governance through an Inter-Organizational Workflow based on Multi-Agent System”**. *International Journal of Applied Information Systems (IJAIS)* 6(6):10-16, December 2013. Published by Foundation of Computer Science, New York, USA
- Meriyem Chergui, Hicham Medromi and Adil Sayouti, **“Inter-organizational Workflow for Intelligent Audit of Information Technologies in terms of Enterprise Business Processes”** *International Journal of Advanced Computer Science and Applications(IJACSA)*, 5(5), May 2014.
- Chergui Meriyem, Sayouti Adil, Medromi Hicham. **“IT Governance Ontology Building Process: Example of developing Audit Ontology”**. *International Journal of Computer Techniques (IJCT)* V2 (1): Page (134-141) Jan 2015. ISSN: 2394-2231. www.ijctjournal.org. Published by International Research Group- IRG
- Chergui, M., Sayouti, A., & MEDROMI, H. **“MULTI-AGENT PLATEFORME FOR COBIT IMPLEMENTATION”**. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY (IJESRT)* Feb 2015.
- M.Chergui, H.Nahla, M., Chakir, S.Elhassnaoui, Y.Sekhara & MEDROMI **“ Empirical Study: Moroccan Information systems specificities for better IT Governance ”***International Journal of Advanced Engineering, Management and Science (IJAEMS)* [Vol-2, Issue-5, May- 2016] Infogain Publication (Infogainpublication.com) ISSN : 2454-1311.
- A.CHAKIR, H.NAHLA, M.CHERGUI, S.ELHASSNAOUI, Y.SEKHARA & H.MEDROMI **“An Empirical Study on the information systems in the Moroccan organizations: An explanatory model to decide differently and to optimize the IT governance”**, *International Journal of Advanced Engineering, Management and Science (IJAEMS)* [Vol-2, Issue-4, May- 2016] Infogain Publication (Infogainpublication.com) ISSN: 2454-1311.
- ELHASNAOUI, S., CHAKIR, A., CHERGUI, M., IGUER, H., FARIS, S., & MEDROMI, H. **“Building an integrated IT GRC platform based on multi**

agent system” on International Journal of Engineering and Innovative Technology” (IJEIT). Issue 4, Vol 8 Feb 2015.

- Chergui, M., Chakir, A Sayouti, A., & MEDROMI “**Strategic IT Governance Platform Based on Matchmaking Multi-Agent System and Loose Inter-Organizational Workflows**” on International Journal of Computer Science and Information (IJCSIS) IJCSIS May 2016 Volume 14 No. 5 ISSN 1947 5500
- Aziza CHAKIR, Meriyem Chergui, Soukaina Elhasnaoui, Hicham MEDROMI, Adil SAYOUTI “**Intelligent Plateform to Select The Best IT-GRC Framework For Treatment The Needs Of Stakeholders**” on International Journal of Applied Engineering Research (IJAER) July 2016 Volume 11, Number 12 (2016) pp 7829-7835.
- Y.Sekhara, H.Nahla, S.Elhasnaoui, M.Chergui, A.Chakir, H.Medromi “**Empirical Study on the Status of Moroccan Information Systems and Proposition of Approach for Choice of Best Practices for Good IT Governance**” International Journal of Advanced Engineering, Management and Science (IJAEMS) [Vol-2, Issue-7, July- 2016] Infogain Publication (Infogainpublication.com) ISSN : 2454-1311.
- Soukaina ELHASNAOUI, Meriyem CHERGUI, Aziza CHAKIR, Youssef SEKHARA, Hamid Nahla, Hicham Medromi ”**Empirical Study on the Interaction and Workflow Management between Information System and Business Departments of an Organization to Integrate IT GRC Processes: Case of Moroccan organizations**” on International Journal of Engineering Research and Management (IJERM) Volume 03 Issue 05 (May 2016).
- Chergui, M., Chakir, A., Medromi, H., & Radoui, M. (2017). **A New Approach for Modeling Strategic IT Governance Workflow**. In *Advances in Ubiquitous Networking 2* (pp. 285-298). Springer Singapore.
- Chakir, A., Chergui, M., Elhasnaou, S., Medromi, H., & Sayouti, A. (2016). **A decision approach to select the best framework to treat an IT problem by using multi-agent system and expert systems**. In *Advances in Ubiquitous Networking* (pp. 499-511). Springer Singapore.
- Hamid NAHLA, Aziza CHAKIR, Youssef SEKHARA, Meriyem CHERGUI, Soukaina ELHASNAOUI, Safia LAMRANI and Hicham MEDROMI “**ANALYSIS OF PHYSIONOMY OF IS FUNCTION IN ORGANIZATIONS BY SIZE AND SECTOR OF ACTIVITY**” International Journal of Computer Engineering & Technology (IJCET) Volume 7, Issue 4, July–Aug 2016, pp. 25–40, Article ID: IJCET_07_04_003.

- Hamid Nahla, Hicham Medromi, Meriyem Chergui, Soukaina Elhasnaoui, Aziza Chakir And Youssef Sekhara : "**Analysis of interactions between Information system, Communication and Marketing in Organizations**" International Journal of Advanced Engineering , Management and Science" (Impact Factor : 3.661 - Thomson Reuter 2016), Paper Id : IJAEMS-AUG-2016-1, Vol-2, Issue-8, Aug-2016.
- Chergui Meriyem Chakir Aziza Hicham Medromi Adil Sayouti:” **Intelligent IT Governance Platform: Strategic level**”. International Journal of Advanced Engineering Research and Science Vol-3,Issue-12,December 2016

Communications dans des congrès nationaux et internationaux :

- M.CHERGUI, A.SAYOUTI, H.MEDROMI, Modélisation multi-agent d’un workflow de gouvernance IT , **JDETIC’12**
- M.CHERGUI, A.SAYOUTI, H.MEDROMI, “Multi-agents Modeling of an IT Governance Workflow”. The 2013 International Conference on Intelligent Information and Network Technology (**IC2IN’13**) FST Settat Morocco November 13th -14th, 2013.
- Chergui, M., Medromi, H., & Sayouti, A. (2014, May). **Intelligent audit of information systems by inter-organizational workflow**. In Next Generation Networks and Services (**NGNS’14**), 28-30 May, 2014 Fifth International Conference on (pp. 1-6). **IEEE**.
- Chakir, A., Chergui, S., Medromi, H., & Sayouti, A “**An approach to select effectively the best framework IT according to the axes of the governance IT, to handle and to set up an objective IT**” The Third World Conference on Complex Systems (**WCCS’15**) November 2015. **IEEE**.
- Chergui, M., Chakir, A Sayouti, A., & H.MEDROMI “**Matchmaking Expert system based on Intelligent Agents and semantic inference engine: Application to IT Governance Context**” International Conference on Advanced Information Technology, Services and Systems (**AIT2S’15**) FST Settat Morocco 16-17 December 2015.
- Chakir, A., Chergui, M., Elhasnaoui, S., Medromi, H., & Sayouti, A. “**A decision approach to select the best framework to treat an IT problem by using multi-agent system and expert systems**”. The International Symposium On Ubiquitous Networking (**UNET’15**). September 08-10, 2015 CASABLANCA MOROCCO.
-

- Chergui, M., Chakir, A., Medromi, H., & Radoui, M. « **A new approach of modelling strategic IT Governance workflow**” In The International Symposium On Ubiquitous Networking (UNET’16). May 30- June 1st, 2016.

- S.ELHASNAOUI, A.CHAKIR, M.CHERGUI, H.MEDROMI and A.SAYOUTI “**A multi agent architecture for an IT GRC Platform using an intelligent communication system**”, 4^{ème} édition de la Journée Doctorale des Sciences de l’Ingénieur (JDSI’2017), Casablanca, Morocco.

- A.CHAKIR, M.CHERGUI, S.ELHASNAOU, H.MEDROMI and A.SAYOUTI “**Intelligent IT Governance Platform: Decision making layer**”, 4^{ème} édition de la Journée Doctorale des Sciences de l’Ingénieur (JDSI’2017), Casablanca, Morocco.