

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITE MOULOU MAMMERI DE TIZI-OUZOU

FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE
DEPARTEMENT INFORMATIQUE



*Mémoire de fin d'études en vue de l'obtention du
Diplôme de Master en Informatique*
Option : Conduite de Projets Informatiques

THEME :

**Test de mise en œuvre d'une ontologie de processus à
l'aide de Protégé**

Présenté par :

M^{elle} MERRAD Lilia

M^{elle} OULD LAMARA Fadoua

Dirigé par :

M^{me} SINI Ghanima

2014-2015



REMERCIEMENTS

C'est avec l'aide d'Allah tout puissant que ce modeste mémoire a pu être réalisé, Dieu qui nous a donnée foi, raison et lucidité.

On porte ce témoignage écrit de notre reconnaissance à tous ceux qui nous ont gratifiés de leur soutien et de leur confiance tout le long de ce travail.

Nous exprimons notre profonde gratitude à notre promotrice Madame Sini qui nous a fait l'honneur de diriger ce travail. Nous la remercions pour sa disponibilité et ses conseils qui nous ont été d'un apport considérable.

Nos vifs remerciements vont également aux membres du jury, pour l'intérêt qu'ils ont apporté à notre travail en acceptant de l'examiner et de l'enrichir par leurs propositions.

Dédicace



A mes chers parents,

A ma sœur,

A mon frère,

A toute la famille Merrad et Belaidi

A notre petite famille du « labo 8 »

A mes amis(es)

Je dédie ce travail.

Merrad Lilia



A mon père,

A ma mère,

Aux deux être les plus chers au monde

*Sans eux je n'aurai jamais été ce que je suis
aujourd'hui*

A mes chers frères (Smaïl, Ahcene, Amine)

A mes chères tantes & chers oncles,

A mes chères cousines & chers cousins,

A ma famille en Or « les fous du labo 8 »

A tous ceux qui me sont chers

Je dédie ce travail.

Ould lamara Fadoua

Table des matières

Remerciements	
Dédicace	
Sommaire	
Liste des figures	
Liste des tableaux	
Acronymes	
Résumé	
Introduction Générale.....	01
<div style="border: 1px dashed green; padding: 10px; display: inline-block;"><h2><i>Chapitre I : les ontologies</i></h2></div>	
Introduction.....	04
I.1. Origine & Définition.....	04
I.2. Cycle de vie.....	06
I.3. Les types d'ontologie.....	08
I.3.1. Selon l'objet de conceptualisation.....	08
I.3.1.1. Ontologie de représentation de connaissance.....	08
I.3.1.2. Ontologie de haut niveau/supérieure.....	09
I.3.1.3. Ontologie Générique.....	09
I.3.1.4. Ontologie du domaine.....	09
I.3.1.5. Ontologie de Tâches.....	09
I.3.1.6. Ontologie d'application.....	09
I.3.2. Selon le degré de formalisme de la représentation.....	10
I.3.2.1. Hautement informelle.....	10
I.3.2.2. Semi-informelle.....	10

I.3.2.3. Semi-formelle.....	10
I.3.2.4. Rigoureusement formelle.....	10
I.3.3. Selon le niveau de détail.....	10
I.3.3.1. La granularité fine.....	10
I.3.3.2. La granularité large.....	10
I.3.4. Selon le niveau de complétude.....	10
I.3.4.1. Niveau Sémantique.....	11
I.3.4.2. Niveau Référentiel.....	11
I.3.4.3. Niveau Opérationnel.....	11
I.4. Composantes d'une ontologie.....	11
I.5. Les méthodes de construction d'ontologie.....	14
I.5.1. La méthode ON-TO- KNOWLEDGE	14
I.5.2. La méthode METHONTOLOGY.....	15
I.5.3. La méthode TERMINAE.....	16
I.6. Les outils de développement d'ontologie.....	16
I.6. 1. Les outils dépendants de formalisme de représentation.....	16
I.6.2. Les outils indépendants de formalisme de représentation.....	18
I.7. Domaines d'applications des ontologies.....	18
I.7.1. Système d'information.....	18
I.7.2. Web sémantique.....	19
Conclusion.....	20

Chapitre II : Présentation de l'ontologie à mettre en œuvre

II.1. Présentation de l'ontologie.....	22
II.2. Définition des composants de l'ontologie.....	23
II.3. Expressivité de la modélisation	24
II.4. Les règles de gestion respectées par l'ontologie.....	25
II.5. Interrogation de l'ontologie.....	26
II.5.1. Interrogation informationnelle (donnée).....	26
II.5.2. Interrogation organisationnelle (rôle).....	26
II.5.3. Interrogation fonctionnelle (tâche, processus).....	26
II.6. Conceptualisation de l'ontologie.....	26
II.6.1. Glossaire des termes de l'ontologie.....	26
II.6.2. Tableau des relations binaires.....	27
II.6.3. Diagrammes de relations binaires.....	28
II.6.4. Dictionnaire des concepts.....	29
II.6.5. Classification des concepts en hiérarchie.....	29
Conclusion.....	30

Chapitre III : Les Processus

Introduction	32
III.1. Processus.....	32
III.1.1. Définition d'un processus.....	32
III.1.2. Caractéristique des processus.....	33
III.1.3. Type de processus.....	33
III.2. Processus métier.....	34
III.2.1. définition du processus métier.....	34
III.2.2. Quand et pourquoi faire ?.....	34

III.2.3. Typologie des processus métiers.....	34
III.2.4. La gestion de processus métier (BPM).....	34
III.2.5. Modélisation des processus métiers.....	35
III.2.5.1. Définition de la modélisation.....	35
III.2.5.2. Définition d'un modèle.....	35
III.2.5.3. Pourquoi modéliser les processus métier ?.....	35
III.2.5.4. Buts d'une démarche de modélisation.....	36
III.3. Présentation du processus réel.....	36
III.4. Conceptualisation de l'ontologie.....	40
III.4.1. Glossaire des termes.....	41
III.5. Formalisation.....	43
III.5.1. Construction de la ABox.....	43
III.5.2. Construction de la TBox.....	43
Conclusion.....	44

Chapitre IV : Test et implémentation

Introduction.....	46
IV.1. Présentation de protégé.....	46
IV.2. Les langages de représentation d'ontologie sous protégé.....	46
IV.2.1. XML.....	47
IV.2.2. RDF.....	47
IV.2.3. RDFS.....	48
IV.2.4. DAML-OIL.....	48
IV.2.5. OWL.....	49

IV.3. Les moteurs d'inférences.....	50
IV.3.1. Pellet.....	50
IV.3.2. Racer.....	51
IV.4. Langage d'interrogation (SPARQL).....	51
IV.5. Logique de description.....	52
IV.6. Implémentation de l'ontologie.....	52
IV.7. Visualisation de l'ontologie.....	56
IV.8. Interrogation de l'ontologie.....	58
Conclusion.....	63
Conclusion Générale.....	65
Références bibliographiques	

Liste des figures

Figure I.1: Le cycle de vie d'une ontologie.

Figure I.2: Les composantes d'une ontologie.

Figure I.3: Le triangle sémantique.

Figure I.4: Architecture du web sémantique.

Figure II.1: Ontologies des concepts retenus pour la description d'un processus métier.

Figure II.2: Diagramme des relations binaires de notre ontologie.

Figure II.3: Taxonomie des concepts.

Figure IV.1: Définition de la hiérarchie.

Figure IV.2: Définition des hiérarchies des concepts de l'ontologie.

Figure IV.3: Définition de l'attribut « Nom_organisation » du concept Organisation.

Figure IV.4: Définition des relations.

Figure IV.5: Définition des restrictions.

Figure IV.6: Visualisation d'un extrait de l'ontologie avec OntoGraf.

Figure IV.7: Visualisation de l'instance « service_cardiologie » avec OntoGraf.

Figure IV.8: Visualisation d'une partie de l'ontologie avec OWLViz.

Figure IV.9: Visualisation de l'ontologie complète avec OWLViz.

Figure IV.10: Exemple 1 de l'exécution d'une requête SPARQL sur Protégé.

Figure IV.11: Exemple 2 de l'exécution d'une requête SPARQL sur Protégé.

Figure IV.12: Exemple 3 de l'exécution d'une requête SPARQL sur Protégé.

Figure IV.13: Exemple 4 de l'exécution d'une requête SPARQL sur Protégé.

Figure IV.14: Exemple 5 de l'exécution d'une requête SPARQL sur Protégé.

Figure IV.15: Exemple 6 de l'exécution d'une requête SPARQL sur Protégé.

Figure IV.16: Exemple 7 de l'exécution d'une requête SPARQL sur Protégé.

Liste des tableaux

Tableau II.1: Glossaire des termes.

Tableau II.2: Les relations binaires.

Tableau II.3: Dictionnaires des concepts.

Tableau III.1 : Instanciation de l'ontologie.

Tableau III.2 : Glossaire des termes.

Tableau III.3 : Tableau des relations binaires.

Tableau III.4 : Assertion sur les individus.

Tableau III.5 : Assertion des relations.

Tableau III.6 : Définition de la Tbox.

Tableau III.7 : Définition des axiomes.

Acronymes

Abox: assertional box

BPM: Business Process Management

CHU: centre hospitalo-universitaire

GUI: graphic user interface

HTML: Hypertext Markup Language

IA: intelligence artificielle

IRI: Internationalized Resource Identifiers

LD: logique de description

nRQL: new Racerpro query Language

ODE: Ontology Design Environment

OilEd: Oil Editor

OTK: ON-TO-KNOWLEDGE

OWL: Web Ontology Language

RACER: Renamed ABox and Concept Expression Reasoner

RDF: Resource Description Framework

RDF-S: Resource Description Framework Schema

SPARQL: Simple Protocol And Rdf Query Language

SWRL: Semantic Web Rule Language

Tbox: Terminology Box

URI: Uniform Resource Identifier

W3C: World Wide Web Consortium

WFMC: Workflow Management Coalition

Résumé

Les ontologies, définies par Gruber [3] comme « *Une spécification explicite d'une conceptualisation* », se sont imposées comme un moyen prometteur pour résoudre les problèmes de compréhension de données et permettre le partage de connaissances. Ces connaissances qui sont collectées par les concepteurs peuvent être incomplètes, insuffisantes voire même incohérentes ce qui rend difficile la compréhension du processus à mettre en œuvre. Cependant, pour un meilleur échange de l'information, toutes les personnes et les organisations devraient utiliser le même vocabulaire.

Notre travail porte sur l'instanciation de l'ontologie proposée par madame Sini[22] avec un processus métier d'un cas réel, afin de vérifier si cette dernière est suffisante pour la description de n'importe quel processus organisationnel en utilisant l'outil protégé.

Mots clés : Ontologie, Processus, Instanciation, Protégé.

Abstract

Ontology defined by Gruber as “an explicit specification of a conceptualization”, is in a good progress to become the most relevant means to solve data understanding problems and allows knowledge sharing. The collected data through software developers may be incomplete, insufficient, as well as incoherent which do not facilitate the understanding of the process to be adopted. However, in order to ensure a better exchange of information, people and organizations have to use the same vocabulary.

The current study deals with Sini's instantiation of ontology applied to a case study. It seeks to examine whether it is appropriate to the description of any given organizational process, by using Protégé.

Keywords: Ontology, Process, instantiation, Protégé

*Introduction
Générale*

Introduction Générale

« *Seul celui qui agit apprend* »

Friedrich Nietzsche

L'être humain ne cesse de chercher à comprendre le monde qui l'entoure. Les différentes phases d'étude nécessaires à cette compréhension ont toujours évolué relativement avec la technologie : observation, constatation, hypothèse, preuve, explication, acquisition et modélisation. Ces dernières années des progrès considérables ont été observés dans le développement des méthodes de modélisation des données et dans les technologies de recherche de l'information afin de manipuler la diversité des connaissances et leur ampleur.

Et avec le développement croissant de ces connaissances et la complexité de leur gestion au sein de l'entreprise, une meilleure technique et langage de représentation des connaissances doivent être présentés. Pour pouvoir représenter des connaissances propres à un domaine particulier, on doit décrire et coder les entités de ce domaine sous une forme qui puisse être interprétée et exploitée par l'homme et par la machine. L'avènement de l'informatique et l'apparition de nouveaux formalismes de traitement nous ont permis d'automatiser plusieurs tâches pénibles et fastidieuses et de ne plus faire manipuler en aveugle des informations à la machine mais de permettre un dialogue et une coopération entre le système et les utilisateurs. Pour cela, le système doit avoir accès non seulement aux termes utilisés par l'être humain mais également à la sémantique qui lui est associée, afin qu'une communication efficace soit possible.

La notion d'ontologie est une solution candidate qui constitue un enjeu dans la représentation et la modélisation des connaissances. Elle sert à représenter de façon générique et réutilisable la sémantique d'un domaine. A l'origine, elle fournit pour un domaine particulier une compréhension commune pour partager les connaissances et les données.

Le travail présenté dans ce mémoire concerne la gestion d'un organisme, et plus particulièrement la mise en œuvre d'une ontologie de processus pour un organisme de santé publique. Après une étape de conceptualisation qui a permis l'identification des connaissances du domaine, une étape d'ontologisation a permis de formaliser les connaissances identifiées (sous forme d'une ontologie) en utilisant un modèle sémantique, une dernière étape appelée opérationnalisation a permis la transcription de l'ontologie dans un langage formel et opérationnel de représentation de connaissances ; pour l'opérationnalisation de l'ontologie,

l'éditeur d'ontologie Protégé (version 4.3) est utilisé pour représenter l'ontologie sur machine.

Notre contribution dans ce contexte consiste en un test de mise en œuvre d'une ontologie de domaine proposée dans les travaux de Mme Sini [22].

L'objectif de notre travail est de tester l'ontologie proposée afin de voir si le modèle est suffisant pour décrire un processus en utilisant un cas réel sous l'éditeur d'ontologie Protégé.

Pour ce faire notre mémoire s'organise comme suit :

Dans le premier chapitre de ce mémoire, nous présenterons les ontologies, nous expliquerons leur cycle de vie, les outils et les méthodes pour leur construction ainsi que leur domaine d'utilisation.

Dans le second chapitre, nous allons présenter et expliquer l'ontologie développée par madame Sini [22] puis on passera à sa conceptualisation.

Dans le troisième chapitre on parlera des différents processus et on se focalisera sur les processus métier et leur modélisation, on présentera notre cas réel et son instanciation.

Dans le quatrième chapitre on passera à l'édition sous protégé de notre ontologie qu'on va tester et inférer. Nous terminerons notre travail par une conclusion et des perspectives.

Chapitre I

Introduction

Aujourd'hui une grande banque de connaissances est distribuée et partagée sur toute la planète grâce à Internet. Pour pouvoir les manipuler les chercheurs ont développé des outils permettant un accès efficace à ces informations, à travers des interfaces automatisées permettant à l'utilisateur de formuler son besoin informationnel.

Pour faciliter cette recherche et la compréhension des ressources, le système doit avoir accès non seulement aux termes utilisés par l'être humain mais aussi à la sémantique qui leur est associée, afin qu'une communication efficace soit possible.

Actuellement, la connaissance visée par les ontologies est un sujet de recherche populaire dans diverses communautés citant l'ingénierie des connaissances, la recherche d'information, les systèmes d'information coopératif et l'e-commerce et sont au cœur du Web sémantique. Elles ont été mises au point dans le but de partager et de réutiliser les connaissances ainsi que les raisonnements de différents secteurs d'activité et de production.

Le présent chapitre a pour objectif de répondre à d'éventuelles questions: Qu'est-ce qu'une ontologie? Quels sont les principes à suivre pour construire une ontologie? Quels sont les composants d'une ontologie? Quels types d'ontologies existent déjà ? Les outils logiciels utilisés ?

En vue de répondre à ces questions, le chapitre est constitué comme suit: Présentation de l'ontologie, son cycle de vie ; les types et composants d'une ontologie, les méthodes et outils de construction de l'ontologie et enfin leur domaines d'applications.

I.1. Les origines et définitions

Introduit en intelligence artificielle(IA) il y'a 23 ans, le terme ontologie est utilisé en philosophie depuis XIX siècle. Dans ce domaine, l'ontologie est un terme grec composé des mots «*ontos*» qui veut dire ce qui existe, l'être, l'existant, et «*logos*» qui veut dire l'étude, le discours, d'où sa traduction par « étude des propriétés générales de ce qui existe ».[1]

Les ontologies informatiques sont des outils qui permettent précisément de représenter un corpus de connaissances sous une forme utilisable par une machine.

En informatique, la littérature fournit un tas de définitions du mot ontologie. Ces définitions, dans leur diversité, offrent des points de vues à la fois différents et complémentaires. Dans la section suivante, nous décrivons quelques unes.

En 1991, Neeches et ses collègues furent les premiers à proposer une définition [2]

« Une ontologie définit les termes et les relations de base du vocabulaire d'un domaine ainsi que les règles qui indiquent comment combiner les termes et les relations de façon à pouvoir étendre le vocabulaire »

Cette définition descriptive donne un premier aperçu sur la manière de construire une ontologie, à savoir l'identification des termes de bases d'un domaine et les relations entre ces termes ainsi que les règles pouvant s'appliquer sur ces derniers.

En 1993, Gruber propose la définition qui est jusqu'à présent la plus célèbre[3]

«Une ontologie est une spécification explicite d'une conceptualisation»

Il a introduit la notion de 'conceptualisation' qui réfère à un modèle abstrait d'un certain domaine du monde réel en identifiant les concepts pertinents décrivant ce domaine. Le terme 'explicite' signifie que les concepts utilisés ainsi que les contraintes sur leur emploi, sont réellement définis d'une manière claire et précise.

Cette définition a été modifiée légèrement par Borst[4] comme étant

«une ontologie est une spécification formelle d'une conceptualisation partagée».

Cette définition précise d'une part, le fait que l'ontologie doit être 'formelle', c'est à dire exprimée sous forme d'une logique pouvant être exploitable par une machine. D'autre part, elle doit être 'partagée' dans la mesure où elle doit capturer des connaissances partagées entre différents individus.

En 1998, Studer et ses collègues ont rassemblé ces deux définitions (celles de Gruber et Borst) dans une seule qui est :

« une ontologie est une spécification formelle et explicite d'une conceptualisation partagée »

Ils l'expliquent comme suit :

- ✓ Spécification explicite signifie que les concepts, les propriétés, les relations, les fonctions, les restrictions et les axiomes de l'ontologie sont définis de façon déclarative ;
- ✓ Formelle : l'ontologie doit être lisible par une machine, ce qui exclut le langage naturel.
- ✓ Conceptualisation le modèle abstrait d'un phénomène du monde réel par identification des concepts clefs de ce phénomène.

- ✓ Partagée : l'ontologie n'est pas la propriété d'un individu, mais elle représente un consensus accepté par une communauté d'utilisateurs.

En 1995, Guarino et Giaretti proposent leur définition :

« Une ontologie est une théorie logique proposant une vue explicite et partielle d'une conceptualisation »

Une ontologie est dite partielle dans le sens où une conceptualisation ne peut pas toujours être entièrement formalisée dans un tel cadre, du fait d'ambiguïtés ou du fait qu'aucune représentation de leur sémantique n'existe dans le langage de représentation choisi.

Selon Gomez et ses collègues [12],

« Une ontologie fournit les moyens de décrire de façon explicite la conceptualisation des connaissances représentées dans une base de connaissances »

les ontologies visent à capturer les connaissances consensuelles de façon générique ainsi que la façon de leur réutilisation et leur partage à travers des applications et des groupes de personnes.

I.2. Cycle de vie

Un cycle de vie d'une ontologie débute du besoin qui permet notamment de cerner précisément le domaine de connaissance, qui se transforme en idée, la concrétisation de l'idée qui se traduit par la conception qui est diffusée pour son utilisation. Vient ensuite l'étape de l'évaluation qui donne naissance, le plus souvent à une étape d'évolution, et de maintenance du modèle. Une réévaluation de l'ontologie et des besoins devra se faire après chaque utilisation significative. [5]

Fernandez et ses collègues insistent sur le fait que les activités de documentation et d'évaluation sont nécessaires à chaque étape du processus de construction.

L'évaluation précoce permettant de limiter la propagation d'erreurs. Le processus de construction peut être intégré au cycle de vie d'une ontologie.

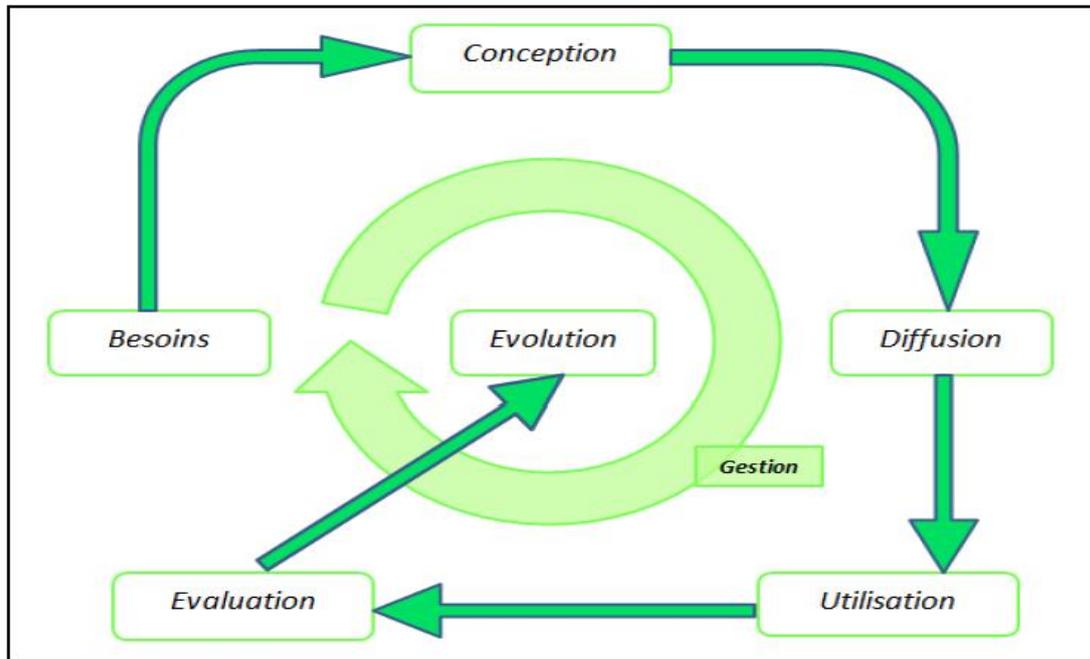


Figure I.1 : Le cycle de vie d'une ontologie

✚ Besoins et évaluation

L'activité de détection des besoins, préalable à la conception, et l'activité d'évaluation, lorsqu'une ontologie est utilisée, posent des problèmes méthodologiques de recueil (analyse d'entretiens, questionnaires et sondages, étude de l'ergonomie et des usages) et d'identification. En complément, la phase de détection des besoins demande un état des lieux initial approfondi, car elle ne peut reposer sur des études précédentes ou des retours d'utilisation, comme c'est le cas pour l'évaluation.

✚ Conception et évolution

Les phases de conception initiale et d'évolution ont en commun un certain nombre de points :

- spécification des solutions
- conceptualisation et modélisation
- formalisation (logiques de description, graphes conceptuels, formalismes du web sémantique RDF, RDF(S) et OWL)
- intégration de ressources existantes
- implantation (graphes conceptuels, logiques de description)

Le choix de représentation et de conceptualisation faits dans l'ontologie représente un problème de conception et d'évolution. Notons aussi que l'évolution pose le problème de la maintenance de ce qui repose déjà sur l'ontologie. L'ontologie est à la fois un ensemble évolutif et un ensemble de primitives pour décrire des faits et des algorithmes sur ces faits. Ses changements donc, ont un impact direct sur tout ce qui a

été construit sur la base de cette ontologie. Le maintien de la cohérence dans une ontologie est un des points clés dans son utilisation la maintenance de l'ontologie soulève donc des problèmes d'intégration technique et des problèmes d'intégration aux usages.

Diffusion

Cette phase s'intéresse au déploiement et mise en place de l'ontologie particulièrement dans le cas de solutions distribuées mais aussi du point de vue de l'information des utilisateurs sur la mise à disposition ou mise à jour de l'ontologie.

Utilisation

Les activités reposant sur la disponibilité de l'ontologie, comme l'annotation des ressources, la résolution de requête, la déduction de connaissances. constituent la phase d'utilisation.

Gestion

L'existence d'une activité permanente de gestion et planification est importante pour assurer une pérennité dans le travail de suivi et la politique globale pour la détection, la préparation l'évaluation des itérations d'un cycle et s'assurer que l'ensemble de ces causes améliorent les systèmes d'information.

Loin d'une méthode concrète ou cadre méthodologique pour construire une ontologie, il en ressort que la phase de construction peut être décomposée en 3 étapes: conceptualisation, ontologisation, opérationnalisation.

I.3. Les types d'ontologies

Plusieurs classifications des ontologies ont été proposées dans la littérature dans ce qui suit nous enoncerons quelques une

I.3.1. Selon l'objet de conceptualisation

I.3.1.1. Ontologie pour la représentation de connaissances [6]

Elle capture les primitifs de représentation employées pour formaliser la connaissance, l'exemple le plus représentatif est celui de Gruber 1993 [1], (*the Frame Ontology*) et (*OKBC Open Knowledge Base Connectivity*),ils définissent de manière formelle, les concepts utilisés principalement dans des langages à base de frames : classes, sous classes, attributs, valeurs, relations et axiomes.

I.3.1.2. Les ontologies supérieures (*Upper ou TopAlevel ontologie*) [7]

Ce type d'ontologies vise à étudier les catégories des choses qui existent dans le monde, comme les concepts de haut niveau d'abstraction tels que les entités, les événements, les processus, les actions, le temps, les relations et les propriétés.

I.3.1.3. Ontologie Générique (*Generic ontology*)

Elle est appelée également noyau ontologique, modélise des connaissances moins abstraites que celles véhiculées par l'ontologie de haut niveau mais assez générales néanmoins pour être réutilisées à travers différents domaines.

Borst a cité deux exemples d'ontologies comme suit [4] :

- **l'ontologie méréologique** qui est une branche de l'ontologie formelle et qui est une application de la logique des prédicats qui traite des relations entre la partie et le tout comme par exemple :
Le Toit d'une maison,
Les organes du corps,
Les cellules de l'organisme
- **l'ontologie topologique** tout comme la théorie de méréologie, l'ontologie topologique contient des relations, associé-à.

I.3.1.4. Les ontologies de domaine [8] [6]

C'est des ontologies qui sont construites sur un domaine particulier de la connaissance. Elles fournissent le vocabulaire des concepts du domaine de connaissance et les relations entre ces derniers, les activités de ce domaine ainsi que les théories et les principes de base de ce domaine. Les ontologies de domaine constituent donc des méta-descriptions d'une représentation de connaissances du domaine.

I.1.3.5. Les ontologies de tâche [8] [7]

Ces ontologies sont utilisées pour gérer des tâches spécifiques liées à la résolution de problèmes dans les systèmes, telles que les tâches de diagnostic, de planification, de configuration. Elles fournissent un ensemble de termes au moyen desquels on peut décrire au niveau générique comment résoudre un type de problème.

I.1.3.6. Les ontologies d'application [6]

Ce sont les ontologies les plus spécifiques. Elles permettent de décrire des concepts dépendants à la fois d'un domaine et d'une tâche. Dans cette classification, la notion d'ontologie d'application définit le contexte d'une application qui décrit la sémantique

des informations et des services manipulés par une ou un ensemble d'applications sur un même domaine.

I.3.2. Selon le degré de formalisme de la représentation

Les ontologies peuvent être distinguées en fonction du degré de formalisme utilisé pour les exprimer. Uschold et Grüninger [9] proposent une classification contenant les quatre catégories:

I.3.2.1. Hautement informelle: Elle est exprimée en langue naturelle (sémantique ouverte).

I.3.2.2. Semi-informelle: Exprimée dans un langage naturel structuré et limité en respectant un ensemble de règles permettant d'augmenter la clarté et la lisibilité.

I.3.2.3. Semi-formelle: Elle est exprimée dans un langage artificiel et formel.

I.3.2.4. Rigoureusement formelle: L'ontologie est exprimée dans un langage contenant une sémantique formelle, des théorèmes, et des preuves pour vérifier les propriétés telles que la validité et la complétude.

I.3.3. Selon le niveau de détail

On peut distinguer les ontologies selon le niveau de détail utilisé lors de la conceptualisation de l'ontologie en fonction de l'objectif opérationnel envisagé pour l'ontologie, deux catégories peuvent être identifiées : granularité fine et granularité large. [10]

I.3.3.1. La granularité fine

Elle correspond à une ontologie très détaillée, possédant ainsi un vocabulaire plus riche capable d'assurer une description détaillée des concepts pertinents d'un domaine ou d'une tâche.

I.3.3.2. La granularité large

Elle correspond à un vocabulaire moins détaillé. Les ontologies génériques possèdent une granularité large, compte tenu du fait que les notions sur lesquelles elles portent peuvent être raffinées par des notions plus spécifiques.

I.3.4. Selon le niveau de complétude

Le niveau de complétude a été abordé par Mizoguchi et Bachimont. À titre d'exemple, nous décrivons la typologie de Bachimont. Ce dernier propose la classification sur trois niveaux suivante :

I.3.4.1. Niveau Sémantique

Tous les concepts (caractérisés par un terme/libellé) doivent respecter les quatre principes différentiels :

- 1) Communauté avec l'ancêtre.
- 2) Différence (spécification) par rapport à l'ancêtre.
- 3) Communauté avec les concepts frères (situés au même niveau).
- 4) Différence par rapport aux concepts frères (sinon il n'aurait pas lieu de le définir).

Deux concepts sémantiques sont identiques si l'interprétation du terme/libellé à travers les quatre principes différentiels aboutit à un sens équivalent.

I.3.4.2. Niveau Référentiel

Les concepts référentiels (ou formels) se caractérisent par un terme/libellé dont la sémantique est définie par une extension d'objets. L'engagement ontologique spécifie les objets du domaine qui peuvent être associés au concept, conformément à sa signification formelle. Deux concepts formels seront identiques s'ils possèdent la même extension (ex : les concepts d'étoile du matin et d'étoile du soir associés à Vénus).

I.3.4.3. Niveau Opérationnel

Les concepts du niveau opérationnel ou computationnel sont caractérisés par les opérations qu'il est possible de leur appliquer pour générer des inférences (engagement computationnel). Deux concepts opérationnels sont identiques s'ils possèdent le même potentiel d'inférence.

I.4. Les composantes d'une ontologie

Les ontologies définissent, le sens des termes et les relations entre eux, elles leur fournissent un vocabulaire commun. Bien que les ontologies peuvent être très différentes surtout au niveau du traitement de leurs composantes de base telles que les choses, les relations, elles caractérisent un même univers.

Ces connaissances sont formalisées en mettant en jeu les composants suivants: concepts, relations, fonction, axiomes et instances.

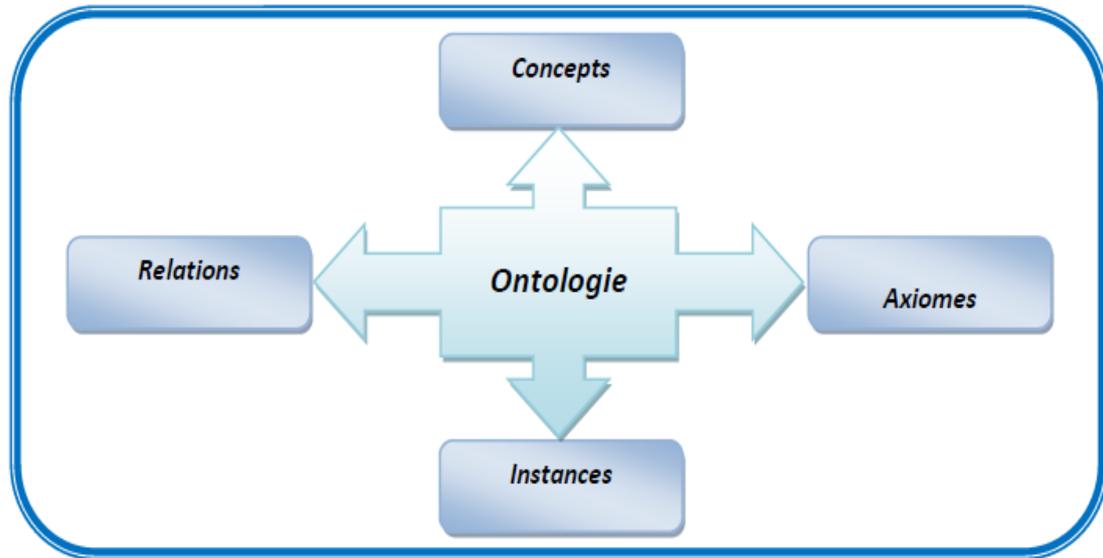


Figure I.2: Les composantes d'une ontologie.

I.4.1. Les concepts

Les concepts appelés termes ou classe de l'ontologie, définissant un ensemble d'objets, abstraits ou concrets, que l'on souhaite modéliser pour un domaine donné. Les connaissances portent sur des objets auxquels on se réfère à travers des concepts. Un concept peut représenter un objet matériel, une notion, une idée [11]. L'ensemble des propriétés d'un concept constitue sa compréhension ou son intension et l'ensemble des êtres qu'il englobe, son extension.

Ces concepts selon Gomez [12] peuvent être classifiés selon trois dimensions:

- Niveau d'abstraction (concrets ou abstraits) ;
- Atomicité (élémentaires ou composés) ;
- Niveau de réalité (réels ou fictifs).

Un concept est composé de trois parties :

- Une notion : elle correspond à la sémantique du concept, elle est définie à travers ses propriétés et ses attributs. Elle est appelée intention du concept;
- Un ensemble d'objets : il correspond aux objets définis par le concept, il est appelé extension du concept. Les objets sont les instances du concept;
- Un (ou plusieurs) terme(s) : les termes permettent de désigner le concept. Ces termes sont aussi appelés labels de concept.

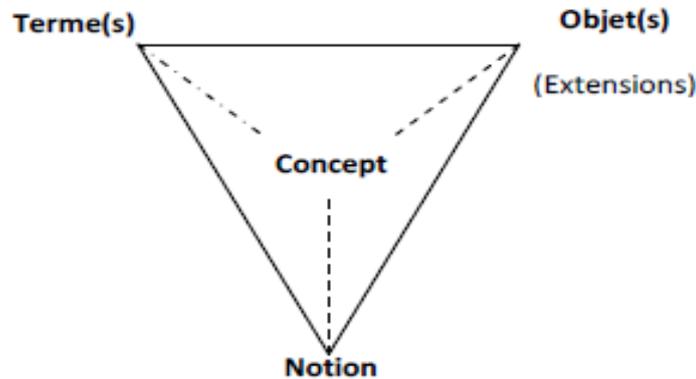


Figure I.3: Le triangle sémantique.

I.4.2. Les relations

Elles représentent des interactions entre concepts permettant de construire des représentations complexes de la connaissance du domaine. Elles sont formellement définies comme tout sous-ensemble d'un produit de n ensembles, c'est-à-dire

$R: C_1 \times C_2 \times \dots \times C_n$.

Ces relations incluent les associations suivantes :

- 1- Sous-classe-de (généralisation, spécialisation souvent appelé is a)
- 2- Partie-de (agrégation ou composition)
- 3- Associée-à
- 4- Instance de

Ces relations nous permettent d'apercevoir la structuration et l'interrelation des concepts, les uns par rapport aux autres.

Exemple : les concepts « Personnalité » et « Film » sont reliés entre eux par la relation sémantique « réalise (Personnalité, Film) ».

I.4.3. Les fonctions

Les fonctions sont des cas particuliers de relations dans lesquelles le n ème élément de la relation est unique pour les $n-1$ précédents. Formellement, les fonctions sont définies ainsi, $F: C_1 \times C_2 \times \dots \times C_{n-1}, C_n$.

Exemple de fonctions : « père-de » et « carré » sont des fonctions binaires. Tandis que, « prix-de voiture-usagée » qui calcule le prix d'une voiture de seconde main en fonction du modèle de voiture, de la date de fabrication et du nombre de kilomètres est une fonction ternaire.

I.4.4. Les axiomes

Les axiomes sont des expressions qui sont toujours vraies. Ils ont pour but de définir dans un langage logique la description des concepts et des relations permettant de représenter leur sémantique. Ils représentent les intentions des concepts et des relations du domaine.

Leur inclusion dans une ontologie peut avoir plusieurs objectifs :

- Définir la signification des composants;
- Définir des restrictions sur la valeur des attributs;
- Définir les arguments d'une relation;
- Vérifier la validité des informations spécifiées ou en déduire de nouvelles.

I.4.5. Les instances

Elles sont utilisées pour représenter les individus d'une ontologie. Ces individus correspondent à une instance concrète de la classe à laquelle ils appartiennent. Dans le cas d'une ontologie contenant des instances, celles-ci devient alors une base de connaissances.

Exemple : une « Ford » est une extension du concept « véhicule ».

I.5. Les méthodes de construction d'ontologie

Ces méthodes et outils sont essentiellement dédiés à l'extraction, à partir de documents, des concepts du domaine et des relations existant entre eux, mais offrent également des fonctionnalités de structuration permettant de bâtir de véritables ontologies.

La conception d'ontologies est une tâche difficile nécessitant la mise en place de procédés élaborés afin d'extraire la connaissance d'un domaine, manipulable par les systèmes informatiques et interprétable par les êtres humains, elle relève plus du savoir-faire que de l'ingénierie. A l'heure actuelle, il n'existe pas de consensus à propos des meilleures pratiques à adopter lors du processus de construction, ni des normes techniques régissant le processus de développement des ontologies, bien que certaines contributions dans cette direction soient déjà disponibles.

Dans ce qui suit on a choisie de détailler celles qui ont été jugées performantes.

I.5.1. Les méthodes ON-TO-KNOWLEDGE (OTK)

Développée dans le cadre d'un projet européen, la méthodologie *On-To-Knowledge* propose de construire une ontologie en tenant compte de comment l'ontologie va être utilisée par l'application plus tard. Elle se décompose en cinq étapes dont seules les quatre dernières portent réellement sur le développement d'ontologies.

La première étape (*Feasibility Study*), consiste à identifier le problème et à réaliser une étude de faisabilité par rapport à celui-ci. Cela permettra alors de sélectionner les outils nécessaires et les personnes ressources.

Les quatre phases suivantes sont quant à elles directement rattachées au cycle de vie de l'ontologie :

1- Phase de lancement (kickoff)

Durant laquelle les objectifs sont définis, les conditions requises sont spécifiées et les sources d'information identifiées. Cela permet d'aboutir à une première ontologie qualifiée de semi-formelle.

2- Phase d'amélioration (refinement)

L'ontologie précédemment créée est améliorée en fonction des retours des experts du domaine. Les relations et les axiomes sont également précisés. Cela permet d'aboutir à une ontologie opérationnelle (*target ontology*) qui doit être conforme aux objectifs identifiés durant la phase de kickoff.

3- Phase d'évaluation

Cette phase passe par trois niveaux (technologique, utilisateur, ontologique) et permet d'aboutir à une ontologie utilisable en production.

4- Phase de production et d'évolution (application & évolution)

Une fois dans le système, l'ontologie continue à évoluer. Il est donc nécessaire qu'elle soit identifiée là où les personnes qui seront chargées de sa maintenance et son évolution.

Les ontologies développées avec cette méthodologie sont fortement dépendantes de l'application.

I.5.2. Les méthodes METHONTOLOGIE

Cette méthodologie développée par A.GOMEZ-PEREZ au sein du groupe d'ontologie à l'université polytechnique de Madrid. METHONTOLOGY est née dans les travaux de software development process et dans knowledge engineering methodologies. Elle clarifie les différentes étapes de la construction d'une ontologie.

- **Spécification:** définit l'objectif de l'ontologie, les utilisateurs finaux ainsi que son dimensionnement.
- **Conceptualisation :** il s'agit ici d'organiser la connaissance. Pour cela, onze activités principales ont été définies allant de la définition des termes jusqu'à la description des instances et des règles. C'est l'une des phases les plus importantes car l'essentiel de la connaissance est acquis au début de la construction.

- **Formalisation** : c'est dans cette phase que les connaissances sont traduites sous la forme d'une ontologie. Il s'agit de passer d'un modèle purement conceptuel à une implémentation informatique.
- **Implémentation** : l'ontologie est alors traduite dans un langage ontologique comme l'OWL ou le RDF.
- **Maintenance** : la phase de maintenance corrige ou améliore l'ontologie créé

I.5.3. TERMINAE [13]

Développée au LIPN de l'Université Paris-Nord II, **TERMINAE** est une méthode et une plateforme logicielle d'aide à l'élaboration de ressources terminologiques et ontologiques à partir de textes. Cet outil intègre un environnement d'étude terminologique, un environnement d'aide à la conceptualisation et un système de gestion d'ontologies. L'un des points marquants de **TERMINAE** est qu'il conserve le lien vers le corpus.

Ainsi, l'utilisateur peut, à tout moment, consulter le contexte dans lequel une notion apparaît.

I.6. Les outils de construction d'une ontologie

On distingue deux types d'outils pour la construction des ontologies:

Les outils de construction d'ontologie dépendants du formalisme de représentation et les outils de construction d'ontologie indépendants du formalisme de représentation.

I.6.1. Les outils de construction d'ontologie dépendants du formalisme

◆ Ontolingua [14]

L'*ontolingua server* est l'environnement le plus connu pour construire des ontologies dans le langage *ontolingua* en utilisant les classes, les relations, les fonctions, les instances et axiomes. Il s'agit donc d'un ensemble d'outils et de services qui assistent la conception d'ontologie commune à laquelle collaborent des groupes de travail opérant depuis des endroits différents. Il a été élaboré par le Knowledge systems laboratory dans le cadre du programme Knowledge sharing effort de l'Arpa à l'université de Stanford. L'architecture du serveur d'ontologie permet d'accéder à une bibliothèque d'ontologies, à des traducteurs de langages de programmation (prolog, cobra's IDL, Clips, Lomm, Kif) et à un éditeur qui permet de créer et de parcourir des ontologies.

Trois types d'interaction sont possibles, il peut s'agir ainsi de :

- Collaborateurs qui souhaitent écrire et examiner des ontologies à distance.
- Applications éloignées susceptibles de vouloir interroger et modifier des ontologies sur le serveur via l'Internet
- Des applications locales.

Il y a trois différentes possibilités d'intégrer les ontologies Ontolingua :

- **Inclusion** : Une ontologie inclut et utilise les définitions d'autres ontologies;

- **Restriction** : l'ontologie importe les définitions depuis d'autres ontologies et les rend plus spécifiques;
- **Raffinement polymorphe** : on redéfinit une définition importée depuis n'importe quelle ontologie.

◆ **OntoSaurus[15]**

L'OntoSaurus créé à l'Information Sciences Institute de l'University of South California est composée de deux modules :

- Un serveur utilisant LOOM comme langage de représentation des connaissances
- Un serveur de navigation créant dynamiquement des pages HTML qui affichent la hiérarchie de l'ontologie; le serveur utilise des formulaires HTML pour permettre à l'utilisateur d'éditer l'ontologie.

En outre, des traducteurs de *Loom* vers *Ontolingua*, *Kif*, *KRSS* et *C++* ont été élaborés.

◆ **WebOnto**

WebOnto du Knowledge Media Institute de l'Open University, est une application Web pour naviguer et développer de façon collaborative les ontologies. Il supporte la navigation collaborative, la création et l'édition d'ontologies sur le Web. Les ontologies WebOnto sont implémentées dans le langage OCML, qui est une combinaison des frames et de la logique de premier ordre et qui permet de représenter les concepts, la taxonomie des concepts, les relations, les fonctions, les axiomes et les instances.

WebOnto supporte l'inclusion d'ontologie au moyen des interfaces graphiques. En ce qui concerne l'édition collaborative d'ontologie, WebOnto est le seul outil qui procure cette fonctionnalité permettant aux ingénieurs des connaissances de tenir des discussions sur les changements et les mises à jour des ontologies lors d'édition ou de navigation, en mode synchrone et asynchrone.

WebOnto distingue quatre types d'ontologies : ontologie de domaine, ontologie de tâche, ontologie de méthode, et ontologie d'application.

◆ **OilEd [16]**

OilEd (*Oil Editor*) est un éditeur d'ontologies utilisant le formalisme OIL. Il est essentiellement dédié à la construction de petites ontologies dont il est possible de tester la cohérence à l'aide de FACT (un moteur d'inférences bâti sur OIL). Le modèle de connaissance a été adapté depuis OIL à DAML+OIL et maintenant il va être adapté à OWL.

I.6.2. Les outils de construction d'ontologie indépendants du formalisme

◆ Protégé 2000 [17]

Protégé est un éditeur d'ontologies distribué en open source par l'université en informatique médicale de Stanford. Protégé n'est un outil spécialement dédié à OWL, mais un éditeur hautement extensible, capable de manipuler des formats très divers. Il permet l'édition, le contrôle, la visualisation et l'extraction d'ontologie à partir de texte. L'outil Protégé possède une interface utilisateur graphique (GUI) lui permettant de manipuler aisément tous les éléments d'une ontologie : classe, méta-classe, propriété, instance. Protégé peut être utilisé dans n'importe quel domaine où les concepts peuvent être modélisés en une hiérarchie des classes.

◆ OntoEdit [18]

OntoEdit est également un environnement de construction d'ontologies indépendant de tout formalisme. Il permet l'édition des hiérarchies de concepts et de relations et l'expression d'axiomes algébriques portant sur les relations, et de propriétés telle que la généralité d'un concept. Des outils graphiques dédiés à la visualisation d'ontologies sont inclus dans l'environnement. ONTOEDIT intègre un serveur destiné à l'édition d'une ontologie par plusieurs utilisateurs. Un contrôle de la cohérence de l'ontologie est assuré à travers la gestion des ordres d'édition. L'OntoEdit n'est pas disponible gratuitement dans sa version complète.

◆ ODE et WebOde [19]

L'outil ODE (*Ontology Design Environment*) permet de construire des ontologies au niveau connaissance, comme le préconise la méthodologie METHONTOLOGY. L'utilisateur construit son ontologie dans un modèle de type frame, en spécifiant les concepts du domaine, les termes associés, les attributs et leurs valeurs, les relations de subsomption.

I.7. Domaines d'applications des ontologies

I.7.1. Système d'information[20]

L'intégration d'une ontologie dans un système d'information vise à réduire, voire éliminer, la confusion conceptuelle et terminologique à des points clés du système, et à tendre vers une compréhension partagée pour améliorer la communication, le partage, l'interopérabilité et le degré de réutilisation possible, ce qui permet de déclarer formellement un certain nombre de connaissances utilisées pour caractériser les informations gérées par le système, et de se baser sur ces caractérisations et la formalisation de leur signification pour automatiser des tâches de traitement de l'information.

Plus concrètement, dans un système informatique, les ontologies peuvent être utilisées à différents niveaux. Basé sur son expérience, Guarino en dénombre sept :

- ✓ la spécification et à l'analyse des besoins du système ;
- ✓ la maintenance du système faisant office de documentation ou en permettant la vérification d'incohérences;
- ✓ la coopération et au partage en tant que format d'échange;
- ✓ la recherche d'informations en servant de base d'index ou de métadonnées;
- ✓ l'interopérabilité entre diverses sources de données hétérogènes;
- ✓ la compréhension du schéma conceptuel et du vocabulaire du système à travers sa visualisation;
- ✓ l'exécution et au traitement de requêtes exprimées en langue naturelle.

I.7.2. Le web sémantique [21]

Le Web actuel est essentiellement syntaxique, la structure des ressources étant bien définie, mais leur contenu restant inaccessible aux traitements machines, seuls les humains étant capables de l'interpréter.

Le Web sémantique a alors l'ambition de lever cette difficulté en associant aux ressources du Web des entités ontologiques comme références sémantiques, ce qui permettra aux différents agents logiciels d'accéder et d'exploiter directement le contenu des ressources et de raisonner dessus. Ce référencement sémantique peut aussi résoudre les problèmes d'interprétation des ressources informationnelles provenant des applications hétérogènes et réparties et de permettre ainsi à ces applications d'être intégrées sémantiquement.

L'architecture du Web sémantique repose sur une hiérarchie des langages d'assertion et de description d'ontologies ainsi que sur un ensemble de services pour l'accès aux ressources au moyen de leurs références sémantiques, pour gérer l'évolution des ontologies, pour l'utilisation des moteurs d'inférences capables d'effectuer des raisonnements complexes ainsi que des services pour la vérification de la validité sémantique de ces raisonnements

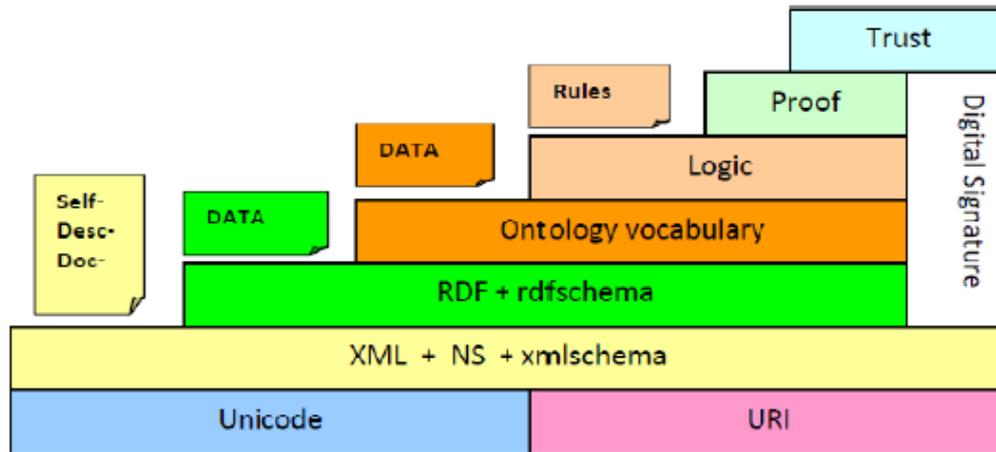


Figure I.4: Architecture du web sémantique.

Conclusion

Dans ce chapitre, nous avons essayé d'éclaircir la notion d'ontologie. Pour cela nous sommes partis des origines philosophiques du terme, puis défini son sens en informatique. Ensuite, nous avons étudié la manière de concevoir et de réaliser une ontologie en énumérant ses composants et en évoquant les méthodes et les outils pour leur construction, les domaines d'application des ontologies et préparer le prochain chapitre qui consiste en la représentation de l'ontologie à mettre en œuvre.

Chapitre II

Introduction

Le traitement et l'analyse de l'information ont été considérablement améliorés par l'outil informatique. La recherche d'information avec ses variations sémantiques, la gestion des connaissances, les systèmes coopératifs, le Web sémantique ; qui ont été considérablement progressés avec le développement des ontologies.

Dans [22] une ontologie de processus a été proposée, dans ce chapitre nous allons la présenter, aborder les règles de gestion respectées par cette dernière. puis on passera à l'interrogation et la conceptualisation.

II.1. Présentation de l'ontologie à mettre en œuvre

La modélisation des processus métier nécessite l'utilisation d'un ensemble de concepts qui va être partagé entre les différentes personnes ou modélisateurs de ces processus. L'obtention de cet ensemble de concepts nécessite donc une modélisation ontologique de processus.

Dans [22] les travaux ont montrés une convergence vers un ensemble de concepts tel que : organisation, processus, tâche, rôle et donnée. Cet ensemble de concepts a donné naissance à une ontologie de processus. Cette ontologie de processus nous permet de décrire un processus ce dernier étant un ensemble de tâche. Une tâche est le traitement le plus élémentaire qui puisse être, elle consiste donc en la transformation/transmission de donnée par un rôle. Ces rôles sont joués par des intervenants humain ou automatique.

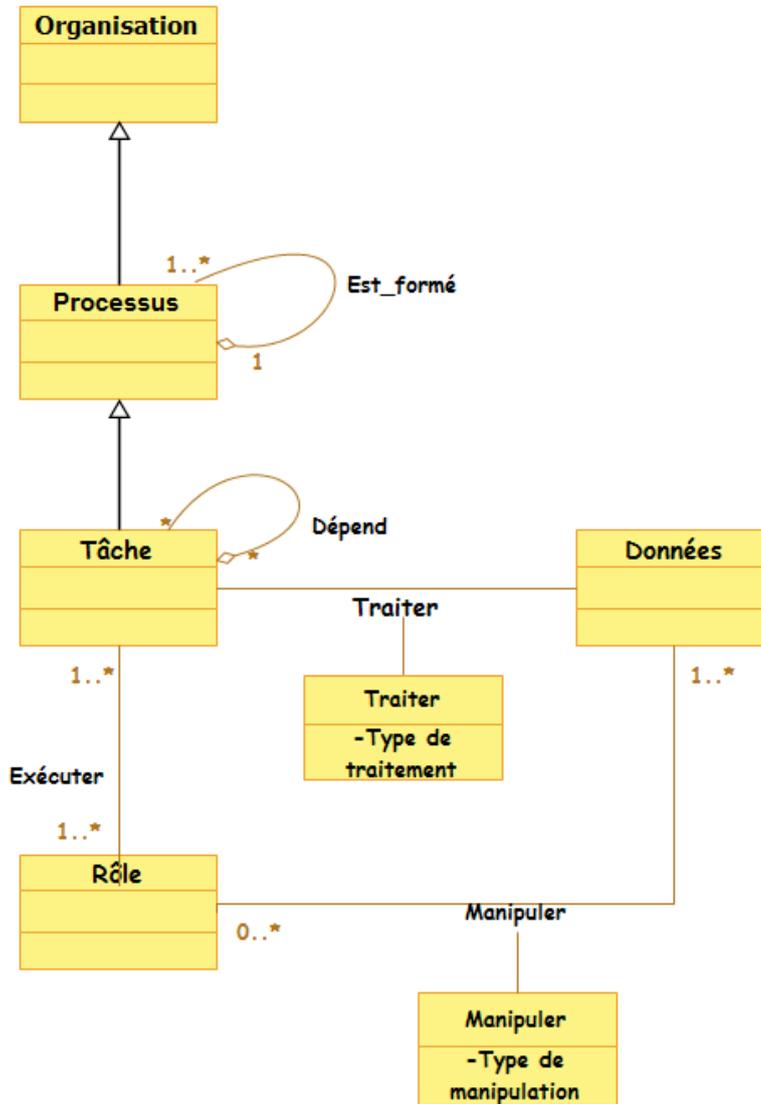


Figure II.1: Ontologies des concepts retenus pour la description d'un processus métier [22].

II.2. Définition des composants de l'ontologie

L'ontologie est composée des cinq concepts, qui sont :

1) Organisation

Une organisation est un ensemble d'individus, regroupés au sein d'une structure régulée, ayant un système de communication pour faciliter la circulation de l'information, dans le but de répondre à des besoins et d'atteindre des objectifs déterminés. [23]

2) Processus [24]

Un processus est un ensemble coordonné de tâches mis en œuvre par des acteurs (humains ou matériels) au sein d'une organisation pour aboutir à un résultat déterminé.

3) Tâche

Une tâche est "une opération, un travail à accomplir, par exemple, écrire une note, répondre au téléphone, préparer une commande, vérifier un tarif, mettre à jour un fichier".

Il faut préciser donc qu'une tâche est faite par un seul acteur.

Exemple : Enregistrement d'une fiche.

D'une façon générale, les tâches ont les caractéristiques suivantes :

- Les tâches sont réalisées par une personne ou un groupe de personnes ;
- Les tâches font appel à des compétences et des savoir-faire spécifiques.

4) Rôle [25]

Un rôle est un élément, personne physique, collectif d'individus physiques, entité organisationnelle ou programme informatique qui intervient dans l'exécution des tâches d'un processus. Un rôle peut manipuler des données.

5) Donnée

Une donnée (*data*) est une description élémentaire d'une réalité. C'est par exemple une observation ou une mesure. Les données sont généralement le résultat d'un travail préalable sur les données brutes qui permettra de leur donner un sens et ainsi obtenir une information. Les données sont un ensemble de valeurs mesurables en fonction d'un étalon de référence. La référence utilisée et la manière de traiter les données (brutes) sont autant d'interprétation implicite qui peuvent biaiser l'interprétation finale [26].

Par exemple, des données dans un graphique permettra à être humain d'y associer un sens (une interprétation) et ainsi créer une nouvelle information.

II.3. Expressivité de la modélisation

L'expressivité de la modélisation d'un processus métier est sa capacité à exprimer tous les éléments d'un processus métier. En effet, plusieurs éléments doivent être pris en compte dans la définition d'un processus métier. Selon Sini dans Sini & Si-Mohammed [27], les éléments clés d'un processus métier peuvent être

classés en groupes. On citera dans ce qui suit juste les ressources disponibles dans notre ontologie.

1. **Ressources informationnelle** : cette ressource représente les entités d'information produite ou manipulées par les différents rôles. Dans notre ontologie cette ressource est « la donnée ».
2. **Ressource fonctionnelle** : cette ressource concerne le fonctionnement du processus. Elle représente les éléments du processus à exécuter, dans notre ontologie cette ressource est représenté par « processus » et « tâche ».
3. **Ressource organisationnelle** : les ressources qui ont la responsabilité d'exécuter les éléments d'un processus. Ces ressources sont appelées participants car elles participent à la réalisation du processus. En effet, un participant est toute personne, application, programme informatique ou entité qui ont le rôle d'exécuter une tâche, notre cas le « rôle ».
4. **Ressources interactionnelles** : ces ressources permettent de capturer les interactions entre les éléments participants dans un processus.
5. **Ressources intentionnelles** : résumant l'objectif du processus, dans notre cas c'est l'organisation du service de cardiologie.
6. **Ressources comportementales** : Représentent le flux de contrôle des éléments à exécuter dans un processus. Ce sont les règles qui permettent de contraindre, contrôler et influencer l'aspect du métier du processus. Ce sont aussi les relations logiques qui contrôlent l'acheminement d'exécution des tâches. Dans notre cas c'est les axiomes.

II.4. Les règles de gestion respectées par l'ontologie

Notre ontologie offre les concepts permettant de décrire des processus structurés en termes d'enchaînement de tâches exécutées par des rôles en utilisant des ressources informationnel (donnée) les règles de gestion respectées par ce méta-modèle sont les suivantes :

1. une organisation se subdivise en un ensemble de processus qui a son tour se subdivise en un ensemble de tâches
2. Un processus peut être subdivisé par un à plusieurs sous processus
3. Une tâche peut dépendre d'une à plusieurs tâche
4. Les tâches sont exécutées par des rôles
5. Les tâches traitent des données
6. Un rôle peut manipuler une à plusieurs données
7. Un rôle exécute une à plusieurs tâches

II.5. Interrogation de l'ontologie

II.5.1. Interrogation informationnelle (donnée)

- Pour chaque donnée quels sont les rôles les manipulant ?
- Quelles sont les données présentes mais non exploitées ?
- Quelles sont les données traitées par chaque tâche ?
- Pour chaque donnée, quels sont les types de manipulation possible ?

II.5.2. Interrogation organisationnelle (rôle)

- Quelles sont Les rôles qui n'ont pas réalisé les tâches qui leurs sont assignées ?
- Pour chaque rôle, quels sont les données manipulées ?
- Pour chaque rôle, quelles sont les tâches exécutées ?

II.5.3. Interrogation fonctionnelle (tâche, processus)

- Pour chaque tâche, quelles sont les données traitées
- Pour chaque tâche, quelles sont les rôles l'exécutant
- Quelles sont les tâches spécifiques pour chaque rôle
- Quelles sont les tâches dépendantes des autres tâches
- Quelles sont les tâches qui ne traitent aucune donnée
- Quelles sont toutes les tâches formant un processus
- Quelles sont les tâches non réalisées dans un processus
- Quels sont les sous processus formant un processus
- Quels sont les processus composant une organisation

II.6. Conceptualisation de l'ontologie

II.6.1. Glossaire des termes de l'ontologie

Construire un glossaire de termes est la première tâche à effectuer dans l'étape de conceptualisation. Il recueille et décrit tous les termes (concepts, instances, attributs, relations entre les concepts, etc.) qui sont utiles et potentiellement utilisables dans le domaine que nous allons représenter leurs descriptions détaillées et non ambiguës en langage naturel.

Le tableau II.1 présente la liste des termes de l'ontologie :

Nom	Description
Organisation	Une organisation est un ensemble d'individus, regroupés au sein d'une structure régulée, ayant un système de communication pour faciliter la circulation de l'information,

Chapitre II *présentation de l'ontologie à mettre en œuvre*

	dans le but de répondre à des besoins et d'atteindre des objectifs déterminés.
Processus	Un processus est un ensemble d'activités corrélées ou interactives qui transforment les éléments d'entrée en éléments de sortie
Tache	Une tache est une activité élémentaire utilisée pour atteindre un objectif en utilisant un moyen donné. Synonymes : unité de travail ; élément de travail
Rôle	Il représente les responsabilités bien identifiées dans l'organisation de l'entreprise. Synonyme : fonctions
Donnée	une donnée est un élément fondamental (un fait, une notion, un chiffre, une instruction, un indice) prélevé à partir d'une expérience, elle est transformée/ traitée par l'être humain ou par des moyens automatiques pour ; servir à un raisonnement, une recherche ou créer de l'information

Tableau II.1 : Glossaire des termes.

II.6.2. Tableau de relations binaires

Le but de cette tâche consiste à construire une table de relations binaires décrites en détail. Pour chaque relation utilisée dans le diagramme des relations binaires, nous définissons le nom de la relation, le nom des concepts sources et cibles, le nom de la relation inverse et les cardinalités source et cible.

Nom Relation	Concept source (Domain)	Concept cible (Range)	Relation Inverse	Cardinalité Source	Cardinalité Cible
Est_formé	Processus	Processus	/	1..1	1..*
Dépend	Tâche	Tâche	/	0..*	0..*
Traiter	Tâche	Données	Traiter_par	0..*	0..*
Exécuter	Rôle	Tâche	Effectuer_par	1..*	1..*
Manipuler	Rôle	Données	Manipuler_par	0..*	1..*
Est_composé	Organisation	Processus	Appartient	1..*	1..*
Est_composé	Processus	Tâche	Appartient	1..*	1..*
Exécuter_par	Tâche	Rôle	Exécuter	1..*	1..*
Traiter_par	Données	Tache	Traiter	0..*	1..*
Manipuler_par	Données	Rôle	manipuler	0..*	1..*
Appartient	Processus	organisation	Est_composé	1..*	1..*
Appartient	Tâche	Processus	Est_composé	1..1	1..*

Tableau II.2 : Les relations binaires.

III.6.3. Diagramme des relations binaires

Une relation binaire permet de relier deux concepts entre eux (un concept source et un concept cible). « Si R est une relation entre deux concepts C1 et C2 alors pour tout couple d'instance des concepts C1 et C2, il existe une relation de type R qui lie deux instances de C1 et C2 ». Cette tâche permet de représenter de manière graphique les différentes relations qui existent entre les divers concepts que ce soit de même ou de différente hiérarchie.

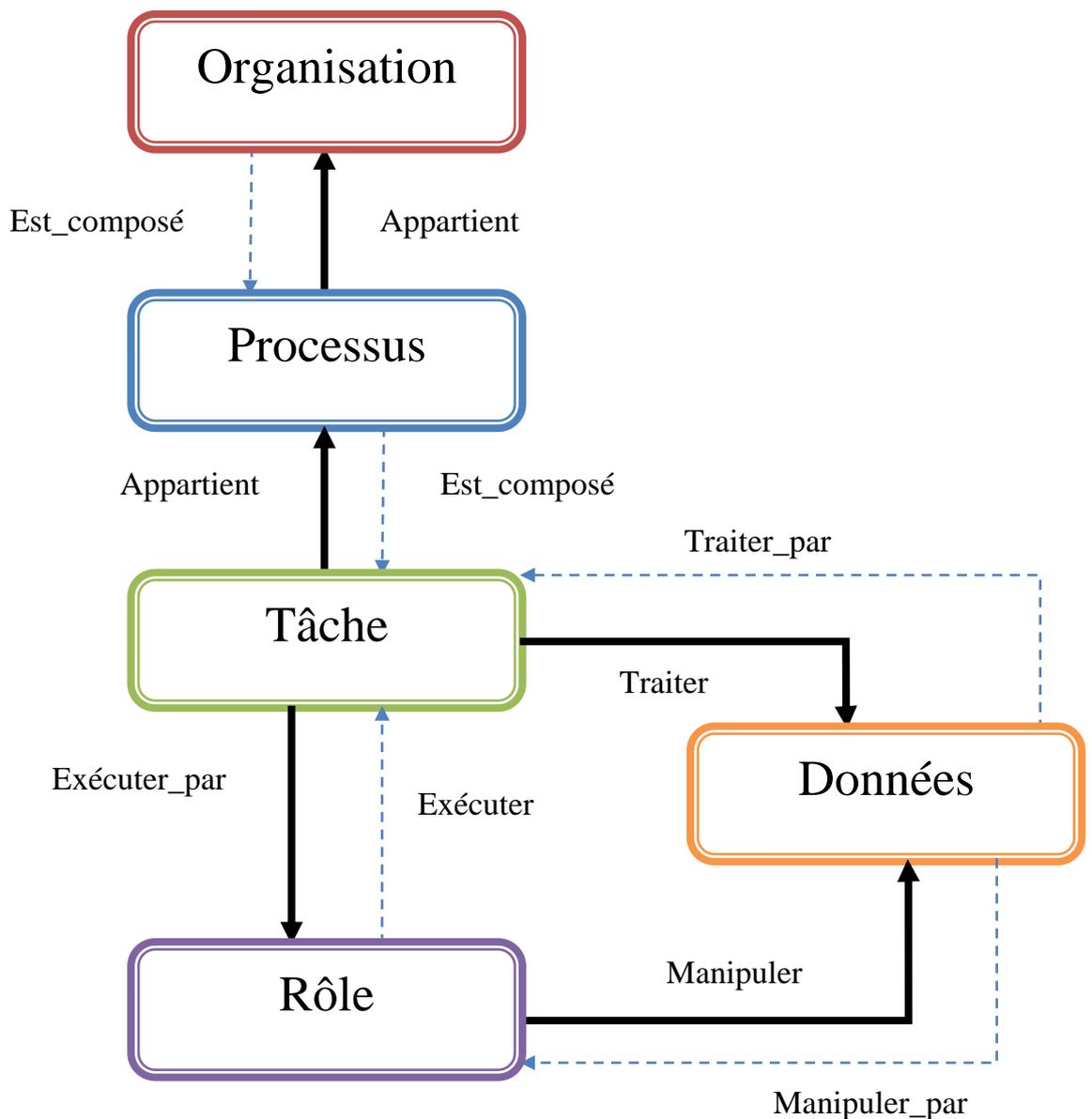


Figure II.2 : Diagramme des relations binaires de notre ontologie

Chapitre II présentation de l'ontologie à mettre en œuvre

II.6.4. Dictionnaires des concepts

Un dictionnaire de concepts contient tout les concepts du domaine, leurs synonymes, leurs acronymes, leurs attributs et leurs relations.

Concept	Attribut	Relation	Acronyme	Synonyme
Organisation	Nom_organisation	Est_composé	Org	Entreprise, Système, Etablissement
Processus	Nom_processus	Est_composé Est_formé	Process	---
Tâche	---	Traiter Executer_par	T	Elément de travail
Rôle	---	Manipuler Executer	Rl	Fonction
Donnée	---	Traiter_par Manipuler_p ar	DN	Elément, Information
Traiter_donnée	type_traitement	Traiter Traiter_par	TR_donnée	---
Manipuler_donné e	type_manipulation	Manipuler Manipuler_p ar	MN_donné e	---

Tableau II.3 : Dictionnaires des concepts.

II.6.5. Classification des concepts en hiérarchie

La hiérarchie de la classification de concepts démontre l'organisation des concepts de l'ontologie dans un cadre hiérarchique qui exprime les relations sous-classe.

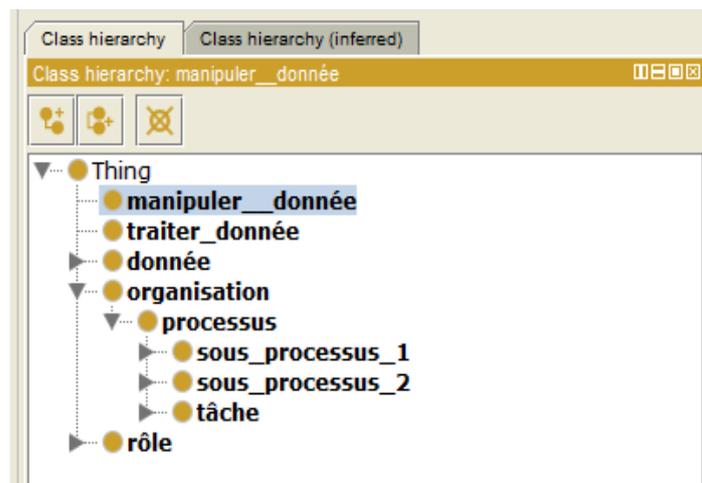


Figure II.3 : Taxonomie des concepts.

Conclusion

Dans ce chapitre nous avons présenté notre ontologie qui est un ensemble de concepts, et lien entre ces concepts qui décrivent un processus métier de n'importe quel organisme. Pour pouvoir mettre en œuvre cette ontologie, il est nécessaire de définir un processus d'un cas réel qui sera l'objet du prochain chapitre.

Chapitre III

Introduction

Depuis des décennies, les organisations ont dû s'investir d'une façon significative dans l'utilisation des technologies de l'information pour automatiser les divers processus intervenant dans le cycle de vie de leurs missions. En effet, un fonctionnement efficace des organisations, impose de s'appuyer sur des processus métiers robustes, et adaptés à leurs activités. La définition et l'exécution de ces processus nécessitent respectivement une modélisation. Cette modélisation doit permettre une compréhension claire et transparente des activités considérées, les dépendances entre les activités et les rôles (gens, machines, informations) nécessaires pour le processus.

Dans ce chapitre, nous allons dans un premier temps aborder la notion de processus et sa modélisation en donnant les différentes définitions existantes dans la littérature, puis nous présenterons un cas de processus métier réel afin d'instancier l'ontologie présentée dans le chapitre précédent.

III.1. Processus

III.1.1. Définition d'un processus

L'organisation est modélisée comme un ensemble de processus permettant d'identifier les besoins des clients et de les transformer en un livrable et/ou service conformes à leurs attentes.

Un processus peut avoir plusieurs définitions parmi elles, on cite :

Définition 1

"Un processus est un ensemble partiellement ordonné d'étapes exécutées en vue de réaliser au moins un objectif" [28].

Définition 2

La norme ISO 9001-2000 [29] voit le processus comme:

«Toute activité utilisant des ressources et gérée de manière à permettre la transformation d'éléments d'entrée en éléments de sortie peut être considérée comme un processus»

Définition 3

Morley [30] s'intéresse plus à la structure d'un processus: *"Un processus représente l'organisation d'un ensemble finalisé d'activités effectuées par des acteurs et mettant en jeu des entités".*

Définition 4

Une définition plus complète provient de Théroude [31], "*Un processus est défini comme un enchaînement partiellement ordonné d'exécution d'activités qui, à l'aide de moyens techniques et humains, transforme des éléments d'entrée en éléments de sortie en vue de réaliser un objectif dans le cadre d'une stratégie donnée*".

III.1.2. Caractéristique des processus

Un processus possède donc les caractéristiques suivantes [32] :

- a un objectif ;
- a des entrées spécifiques ;
- a des sorties spécifiques ;
- utilise des ressources ;
- a un nombre d'activités qui se déroulent en un ordre précis ;
- peut concerner plus d'une unité d'organisation ;
- crée de la valeur.

III.1.3. Type de processus

Quand on parle de processus, il faut bien préciser de quel "type" il s'agit. La norme AFNOR de juin 2000 sur le management des processus [33] propose trois grandes familles de processus :

1) les processus métiers (de réalisation ou opérationnels)

Ils contribuent directement à la réalisation du produit, depuis la détection du besoin client jusqu'à sa satisfaction. Ils regroupent des activités liées au cycle de vie du produit : recherche de nouveaux produits, conception, achats et approvisionnements, logistique, production, commercialisation.

2) Les processus matériels

Processus se caractérisant par la manipulation, l'assemblage, la livraison, la transformation, la mesure et le stockage d'objets physiques. Ces processus sont liés à des activités manuelles ou automatisées. Il ne s'agit pas d'activités administratives ou intellectuelles.

3) Les processus informationnels

Les processus informationnels sont liés à des activités automatisées (exécutées par ordinateur) ou semi automatisées (accomplies par des humains en interaction avec des programmes). Ces activités créent, traitent, gèrent et fournissent de l'information. L'infrastructure de base des processus informationnels est fournie par des systèmes

d'information de l'entreprise, tels que les systèmes de gestion de bases de données, les systèmes de gestion de transactions.

III.2. Processus métier

III.2.1. Définition du processus métier

Le *Workflow Management Coalition* (WfMC) [34] définit un processus métier comme étant : « un ensemble d'une ou plusieurs procédures ou activités liées entre elles pour réaliser collectivement un objectif ou une politique métier en définissant les rôles et les interactions fonctionnelles au sein d'une structure organisationnelle ».

Morley définit un processus métier comme suit [35] : « Un processus métier est une orchestration d'activités, incluant une interaction entre différents acteurs sous la forme d'échange d'informations, réalisant des objectifs métiers ».

III.2.2. Quand et pourquoi faire ?

La description des processus métiers apporte une vision du métier réel, et constitue un excellent instrument de formalisation et d'analyse, dans la construction des systèmes. A ce titre, ils occupent une place centrale dans les systèmes d'information d'entreprise d'aujourd'hui (banque, assurance, télécom).

Dans le cadre d'un projet de développement, ils sont surtout utilisés dans les activités orientées métier (expression des besoins, spécification ou analyse suivant la terminologie employée). [36]

III.2.3. Typologie des processus métiers

Par ailleurs, il existe différentes catégories de processus métier d'entreprise. Paul Allen [37] propose la classification suivante :

- Les processus clients (*customer process*), qui fournissent les produits ou services aux clients. Par exemple, la vente de produit.
- Les processus supports (*sustaining process*), responsables de la valeur ajoutée aux clients, sans toutefois échanger directement avec eux. Par exemple : La mise à jour du catalogue produit.
- Les processus internes (*enabling process*), qui n'ont pas d'interaction avec le client, fournissent des services de gestion interne comme la paie du personnel ou le processus d'embauche.

III.2.4. La gestion de processus métier (BPM)

Van der Aalst et ses collègues définissent le BPM (*Business Process Management*) comme : « la gestion des processus métiers en utilisant des méthodes, des techniques

et des logiciels pour modéliser, exécuter, contrôler et analyser les processus opérationnels en s'appuyant sur des acteurs qui peuvent être : des êtres humains, organisations, des applications, documents et autres sources d'information » [38]. En effet, le BPM consiste à gérer les processus métiers: (1) sur un plan global en cherchant à prendre en compte les processus de bout en bout, depuis la chaîne d'approvisionnement jusqu'aux activités internes et externes d'une entreprise. (2) sur un plan de cycle de gestion en s'intéressant aux différentes étapes du cycle de vie des processus depuis la modélisation jusqu'à l'exécution et le diagnostic.

Le BPM permet à une entreprise d'harmoniser ses processus avec sa stratégie, débouchant ainsi sur une meilleure performance grâce à l'amélioration d'activités spécifiques que ce soit au sein d'un service particulier, d'une entreprise ou de plusieurs organisations.

III.2.5. Modélisation des processus métiers

III.2.5.1. Définition de la modélisation

Le Moigne [39] définit la modélisation comme *"l'élaboration et la construction intentionnelle par composition de symboles, de modèles susceptibles de rendre intelligible un phénomène perçu complexe, et d'amplifier le raisonnement de l'acteur projetant une intention délibérée au sein du phénomène"*.

III.2.5.2. Définition d'un modèle

Un modèle est une abstraction de la réalité. Cette représentation est construite, vérifiée, analysée et manipulée pour maîtriser la réalité et mieux la comprendre [40]

III.2.5.3. Pourquoi modéliser les processus métiers ?

La modélisation des processus métiers est au cœur même de la démarche d'analyse dynamique d'une organisation. Que ce soit dans le cadre d'une démarche d'amélioration ciblée ou d'une réorganisation plus globale, la modélisation des processus permet de formaliser le fonctionnement précis d'une organisation en utilisant un langage standard et aisément compréhensible. [41]

Les objectifs d'une modélisation des processus métiers sont :

- ✓ l'optimisation de la chaîne de valeur de l'entreprise en définissant, supervisant et améliorant les processus métiers ;
- ✓ la capitalisation sur l'organisation (personnels, rôle) et sur le système d'information ;
- ✓ offrir une flexibilité aux processus pour qu'ils s'adaptent aux changements, et l'automatisation des processus qui peuvent l'être.

III.2.5.4. Buts d'une démarche de modélisation

On modélise pour une raison essentielle, les modèles permettent de mieux comprendre le système que l'on développe.

Les objectifs d'une modélisation de processus sont multiples [42] :

- Expliquer le fonctionnement de l'organisation ;
- Comprendre les processus pour les modifier ou évaluer leur valeur ajoutée ;
- Optimiser les échanges et la collaboration entre acteurs (internes ou externes) ;
- Quantifier les ressources (acteurs) nécessaires à l'exécution des processus ;
- Concevoir des moyens ou des supports adaptés à l'exécution et notamment le système d'information ;
- Pour simuler un processus : à des fins pédagogiques, de détection d'anomalies de fonctionnement, de diagnostic de pannes, de conception assistée par ordinateur.

III.3. Présentation du processus réel

Processus : Admission des malades au sein du service de cardiologie du CHU.

1. Sous_ processus 1 : Mode d'admission

1.1. Voix des urgences

- Le malade passe au service des urgences, ce dernier lui établit un dossier malade.
- Le service des urgences décide de transférer le malade vers le service de cardiologie.
- Le service des admissions établit un dossier administratif (fiche navette + bulletin d'admission) pour le malade.
- Le malade passe au service cardiologie muni du dossier administratif et du dossier malade fourni par le service des urgences.
- La secrétaire médicale du service de cardiologie établit un nouveau dossier malade incluant l'ancien dossier.
- La secrétaire médicale transmet le dossier administratif aux infirmiers et le dossier malade au médecin traitant.

1.2. Evacuation

- Envoi d'une demande d'hospitalisation au service cardiologie par l'établissement qui évacue le malade.
- La secrétaire médicale établit une réponse:
 - s'il y a une place la secrétaire médicale établit un RDV
 - sinon la demande est mise en attente en fonction de sa date d'arrivée.

- Si l'évacuation était pour un pace maker, la secrétaire médicale établit un bon de commande.

1.3. Voix d'un privé

- Le malade se présente au service cardiologie muni d'une lettre d'hospitalisation fournie par l'établissement privé.
- Le malade remplit une demande d'hospitalisation au service cardiologie, ce dernier l'envoie au service des admissions.
- Le service admissions établit une fiche navette et un bulletin d'admission (dossier administratif).
- La secrétaire médicale du service cardiologie établit un dossier malade.
- Le dossier administratif est remis aux infirmiers, et le dossier malade au médecin traitant.

1.4. Hôpital du jour

- Le service des urgences décide de transférer le malade vers le service cardiologie pour une hospitalisation d'une journée.
- Etablissement d'un billet de salle et d'une fiche navette par le service des admissions.
- Le médecin traitant décide d'hospitaliser le malade.
- Réadmission du malade au service de cardiologie en établissant une fiche navette et un bulletin d'admission, et en le supprimant de la liste HDJ.
- La secrétaire médicale établit un nouveau dossier malade.
- Remise du dossier malade au médecin traitant et du dossier administratif aux infirmiers.

2. Sous_processus2 : Mode de sortie

2.1. Décision du médecin

- La secrétaire médicale remet un compte rendu établi par le médecin traitant qui comporte : résumé clinique, résumé standard, et une carte de sortie.
- La fiche navette est envoyée au service des admissions pour remplir le cadre qui leur est réservé, puis le remet à la secrétaire médicale du service de cardiologie.
- Envoi du dossier malade au bureau des archives.

2.2. Contre avis médical

- Le malade sort du service après son hospitalisation sans l'accord de son médecin traitant.
- Le médecin traitant signale et mentionne un contre avis médical sur la fiche navette du malade et l'envoi au service des admissions.

- Le service des admissions remplit le cadre qui lui est réservé sur la fiche navette et la renvoi au service de cardiologie.
- Envoi du dossier malade au bureau des archives.

2.3. Décès

- Le malade est décédé.
- Le médecin qui constate la mort du malade mentionne le décès, la date du décès et l'heure sur la fiche navette du malade.
- Le médecin envoie la fiche navette au service des admissions pour signaler le décès.

2.4. Evacuation

- Le malade évacué vers un autre établissement ou autre service.
- La secrétaire établit un compte rendu (résumé clinique, résumé standard, carte de sortie).
- La fiche navette transmise au service des admissions.

Après avoir décrit notre processus textuellement, nous allons essayer d'instancier l'ontologie et voir est ce qu'elle permet de décrire toute la réalité en utilisant ces concepts.

Concepts	Cas réel
Organisation	Service de cardiologie
Processus	Admission des malades au sein du service de cardiologie du CHU
Sous processus 1	Mode d'admission
Sous processus 1.1	Voix des urgences
Sous processus 1.2	Evacuation admission
Sous processus 1.3	Voix d'un privé
Sous processus 1.4	Hôpital du jour
Sous processus 2	Mode de sortie
Sous processus 2.1	Evacuation sortie
Sous processus 2.2	Décision du médecin
Sous processus 2.3	Contre avis médical
Sous processus 2.4	Décès
Tâche	
	Passage au service des urgences
	Etablissement d'un dossier malade
	Décision du transfert du malade vers le service de cardiologie
	Etablissement d'un dossier administratif (fiche navette + bulletin d'admission)

	Passage au service de cardiologie
	Etablissement d'un nouveau dossier malade
	Remise du dossier administratif aux infirmiers
	Remise du dossier malade au médecin traitant
	Envoi d'une demande d'hospitalisation au service de cardiologie
	Etablissement d'une réponse
	Etablissement d'un rendez vous
	Mise en attente de la demande
	Etablissement d'un bon de commande pour un pace maker
	Etablissement d'une lettre d'hospitalisation
	Présentation au service de cardiologie
	Remplissage d'une demande d'hospitalisation au service de cardiologie
	Envoi de la demande d'hospitalisation au service administratif
	Transfert du malade au service cardiologie
	Etablissement d'un billet de salle et d'une fiche navette
	Hospitalisation du malade au service de cardiologie
	Réadmission du malade au service cardiologie
	Suppression du malade de la liste HDJ
	Etablissement d'un compte rendu (résumé clinique, résumé standard, carte de sortie)
	Remise du compte rendu (résumé clinique, résumé standard, carte de sortie)
	Envoi de la fiche navette au service des admissions
	Remplissage du cadre réservé pour la sortie du malade
	Remise de la fiche navette à la secrétaire médicale
	Envoi du dossier malade au bureau des archives
	Sortie du malade du service sans l'accord du médecin

	Signalisation et mention d'un contre avis médicale sur la fiche navette
	Mention du décès du malade (date, heure) sur la fiche navette
	Envoi de la fiche navette au service des admissions
	Evacuation du malade vers un autre service
	Etablissement d'un compte rendu
	Transmission de la fiche navette au service des admissions
Rôle	Malade
	Service des urgences
	Médecin traitant
	Service des admissions
	Secrétaire médicale
	Etablissement privé
	Infirmiers
	Bureau des archives
	Autres établissement
Données	Dossier administratif
	Dossier malade
	Demande d'hospitalisation
	Lettre d'hospitalisation
	Billet de salle
	Compte rendu
	Carte de sortie
	Résumé clinique
	Résumé standard
	Bulletin d'admission
	Fiche navette
	Liste des malades HDJ
	Bon de commande

Tableau III.1 : Instanciation de l'ontologie.

III.4. Conceptualisation de l'ontologie

Cette étape consiste à identifier et à structurer, à partir des sources d'informations les connaissances du domaine. Elle permet d'aboutir à un ensemble de représentations intermédiaires semi-formelles indépendamment des langages de formalisations à utiliser pour l'ontologie. A la fin de cette phase nous obtenons une ontologie conceptuelle.

III.4.1. Glossaire des termes

Ce glossaire contient tous les termes relatifs à l'organisation du service de cardiologie. Le tableau III.2 fournit une liste des termes les plus importants qui seront plus tard repris dans l'ontologie.

Terme	Signification
Service cardiologie	C'est un service hospitalier, qui accueille les patients ayant des problèmes cardiaques
Service d'admission	L'acceptation formelle par un hôpital ou un autre établissement de soins de santé en milieu hospitalier d'un patient qui doit être fournie avec salle, un conseil, et un service de soins infirmiers continue dans une zone de l'hôpital ou de l'établissement où les patients résident généralement au moins une nuit.
Service des urgences	Service hospitalier qui accueille, à toutes heures de la journée, les patients amenés par les services de secours ou tout autre patient vivant une situation d'urgence médicale.
Etablissement privé	Centre hospitalier, appartenant à des individus non de la collectivité ou de l'État
Secrétaire médicale	Est un travailleur exerçant des fonctions de secrétariat pour le compte d'un ou plusieurs médecins dans un cabinet médical
Infirmier	Un infirmier est un professionnel de la santé qui soigne les patients tout en s'assurant de leur confort, leur hygiène et leur bien-être.
Un malade	Dont la santé est altérée; qui est atteint d'une maladie ; qui éprouve un malaise
Médecin traitant	Est celui qui vous soigne habituellement, qui vous connaît le mieux, ça peut être par exemple votre médecin de famille. Il se charge de vous prescrire les examens médicaux, il vous envoie vers un spécialiste, coordonne votre suivi et tient également à jour votre dossier médical
Dossier malade	Les informations médicales portant sur le patient, comme les résultats d'examen ; les comptes rendus de consultation, d'intervention, d'exploration ou d'hospitalisation ; les protocoles et des prescriptions thérapeutiques mises en œuvre

	; les feuilles de surveillance ; les correspondances entre professionnels de santé
Demande d'hospitalisation	Formulaire contenant des champs d'informations concernant le malade.
Fiche navette	Formulaire contenant des informations concernant le malade ainsi que les actes médicaux, chirurgicaux, les médicaments et traitements reçus.
Carte de sortie	Carte ayant tous les renseignements nécessaires sur la sortie du patient
Résumé clinique	Résumé standardisé et concis du dossier médical du patient que les hôpitaux généraux sont tenus d'enregistrer.
Lettre d'hospitalisation	message rédigé sur une feuille de papier adressée à un hôpital pour une admission et un séjour
Dossier administratif	Il est constitué d'un bulletin d'admission et d'une fiche navette
Bulletin d'admission	Information officielle sous une forme concise afin d'introduire un patient dans un hôpital
Bureau des archives	Bureau où tous les dossiers médicaux sont archivés après la sortie du malade
Billet de salle	Billet contenant l'identifiant du malade avec le numéro de chambre
Bon de commande	Un document établi par la secrétaire médical pour obtenir un pace maker
Résumé standard	Document contenant le traitement suivi par le malade
Liste HDJ	Liste contenant le nom des malades hospitalisés pour une journée

Tableau III.2 : Glossaire des termes.

III.5. Formalisation

Dans cette étape, nous allons utiliser le formalisme des logiques de description afin de formaliser le modèle conceptuel obtenu dans l'étape de conceptualisation.

III.5.1. Construction de la ABox

Voici quelques assertions sur les individus et relations représentés dans le tableau suivant :

Assertion sur les individus
Organisation (Service de cardiologie)
Processus (Admission des malades)
Sous_processus1 (Mode d'admission)
Sous_processus2 (Mode de sortie)
Traiter-donnée(Consultation)
Traiter-donnée (Enregistrement)
Manipuler-donnée(Modification)
Manipuler-donnée(Suppression)

Tableau III.3 : Assertion sur les individus.

Assertion des relations
Appartient (Admission des malade, service de cardiologie)
Exécuter par (Etablissement d'un dossier administratif, service des admissions)
Manipuler (secrétaire médicale, bon de commande)
Traiter par (demande d'hospitalisation, remplissage d'une demande d'hospitalisation au service cardiologie)

Tableau III.4 : Assertion des relations.

III.5.2. Construction de la TBox

Voici quelques définitions et axiomes terminologiques représentés dans le tableau suivant :

Concept	Définition	Relation et subsomption
Organisation	$\equiv \exists \text{est-composé. processus}$	$\text{Organisation} \subseteq \text{Thing}$
Processus	$\equiv (\exists \text{ Est_formé.Sous_processus1} \cap \exists \text{ est_formé.Sous_processus2}) \cup \exists \text{ est-composé. T\^ache}$	$\text{Processus} \subseteq \text{Organisation}$
T\^ache	$\equiv \forall \text{ Traiter. donn\^ee} \cup \exists \text{ ex\^ecuter_par. R\^ole}$	$\text{T\^ache} \subseteq \text{Processus}$
R\^ole	$\equiv \exists \text{ Ex\^ecuter. t\^ache} \cup \forall \text{ manipuler. Donn\^ee}$	$\text{R\^ole} \subseteq \text{Thing}$
Donn\^ee	$\equiv \forall \text{ Traiter_par. t\^ache} \cap \forall \text{ manipuler_par. R\^ole}$	$\text{Donn\^ee} \subseteq \text{Thing}$

Tableau III.5 : Définition de la Tbox.

Conclusion

Dans ce chapitre nous avons mis l'accent sur la notion de processus, leurs caractéristiques ainsi que les différents types de processus constituant l'entreprise.

Nous nous sommes ensuite concentrés sur les processus métiers en évoquant leur place, leur gestion et leur modélisation qui sont impératives. Pour mieux expliquer la notion de modélisation des processus, nous avons défini cette dernière tout en expliquant le but d'une démarche de modélisation.

A la fin de ce chapitre nous avons présenté le cas réel utilisé pour instancier l'ontologie présentée dans le deuxième chapitre.

Dans le chapitre suivant nous allons passer à l'implémentation de notre ontologie sous l'éditeur protégé 4.3.

Chapitre IV

Introduction

Ce présent chapitre présente notre contribution à la problématique posée dans ce mémoire à savoir l'instanciation de l'ontologie décrite précédemment avec un cas réel et son implémentation sous protégé.

Dans ce chapitre nous allons présenter le travail d'implémentation que nous avons réalisé, et qui consiste premièrement à l'édition de notre ontologie sous protégé. Nous commençons par une présentation de cet outil, ensuite nous détaillons les étapes d'opérationnalisation et d'implémentation de notre ontologie. Enfin, nous illustrons la phase de vérification et test de consistance par le biais du moteur d'inférence PELLET.

IV.1. Présentation de protégé

Protégé est un éditeur d'ontologies distribué en open source par l'université d'informatique médicale de Stanford. Il fournit à une communauté d'utilisateurs une série d'outils logiciels pour construire des modèles de domaine et des applications basées sur la connaissance des ontologies. Protégé met en application un ensemble riche de structures de «connaissance, modélisation et actions» qui soutiennent la création, la visualisation, et la manipulation des ontologies dans divers formats de représentation, c'est un éditeur hautement extensible. Pour l'instanciation de notre ontologie nous avons utilisé la dernière version Protégé 4.3.

Protégé est aussi une plate-forme extensible, grâce au système de plug-ins, qui permet de gérer des contenus multimédias, interroger, évaluer et fusionner des ontologies, etc. L'outil Protégé possède une interface utilisateur graphique (GUI) lui permettant de manipuler aisément tous les éléments d'une ontologie : classe, méta-classe, propriété, instance,...etc. Protégé peut être utilisé dans n'importe quel domaine où les concepts peuvent être modélisés en une hiérarchie des classes.

Protégé permet aussi de créer ou d'importer des ontologies écrites dans les différents langages d'ontologies tels que : RDF-Schéma, OWL, DAML, OIL. Cela est rendu possible grâce à l'utilisation de plugins qui sont disponibles en téléchargement pour la plupart de ces langages.

IV.2. Les langages de représentation d'ontologie sous protégé

Il existe de nombreux langages informatiques, plus ou moins récents, spécialisés dans la création et la manipulation des ontologies. Nous en décrivons quelques-uns dans la suite.

IV.2.1. XML

XML (*eXtended Markup Language*) [43] est un langage permettant de générer des balises pour la structuration de données et de documents. Il permet la représentation et l'échange de documents semi-structurés. **XML-schéma** [44] permet de définir la structure, les contraintes, et la sémantique de documents XML. Ce langage n'est pas vu comme un langage d'ontologies car il a été créé pour vérifier la structure de documents XML. Les primitives qu'il met en place sont plutôt orientées application que concept. En effet, la sémantique définie dans le document est interprétable dans le contexte de l'opération faite sur le document mais ne permet pas d'établir des inférences en dehors de ce contexte. XML et XML-schéma sont considérés comme des langages définissant le format de « message » alors qu'un langage d'ontologies a pour but de « représenter » la connaissance.

IV.2.2. RDF [45]

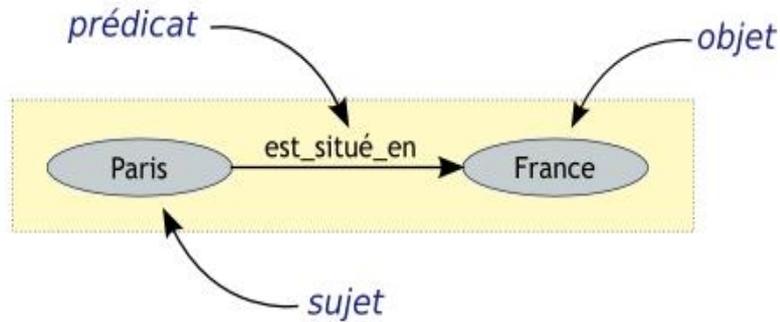
RDF (*Resource Description Framework*) est un modèle de représentation sémantique des informations du Web qui utilise la syntaxe d'XML. Ces représentations comportent des méta-données sur les ressources du Web comme les auteurs de pages Web et leur date de création. Les ressources du Web sont l'élément de base de RDF. Chaque ressource est pourvue d'un identifiant uniforme de ressource (URI).

Développé par le *World Wide Web Consortium* (W3C) en 1999, le Resource Description Framework (RDF) dans le but de standardiser les définitions et les usages des méta-données, RDF est également utile à la représentation de données en elles-mêmes. Les éléments principaux de RDF sont les objets, leurs attributs et les valeurs de ces attributs.

RDF décrit les ressources simplement et clairement en utilisant des triplets constitués d'un sujet, d'un prédicat et d'un objet :

- Le sujet représente la ressource à décrire qui peut être une URI ;
- Le prédicat représente un type de propriété applicable à cette ressource et qui peut être réutilisé et identifié par URI de la relation ;
- L'objet représente une donnée ou une autre ressource.

L'expression de l'assertion " Paris est en France " peut se faire par le biais de l'écriture d'un triplet RDF, que l'on peut représenter sous la forme d'un graphe sujet-prédicat-objet



L'intérêt principal de RDF est de définir un mécanisme permettant de décrire des données indépendamment de tout domaine et de toute spécificité. De même qu'avec XML, RDF ne permet pas la déclaration de propriétés particulières ; leur définition est totalement libre.

IV.2.3. RDFS

Les schémas RDF (RDFS) permettent de définir le vocabulaire utilisé dans les descriptions RDF. Il confère un formalisme de représentation riche incluant des classes, sous classes, propriétés, sous-propriétés, des règles d'héritages de propriétés avec la possibilité de restreindre leur domaine d'origine (`rdf:domain`) et leur domaine d'arrivée (`rdf:range`), mais ne normalise pas les inférences que l'on pourrait faire avec. La structure objet-classe des RDFS permet de représenter un modèle du domaine en définissant des objets du domaine et leurs relations pour rendre compte d'une ontologie. [46]

IV.2.4. DAML-OIL

Dans l'optique d'une utilisation d'ontologies sur le Web, le langage RDF-S a été enrichi par l'apport du langage OIL (*Ontology Interchange Language*) qui permet d'exprimer une sémantique à travers le modèle des frames tout en utilisant la syntaxe de RDF-S. OIL offre de nouvelles primitives permettant de définir des classes à l'aide de mécanismes ensemblistes issus des logiques de description (intersection de classes, union de classes, complémentaire d'une classe). Il permet également d'affiner les propriétés de RDF-S en contraignant la cardinalité ou en restreignant la portée [47]. Le langage OIL a été fusionné avec le langage DAML pour former le DAML+OIL. DAML (*Darpa Agent Markup Language*) est conçu pour permettre l'expression d'ontologies dans une extension du langage RDF. Il offre les primitives usuelles d'une représentation à base de frames et utilise la syntaxe RDF [48]. L'intégration de OIL rend possible les inférences compatibles avec les logiques de description, essentiellement les calculs de liens de subsomption.

IV.2.5. OWL

Ce langage est recommandé par le W3C comme un standard pour le web sémantique depuis 2004 et constitue un pilier pour le web sémantique, selon Tim Berners-Lee [49]. OWL dérive de la fusion des deux langages OIL et DAML.

Le langage OWL permet une interprétation du contenu Web par les machines supérieure à celle offerte par les langages XML, RDF et le schéma RDF (RDF-S), en fournissant un vocabulaire supplémentaire avec une sémantique formelle [50].

OWL est composé de trois parties [51] : Une classe, c'est à dire un groupe d'individus partageant les mêmes caractéristiques. Les classes peuvent être organisées hiérarchiquement selon une taxonomie (classification). Les classes définies par l'utilisateur sont d'ailleurs toutes des enfants de la « super-classe » OWL : Thing. Peut être comparé à une table dans le domaine des bases de données relationnelles. Une propriété qui permet de définir des faits ou des relations entre ces classes. Il existe en OWL deux types de propriétés : propriété d'objet (owl:ObjectProperty) qui définit une propriété entre deux individus d'une classe ou de plusieurs classes, et une propriété de type de données (owl:DatatypeProperty), c'est à dire une relation entre une valeur ou donnée et un individu d'une classe, l'équivalent d'un champ d'une table dans une base de données relationnelles. Les propriétés peuvent aussi être organisées hiérarchiquement. Une instance, c'est-à-dire un individu d'une classe qui peut prendre les caractéristiques définies par les propriétés.

OWL est un langage de représentation des connaissances normalisé par le W3C. La première mouture d'OWL était découpée en trois sous-ensembles, OWL-Lite, OWL-DL et OWL-Full, classés par ordre d'expressivité croissante.

a) OWL-Lite [52]

OWL Lite permet de représenter des classifications sous forme hiérarchique et d'exprimer des contraintes simples (contraintes de cardinalité de type 0 ou 1). La disjonction de classes, la définition de classes à partir d'une union de classes, et bien d'autres possibilités offertes par OWL DL, ne sont pas autorisées.

b) OWL-DL

OWL-DL appelé ainsi en référence aux logiques de description est un sous ensemble d'OWL-Lite qui offre un maximum d'expressivité tout en maintenant la complétude et la décidabilité des algorithmes d'inférence. OWL-DL offre notamment la possibilité d'exprimer des concepts unions, des concepts énumérés, des concepts disjoints et la négation de concepts.

c) OWL-Full

Est un sous ensemble de OWL-DL qui offre la possibilité de recouvrement des types :

- Un concept peut aussi être un individu ou une propriété et réciproquement ;
- La contrepartie de cette expressivité est la perte de la décidabilité : rien ne garantit qu'un moteur d'inférence fournisse une réponse en un temps fini.

IV.3. Les moteurs d'inférence

La sémantique formelle du langage OWL permet l'application des techniques de raisonnement pour effectuer des dérivations logiques. Ces dérivations sont effectuées par des moteurs d'inférence (également nommés moteurs de raisonnement, raisonneurs sémantiques, ou tout simplement des raisonneurs), ce sont des programmes qui peuvent lire des ontologies à partir de fichiers OWL ou des serveurs web distants, ce sont donc des systèmes capables de gérer et d'utiliser la sémantique du langage de l'ontologie.

IV.3.1. Pellet

Pellet [53] est un des projets du MINDSWAP Group, un groupe de recherche sur le web sémantique de l'université du Maryland. Il est disponible en OpenSource et offre des évolutions fréquentes. Pellet travaille sur des ontologies décrites en RDF ou OWL et permet les requêtes avec RDQL et SPARQL sur la ABox et la TBox.

Les points négatifs:

- ◆ Pellet possède une documentation pauvre comparée à celle de Racer.
- ◆ Actuellement Pellet ne permet pas l'utilisation de règles SWRL.
- ◆ Pellet n'offre pas de système de souscription à un concept.

Les atouts de Pellet :

- ◆ C'est un open-source et développé en Java.
- ◆ C'est un raisonneur OWL DL complet.
- ◆ Il propose en cas d'incohérence dans l'ontologie des réparations possibles, ainsi qu'une heuristique permettant d'obtenir les informations à ajouter dans l'ontologie pour passer au sous-langage OWL inférieur (OWL Full > OWL DL > OWL Lite).
- ◆ Pellet permet l'utilisation des types de données utilisateurs.

IV.3.2. Racer

Le système Racer (*Renamed ABox and Concept Expression Reasoner* ou raisonneur d'expression de concept et de ABox renommées) est un système de représentation de connaissance pour le calcul DL.

Racer [54] est le moteur d'inférence sans doute le plus connu et l'un des plus utilisés pour ses performances et sa stabilité. Racer travaille sur les ontologies modélisées par son langage, mais il accepte des ontologies décrites en RDF ou OWL, ces dernières étant traduites vers le langage utilisé par Racer. Ce moteur d'inférence possède également son propre langage de requête nRQL (*new Racerpro query Language*) pour interroger les ontologies sur la ABox et la TBox. Racer se présente sous la forme d'un serveur qui peut être accédé par le protocole TCP ou http.

Racer possède quelques atouts :

- ◆ La documentation sur Racer est importante, provenant des concepteurs et des utilisateurs.
- ◆ Racer permet l'utilisation d'un mécanisme d'abonnement à un concept qui permet d'être informé de la création de nouvelles instances de ce concept.
- ◆ Racer permet l'ajout d'assertions et d'individus dans les ABox après le chargement de l'ontologie.
- ◆ Racer permet l'utilisation de règles SWRL.

Racer possède quelques points négatifs:

- ◆ Racer suppose que toutes les propriétés sur les *datatypes* sont fonctionnelles (pas de valeurs multiples pour un *datatype property*).
- ◆ Racer ne permet pas l'utilisation de type de données utilisateur (type défini par l'utilisateur), car il possède ces propres types de données et il effectue une conversion avec les types de base.
- ◆ Racer est un produit commercial, il n'existe pas de version libre d'utilisation. Cependant il est possible d'obtenir une licence gratuite dans le cadre de la recherche scientifique.

Pour vérifier la consistance de notre ontologie, nous avons eu a eu recours au moteur d'inférence Pellet.

IV.4. Langage d'interrogation « SPARQL »

SPARQL [55] (protocole SPARQL et langage de requête RDF) en informatique est un langage de requête, devenu le 15 Janvier 2008, dans le cadre de l'activité Web sémantique du W3C, une recommandation W3C.

Le langage SPARQL définit la syntaxe des requêtes effectuées sur un graphe de données RDF. Inspiré du langage SQL, il se base sur les triplets contenus dans le graphe.

Une requête est composée de trois parties distinctes :

- Les PREFIX qui sont en fait les préfixes des URI des ressources à utiliser.
- La clause SELECT, semblable à celle du langage SQL, permet de définir les différentes ressources retournées par la requête.
- La clause WHERE, composée d'un ensemble des triplets, permet de définir les conditions dans la sélection.

Exemple :

```
SELECT ?subject ?object  
WHERE { ?subject ?object }
```

IV.5. Logiques de descriptions

Les logiques de description (LDs) découlent directement des travaux fondateurs de Bachmann et de son système KL-ONE. Depuis le début des années 90, la recherche en logique de description s'est considérablement développée.

Les logiques de description peuvent être considérées comme un fragment de la logique du premier ordre, dans lequel les formules ont une variable libre pour les descriptions de concepts et deux variables libres pour les descriptions de relations [56].

Une LD est composée de deux parties : un langage terminologique TBOX et un langage assertionnel ABOX. Le langage assertionnel est dédié à la description de faits et le langage terminologique à la description de concepts et de rôles. La principale tâche de raisonnement au niveau terminologique est de calculer les relations de subsomption entre concepts [57].

IV.6. Implémentation de l'ontologie

Pour implémenter notre ontologie, nous utiliserons l'éditeur Protégé 4.3. Cette implémentation se fait de la manière suivante :

◆ **Définition des concepts Héritant de Thing et des relations qui existent entre eux**

Ces concepts sont ceux présentés précédemment et sont : Organisation, Rôle, Donnée, Traiter_donnée et Manipuler_donnée. La figure IV.1 donne un aperçu de protégé

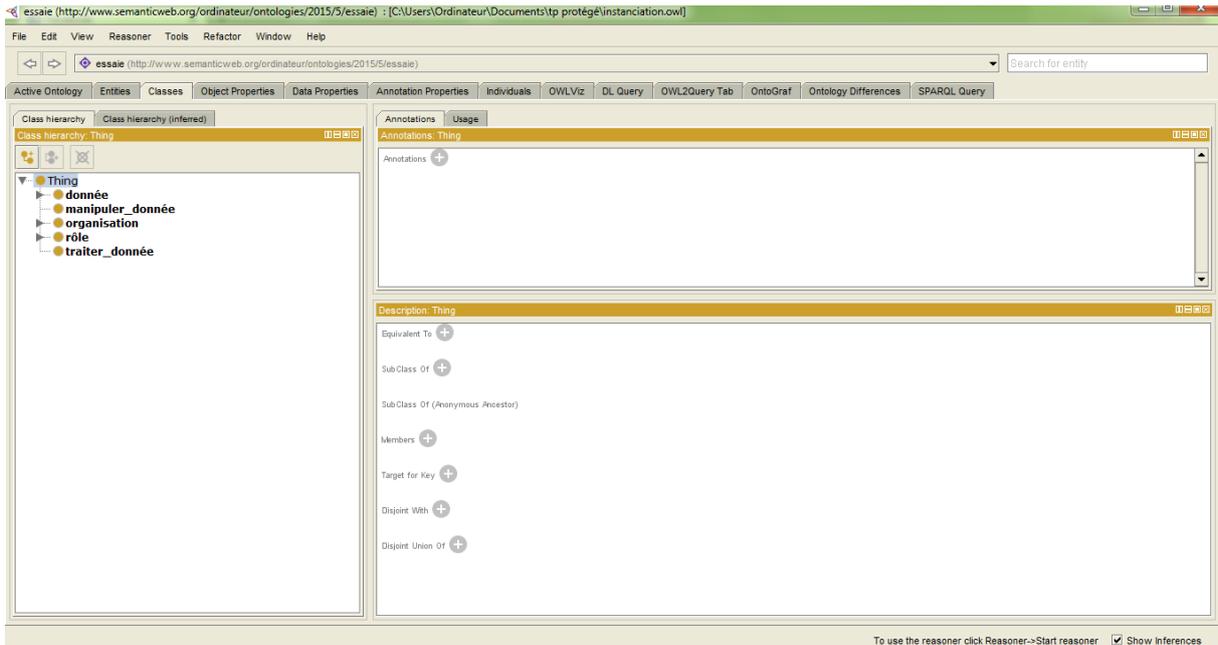


Figure IV.1 : Définition de la hiérarchie.

Après les concepts principaux, nous passons à la Hiérarchies des concepts fils. Chaque concept est relié à son concept père en utilisant la relation de subsumption « is-a ». Protégé permet de définir les sous-classes comme le montre la figure IV.2.

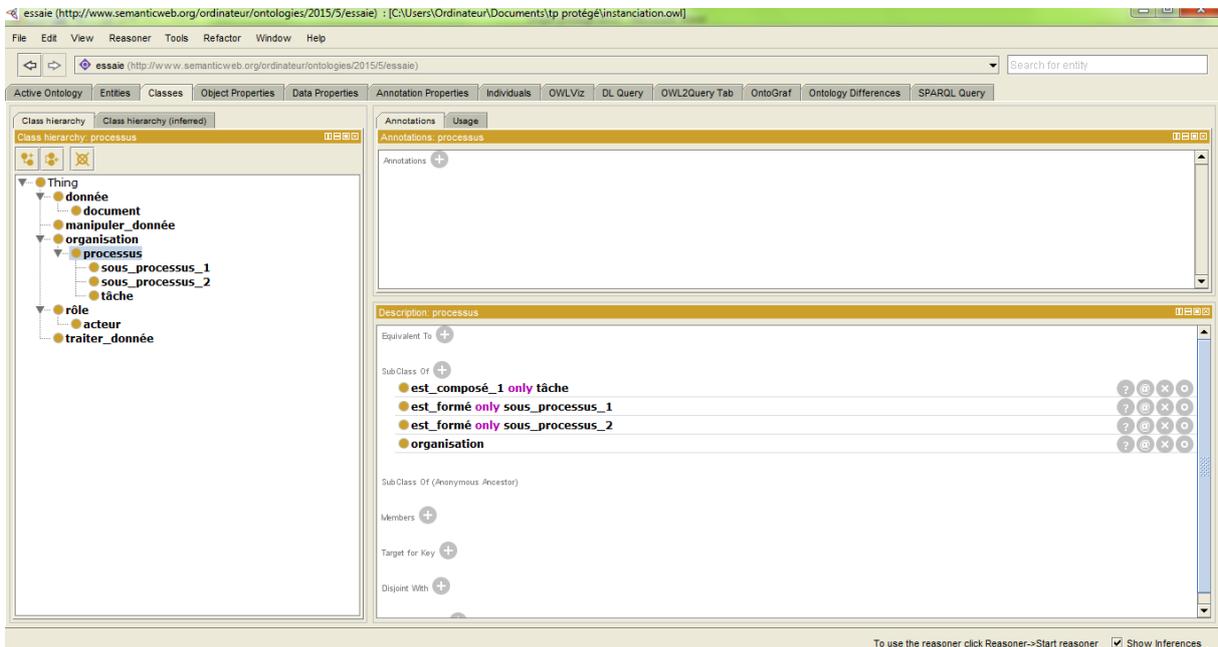


Figure IV.2 : Définition des hiérarchies des concepts de l'ontologie.

Une fois que nous avons implémenté tous les concepts de l'ontologie. Nous passons aux attributs de chaque concept. Par exemple, pour le concept Organisation, nous

pouvons ajouter un attribut « Nom_organisation » qui a comme instance « Service de cardiologie », comme illustré dans la figure IV.3.

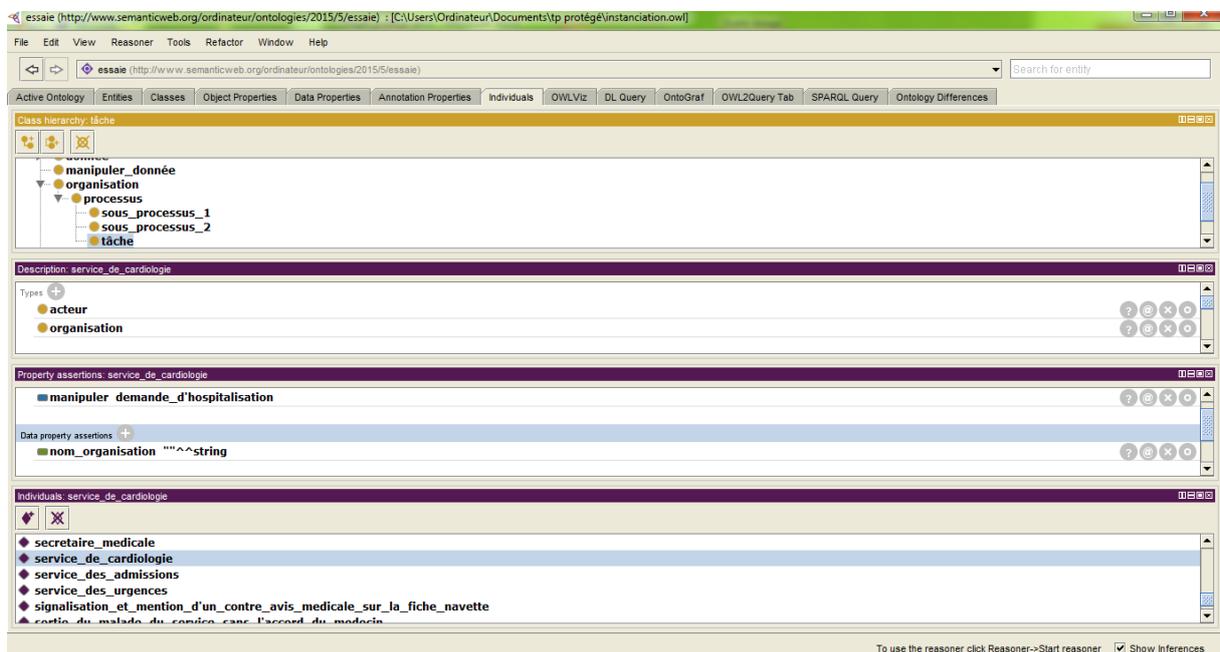


Figure IV.3 : Définition de l'attribut « Nom_organisation » du concept Organisation.

◆ Définition des relations binaires qui vont relier les concepts entre eux.

Les relations constituent un des éléments les plus importants dans une ontologie. Nous allons à présent implémenter les relations obtenues dans la formalisation. Dans protégé, les relations peuvent être définies en tant que *ObjectProperty*. Leur définition se fait en spécifiant le domaine « Domain » qui contient le concept de la relation, et le rang « Range » qui contient le concept cible de la relation. La figure IV.4 donne un exemple de définition de la relation « Manipuler ».

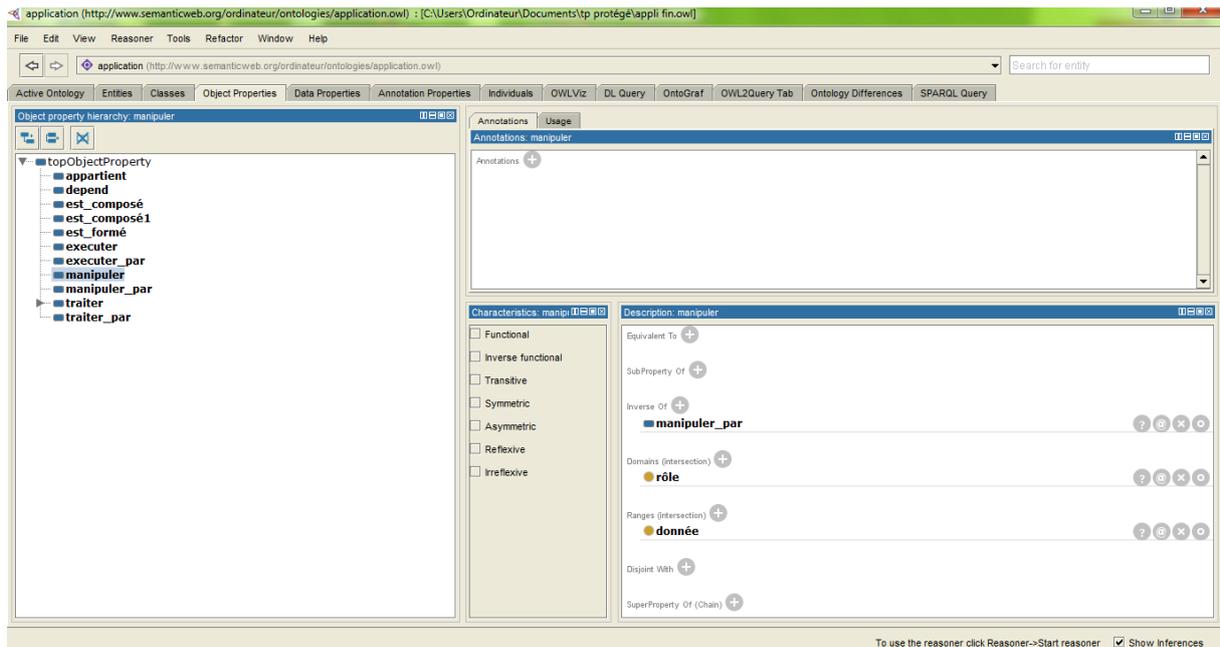


Figure IV.4: Définition des relations.

Pour avoir une ontologie consistante, il est nécessaire d'utiliser les restrictions sur les relations reliant les concepts. Cela permet d'effectuer des raisonnements sur le niveau terminologique TBox, comme illustré dans la figure IV.5

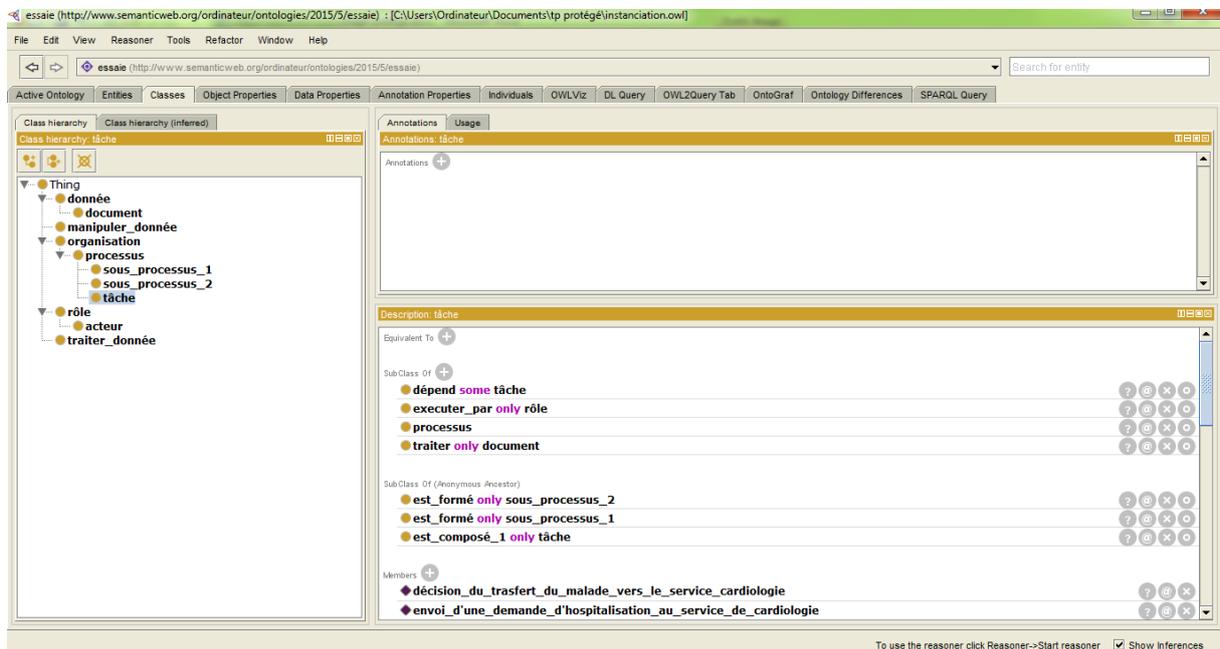


Figure IV.5 : Définition des restrictions.

IV.7. Visualisation de l'ontologie

Nous pouvons visualiser l'ontologie en utilisant deux outils : OWLViz et OntoGraf.

- **Visualisation de l'ontologie en utilisant l'OntoGraf**

OntoGraf fournit un schéma d'une partie de l'ontologie. Voici un extrait de l'ontologie avec OntoGraf.

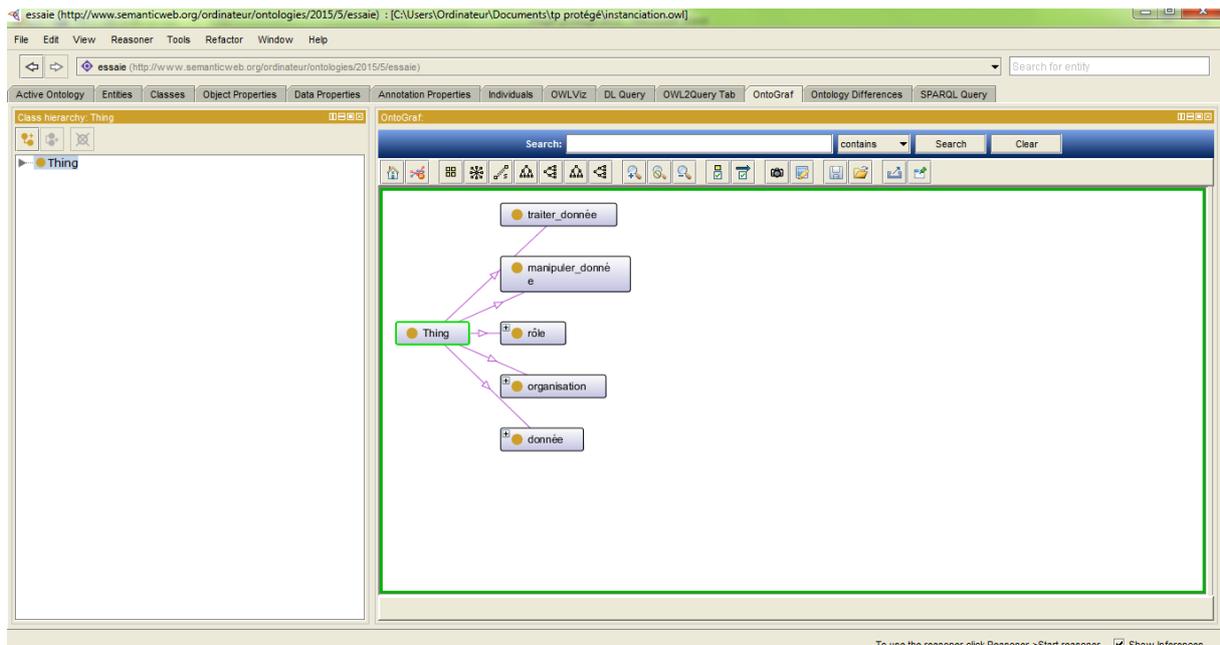


Figure IV.6 : Visualisation d'un extrait de l'ontologie avec OntoGraf.

OntoGraf permet de visualiser les concepts, les instances et les relations à la fois. La figure IV.7 donne un exemple où nous voulons visualiser l'instance « service de cardiologie » du concept « organisation ».

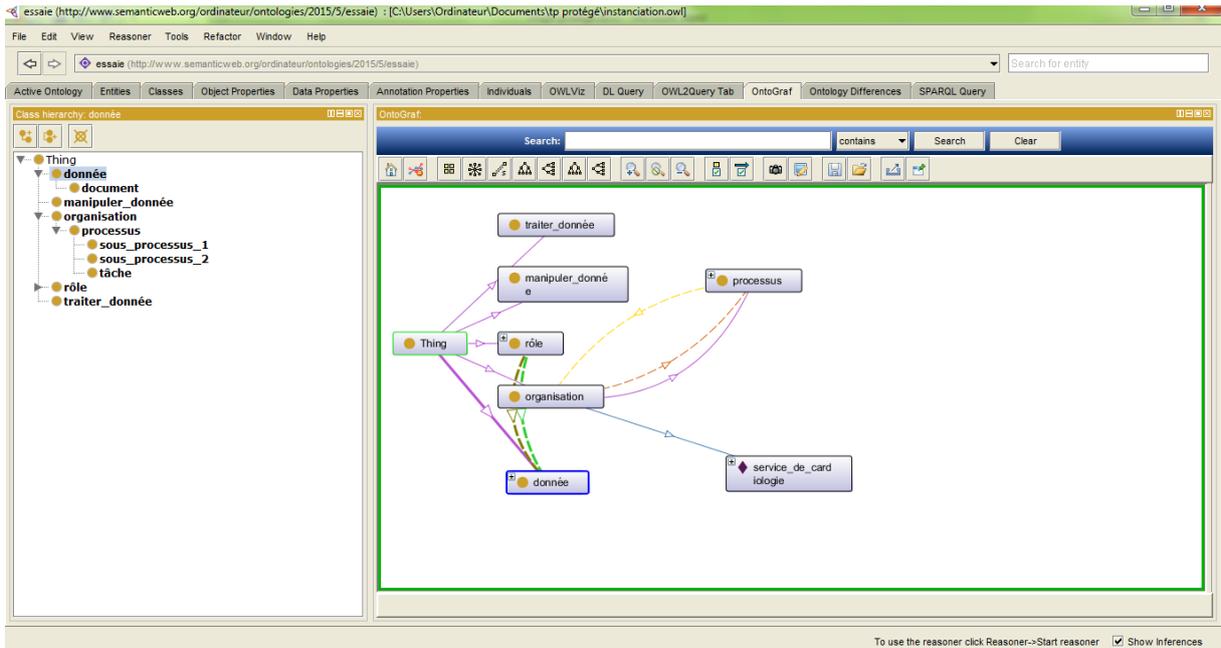


Figure IV.7 : Visualisation de l’instance « service_cardiologie » avec OntoGraf.

- Visualisation de l’ontologie en utilisant OWLViz

OWLViz fournit un schéma représentant l’ontologie complète ou une partie. Ce schéma est un arbre qui a comme nœud père « Thing » et comme nœuds fils les concepts principaux. Puis pour chaque concept, une hiérarchie lui sera reliée. Les figures IV.8 et IV.9 montrent la visualisation de l’ontologie en utilisant OWLViz.

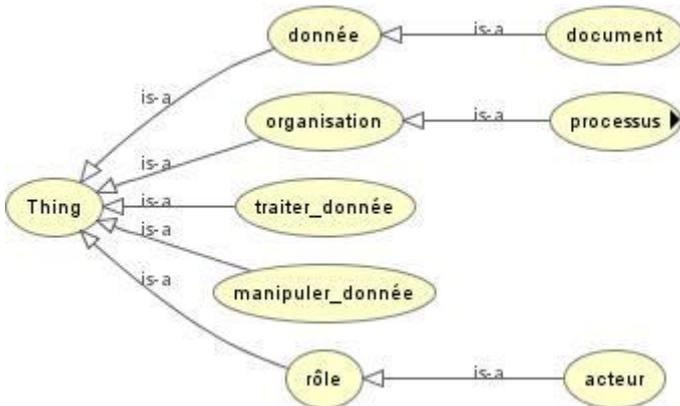


Figure IV.8 : Visualisation d’une partie de l’ontologie avec OWLViz

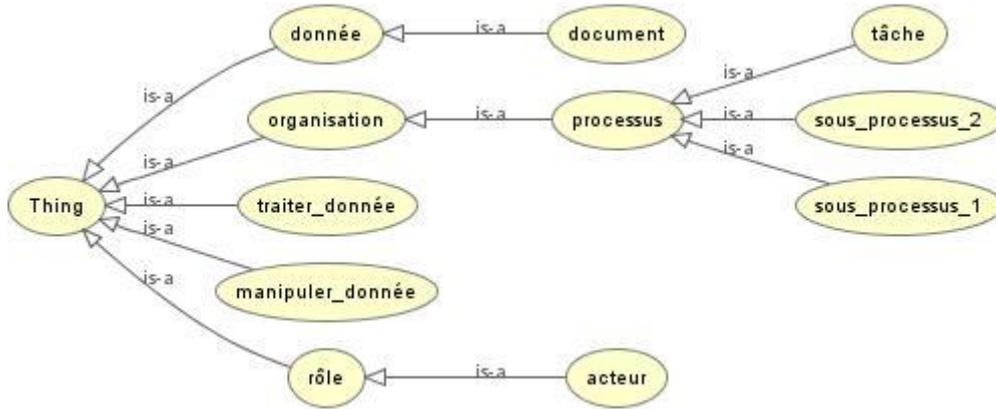


Figure IV.9: Visualisation de l'ontologie complète avec OWLViz.

IV.8. Interrogation de l'ontologie

Protégé propose un plugin qui permet d'exécuter des requêtes SPARQL directement sur l'ontologie. C'est « SPARQL Query ». nous allons exécuter des requêtes en utilisant le langage SPARQL.

Requête 1: Cette requête permet de récupérer tous les concepts qui sont reliés par la relation « SubClassOf », la relation de subsumption ou d'héritage.

SELECT ?subject ?object

WHERE { ?subject rdfs:subClassOf ?object }

Le résultat de la requête est illustré dans la figure IV.10 :

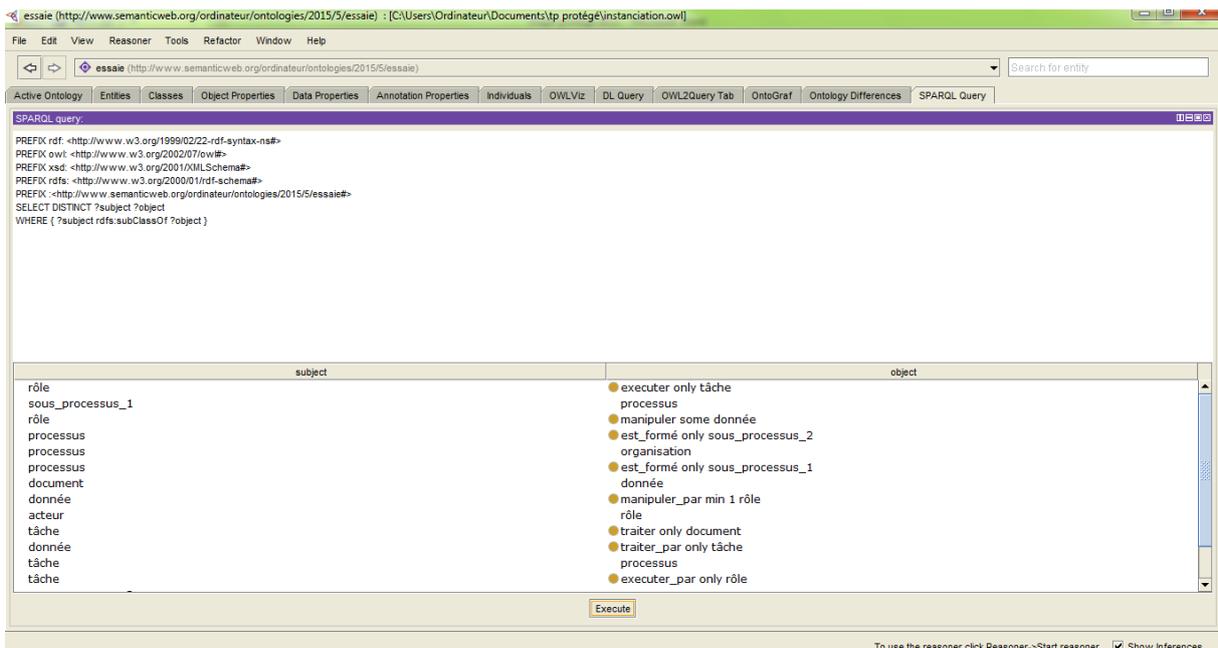


Figure IV.10 : Exemple 1 de l'exécution d'une requête SPARQL sur Protégé.

Requête 2 : cette requête permet de récupérer toutes les instances de notre ontologie.

```
SELECT ?sous_processus_1
WHERE {?sous_processus_1 rdf:type:sous_processus_1}
```

Le résultat de la requête est illustré dans la figure IV.11.

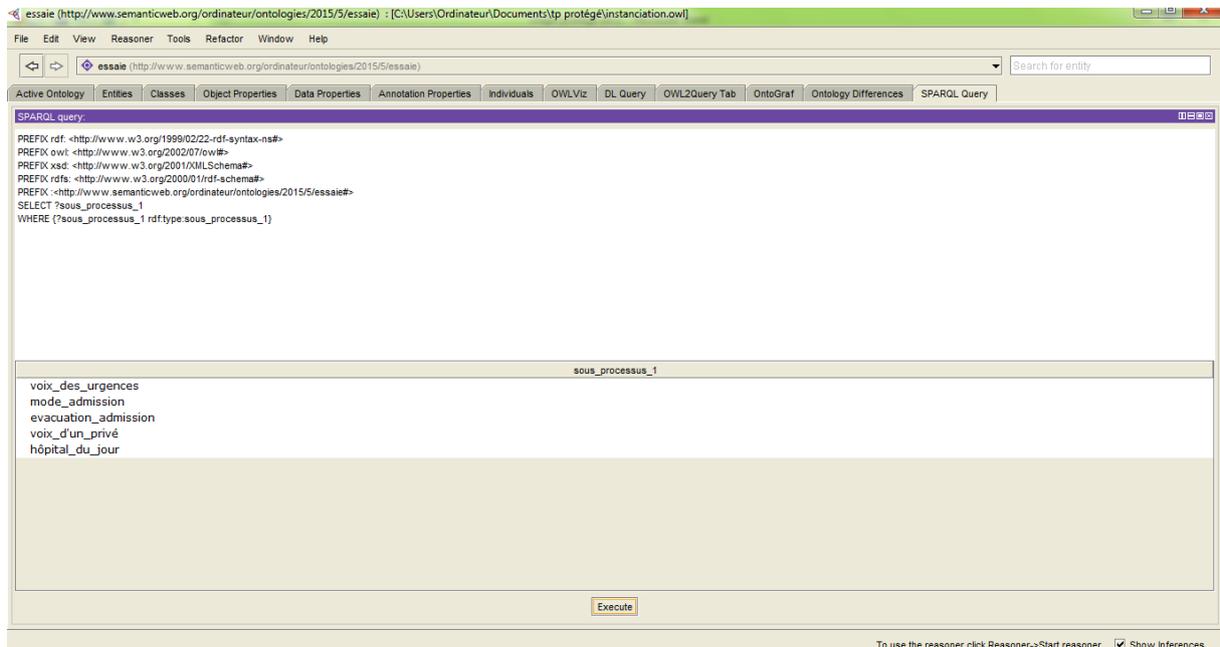


Figure IV.11 : Exemple 2 de l'exécution d'une requête SPARQL sur Protégé.

Requête 3 : cette requête permet de récupérer tous les attributs de notre ontologie.

```
SELECT DISTINCT ?attribut
WHERE{ ?attribut rdf:type owl:DatatypeProperty}
```

Le résultat de la requête est illustré dans la figure IV.12.

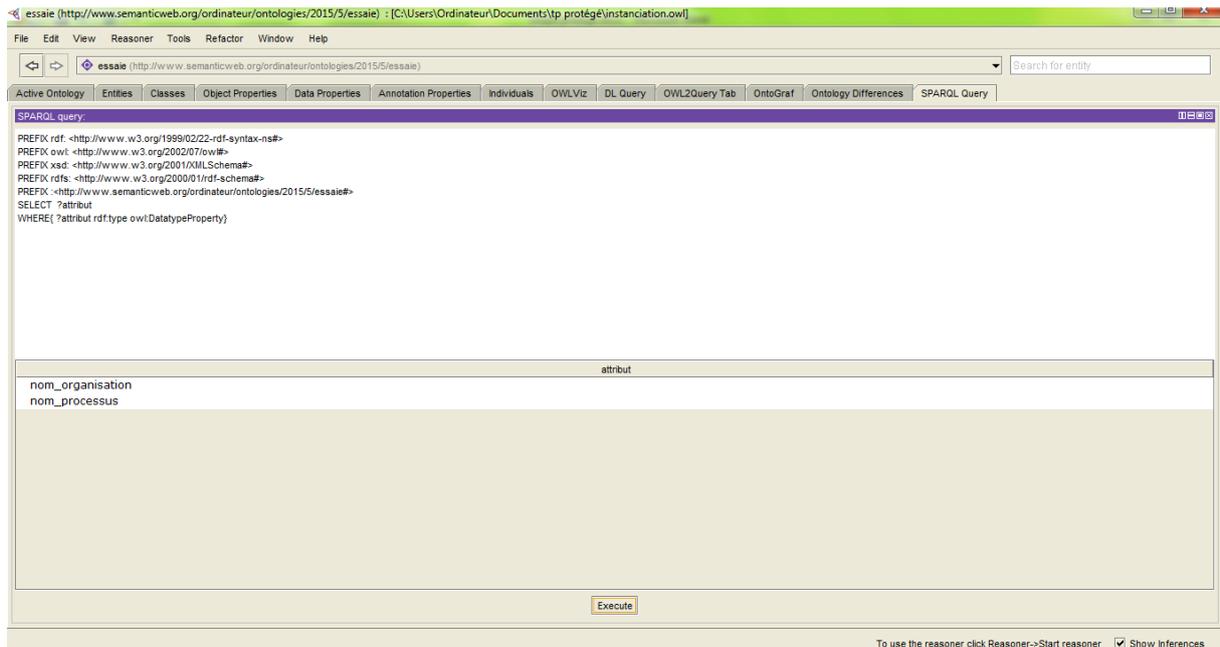


Figure IV.12 : Exemple 3 de l’exécution d’une requête SPARQL sur Protégé.

Requête 4 : cette requête précise quels sont les classes, les relations et les attributs.

SELECT ?subject ? Object
WHERE { ?subject rdf:type ? Object }

Le résultat de la requête est illustré dans la figure IV.13.

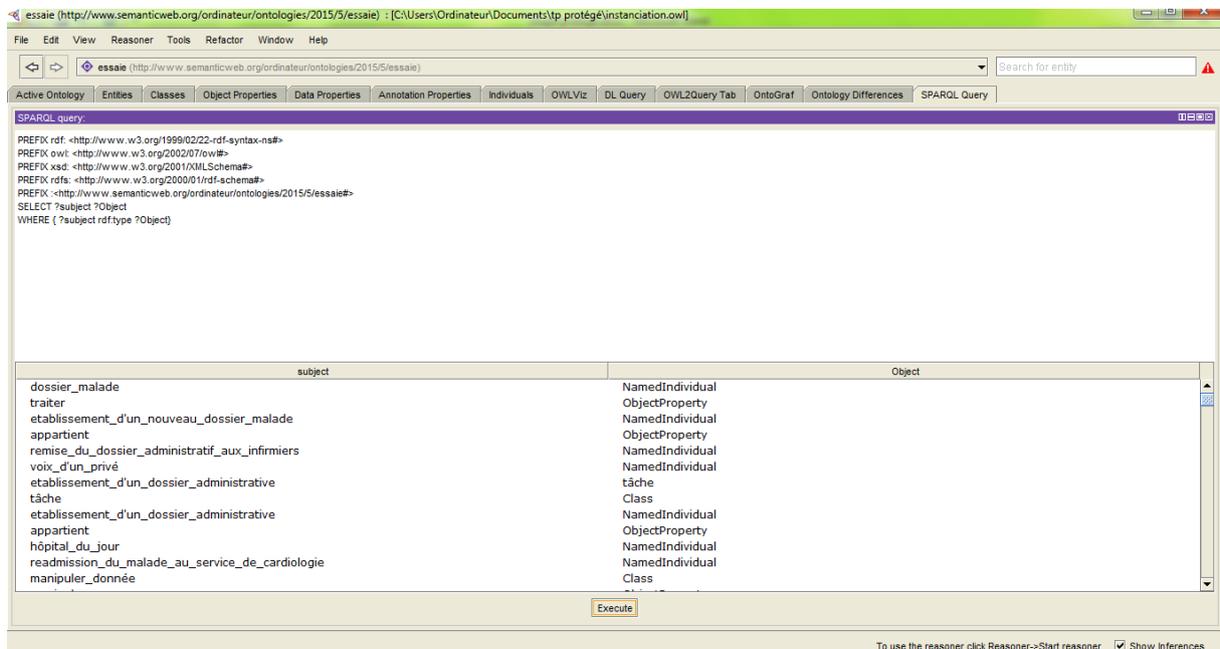


Figure IV.13: Exemple 4 de l’exécution d’une requête SPARQL sur Protégé.

Requête 5: cette requête permet de récupérer toutes les classes de notre ontologie

```
SELECT DISTINCT ?Classe  
WHERE {?Classe rdf:type owl:Class}
```

Le résultat de la requête est illustré dans la figure IV.14.

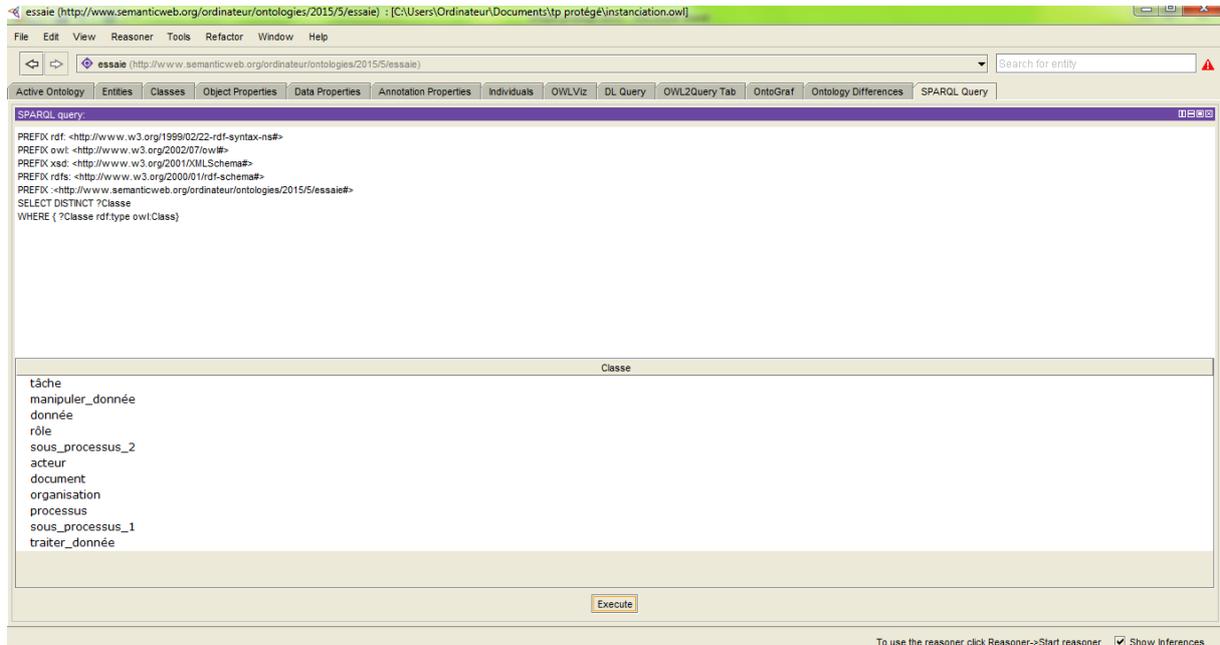


Figure IV.14: Exemple 5 de l'exécution d'une requête SPARQL sur Protégé.

Requête 6 : cette requête permet d'afficher qui manipule quelle donnée.

```
SELECT ?donnée ?object  
WHERE {?donnée :manipuler_par ?object}
```

Le résultat de la requête est illustré dans la figure IV.15.

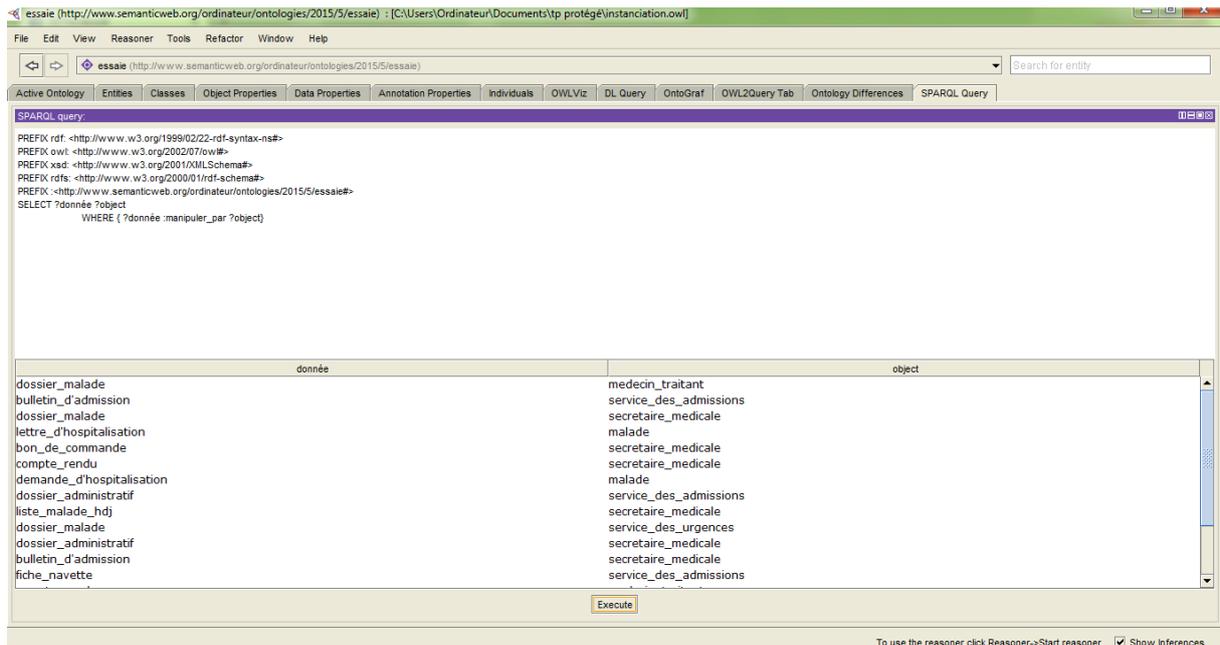


Figure IV.15: Exemple 6 de l'exécution d'une requête SPARQL sur Protégé.

Requête 7 : cette requête permet d'afficher qui execute quelle tâche.

Le résultat de la requête est illustré dans la figure IV.16.

SELECT ?tâche ?object
WHERE { ?tâche :executer_par ?object }

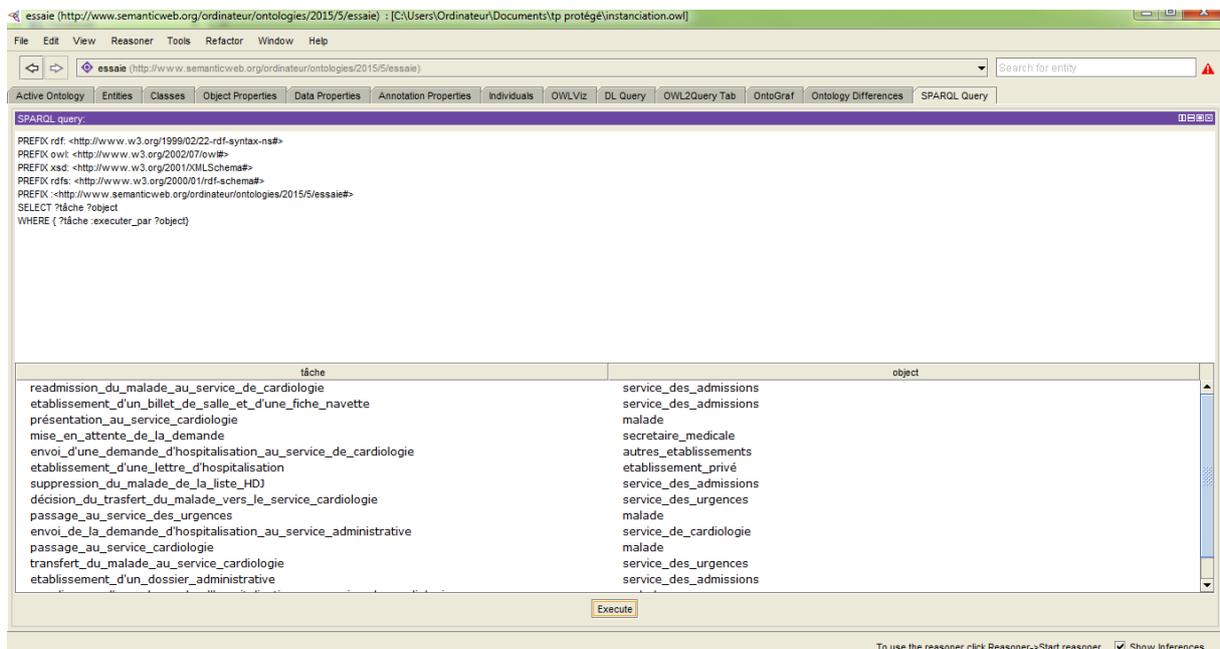


Figure IV.16: Exemple 7 de l'exécution d'une requête SPARQL sur Protégé.

Conclusion

Dans ce chapitre nous avons présenté en détail l'implémentation de notre ontologie. Nous sommes passés d'une ontologie formelle à une ontologie consistante. Et pour cela nous avons utilisé l'éditeur d'ontologie protégé. Nous avons d'abord commencé par la définition des concepts, attributs et relations. Puis nous avons défini toutes les instances de notre domaine et nous les avons reliées entre elles en instanciant les relations définies précédemment. Puis, nous avons testé la consistance de l'ontologie obtenue en utilisant le raisonneur Pellet.

A la fin nous nous sommes intéressés aux requêtes SPARQL afin d'interroger notre ontologie en utilisant « SPARQL Query ». Cette dernière peut être enrichie avec de nouveaux concepts.

*Conclusion
Générale*

Conclusion Générale & Perspectives

Learn from yesterday, live for today, hope for tomorrow. The important thing is not stop questioning »

Albert Einstein

Dans ce travail, nous avons commencé avec la définition des ontologies, leur cycle de vie, les types d'ontologies. Nous avons aussi énuméré les composants de l'ontologie ainsi que les domaines d'application. Ensuite, nous avons présenté l'ontologie de madame Sini[22], expliquer les règles de gestion respectées par cette dernière et procédé à sa conceptualisation. De plus, nous avons défini les processus, les types de processus et nous nous sommes intéressés de plus près au processus métier et leur gestion, on a présenté le cas réel que nous avons instancié.

Enfin, dans le dernier chapitre on a développé l'aspect pratique proprement dit. Nous avons mis l'accent sur la réalisation, l'inférence et l'interrogation de l'ontologie par des requêtes.

A la fin de ce travail, nous avons obtenu une ontologie consistante que nous avons testée en l'interrogeant. Cela nous conduit à dire que bien que l'ontologie est restreinte elle est suffisante pour décrire n'importe quel processus métier.

Comme perspectives du travail réalisé dans ce mémoire :

- ✚ Compléter et enrichir l'ontologie avec de nouveaux concepts,
- ✚ Enrichir notre ontologie par des règles SWRL, afin d'atteindre un niveau de formalisme plus complexe,
- ✚ Faire une analyse plus détaillé concernant les processus, et donner plus d'importance à la spécification des besoins,
- ✚ Soumettre l'ontologie à des contraintes et au changement du monde réel, et la mettre à disposition des experts du domaine, pour l'évaluer.

Bibliographie



- [1]: Natalya F. Noy et Deborah L. McGuinness, développement d'une ontologie 101 : guide pour la création de votre première ontologie". *Stanford Knowledge Systems Laboratory Technical Report*, Mars 2001.
- [2]: R. Neches, R.E. Fikes, T. Finin, T. Gruber, T. Senator, W.R. Swartout, "Enabling technology for knowledge sharing", *AI Magazine*, 1991.
- [3]: T. Gruber, "A translation approach to portable ontology specification", 1993
- [4]: W. N. Borst, "Construction of engineering ontologies". University of Twente, Enschede, Centre for Telematica and Information Technology, 1997.
- [5]: Dien et al. 01, DIENG R., CORBY O., GANDON F., GIBOIN A., GOLEBIEWSKA J., MATTA N. et RIB IERE M., «Méthodes et outils pour la gestion des connaissances : une approche pluridisciplinaire du knowledge Management ». Dunod, 2 edition. 2001.
- [6]: G. Van Heijst, Ath. Schreiber, BJ. Wielinga. : Using explicit ontologies in KBS
- [7]: N. Guarino.: Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy, pp. 3-15, 1998.
- [8]: R. Mizoguchi, J. Vanwelkenhuysen, M. Ikeda.: Task Ontology for reuse of problem solving knowledge. In: Mars N (ed) Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing (KBKS'95). University of Twente, Enschede, The Netherlands. IOS Press, Amsterdam, The Netherlands, pp 46-57,1995.
- [9]: Uschold, M., & Grüninger, M. Ontologies: Principles, Methods and applications.*Knowledge Engineering Review* , 11 (2), 93-155, 1996.
- [10]: Psyché, V., Mendes, O., & Bourdeau, J. Apport de l'ingénierie ontologique aux environnements de formation à distance, 2003.
- [11]: M. Uschold, M. King. Towards a methodology for building ontologies, in *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI'95*, 1995.

[12]: Gomez Pérez A., Benjamins V.R. “*Overview of Knowledge Sharing and Reuse Components: Ontologies and problem-Solving Methods*”. Proceeding of the IJCAI-99 workshop on Ontologies and problem-Solving Methods (KRR5), Stockholm (Suède), pp. 1.1-1.15, 1999.

[13]: Thibault Mondary « Construction d’ontologies à partir de textes l’apport de l’analyse de concepts formels », université paris 13, 2011.

[14]: Farquhar (A.), Fikes (R.), Rice (J.): «*The Ontolingua Server: A Tool for Collaborative Ontology Construction*», *Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Alberta, Canada, p. 44.1-44.19, 1996.

[15]: B. Swartout, P. Ramesh, K. Knight and T. Russ, “*Toward Distributed Use of Large-Scale Ontologies*”, In Symposium on Ontological Engineering of AAAI, Stanford, California, March, 1997.

[16]: OILED, OIL Editor Home Page, [http : //oiled.man.ac.uk/](http://oiled.man.ac.uk/), 2004.

[17]:PROTEGE2000,Protege2000OntologyEditorHomePage,2002
<http://protege.stanford.edu/>

[18]: Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer and Wenke, D., “*OntoEdit: Collaborative Ontology Engineering for the Semantic Web*”. In Proceedings of the International Semantic Web Conference 2002 (ISWC 2002), Sardinia, Italia, June 2002.

[19]: Azouaou F., Cao TD., Dehors S., Desmoulins C., Dieng-Kuntz R., Faron-Zucker C., Luong PH., (2005). « Les Outils du Web sémantique et du E-Learning », La Journée thématique « Web sémantique pour le E-learning » journée finale de l’Action Spécifique du CNRS & COLORS WebLearn, dans le cadre de la plate-forme AFIA’2005. Nice le 31 mai 2005.

[20]: HADJOUI Fatima Zohra, « Gestion des Ontologies Médicales (G.O.M) », Université Abou Bakr Belkaid– Tlemcen, 2011-2012.

[21]: M. Uschold and M.Gruninger, “*Creating semantically integrated communities on the World Wide Web*”. Honolulu: Semantic Web Workshop, 2002.

[22]: Sini, G. Comparot, C. Si Mohammed, M. « MASP A Model for Business processus Analysis and Specification », International Journal of Advanced Research in Computer Volume, No. 3, May-June 2012.

[23]: <http://www.techno-science.net/?onglet=glossaire&definition=762>

[24]: W, van der Aalst, K. van Hee. Workflow Management: Models, Methods and Systems. MIT Press 2002.

[25]: Rodriguez S., Gaud N., Hilaire V., Galland S., and Koukam A. "An analysis and design concept for self-organization in Holonic Multi-Agent Systems" In Lecture Notes in Artificial Intelligence, Engineering Self-Organizing Applications, number 4335, pp.15-27. Eds. Brueckner, Sven and Hassas, Salima and Jelasity, Mark and Yamins, Daniel.

[26]: <http://fr.wikipedia.org/wiki/Donnée>

[27]: G. SINI, M. SI-MOHAMMED, «Progressive knowledge acquisition for the processus formalization. », Proceedings of the 2010 International Joint Conferences on e-CASE and e-Technology, January 25-27.2010

[28]: F. Vernadat, "Techniques de modélisation en entreprise : Applications aux processus Opérationnels", economica 1999.

[29]: Norme européenne NF EN ISO 9001 version 2000, *Systèmes de management de la qualité – Exigences*, AFNOR, 2000.

[30]: C. Morley, "La modélisation des processus : typologie et proposition utilisant UML", *Processus et Systèmes d'information – Journées ADELI*, Paris, France, 2002.

[31]: F. Théroude, "Formalisme et système pour la représentation et la mise en œuvre des processus de pilotage des relations entre donneurs d'ordre et fournisseurs", thèse de Doctorat de l'Institut National Polytechnique de Grenoble, 2002.

[32]: Sparx systems « the business process model », UML tutorials, 2004.

[33]: M. Catton, "Management des processus, une approche innovante", Paris : Afnor, 2000.

[34]: The Workflow Management Coalition, « Final XPDL 2.1 Specification », dans le Rapport de spécification numéro WFMC-TC-1025-Oct-10-08-A, 2008.

[35]: Morley, C., Hugues, J., Leblanc, B., & Hugues, O. 2007. *Processus métiers et S.I. – Evaluation, modélisation, mise en œuvre*, DUNOD.

[36]: Equipe Conseil Softeam supervisé par Philippe Desfray, « le guide pratique des processus métiers », version 1.0, www.softeam.fr

[37]: Realizing e-Business with component, Paul Allen, Addison Wesley 2001.

[38]: Van der Aalst, Wil M. P, ter Hofstede, Arthur H.M, Weske, Mathias, «Business Process Management: A Survey». Dans BPM 2003, LNCS 2678, pp. 1–12, 2003. Springer-Verlag Berlin Heidelberg 2003

[39]: J.L Le Moigne, "*La modélisation des systèmes complexes*", Afcet-systèmes, Dunod, 1990.

[40]: G.E.P Box, "Robustness in scientific model building," Robustness in statistics", In R.L. Launer and G.N. Wilkinson (Editors), Academic Press, New York, 1979, pp. 201–236

[41]: <http://www.piloter.org/process-management/modelisation-processus.htm>

[42]: Jacques Sassoon, « Pratiques de l'urbanisme des systèmes d'information en entreprises », édition 2003

[43]: N. Bradley, The {XML} Companion, Addison-Wesley Professional Publisher, 2001.

[44]: D.C. Fallside, XMLSchema, World Wide Web Consortium (W3C), W3C Recommendation, <http://www.w3.org/XML/Schema>, 2001.

[45]: Xavier Lacot, « introduction à l'owl, un langage XML d'ontologie web »,2005.

[46]: <http://www.w3.org/TR/rdf-schema/>

[47]: Fensel D., Horrocks I., van Harmelen F. et De S., 2000. Oil in a nutshell. In Proceedings of European Knowledge Acquisition Workshop (EKAW'2000), 1937: 1-16.

[48]: Hendler J. et McGuinness D., 2001. The Darpa Agent Markup Language. <http://www.daml.org>.

[49]: T. Berners-Lee, J. Hendler and O. Lassila (2001). The Semantic Web. Scientific American Magazine, 284(5), 34-43.

[50]: D. L. McGuinness, F. van Harmelen (2004). Owl web ontology language overview World Wide Web Consortium, Recommendation REC-owlfeatures-20040210.

[51]: Frank van Harmelen and Deborah L. McGuinness. OWL web ontology language overview. W3C recommendation, W3C, 10 February 2004. <http://www.w3.org/TR/2004/REC-owl-features-20040210/>

[52]: <http://www.w3.org/TR/owl-features/>

[53]: E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur and Y. Katz, *Pellet: A Practical OWL-DL Reasoner*, University of Maryland, 2005.

[54]: V. Haarslev, R. Moller and M. Wessel, *RACER User's Guide and Reference Manual, version 1.6. Technical report*, University of Hamburg, Computer Science Department, 2001.

[55]: <http://www.w3.org/TR/rdf-sparql-query/>

[56]: M. Buchheit, f. Donini, and A. Shaerf "Decidable reasoning in terminological knowledge representation systems". *Journal of artificial intelligence research*, 1: 109-138, 1993.

[57]: A. Napoli, "une introduction aux logiques de descriptions" N° 3314, Décembre 1997.