



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

جامعة وهران 2 محمد بن أحمد
Université d'Oran 2 Mohamed Ben Ahmed

معهد الصيانة والأمن الصناعي
Institut de Maintenance et de Sécurité Industrielle

Département De Maintenance en Instrumentation

MÉMOIRE

Pour l'obtention du diplôme de Master

Filière : Génie Industriel
Spécialité : Génie Industriel

Thème

**Pilotage de performances dans un service
hospitalier par les techniques du web
sémantique**

Présenté et soutenu publiquement par :

KADA BENABDALLAH Aya.

Et

BELAKHDAR Abdelkarim.

Devant le jury composé de :

Nom et Prénom	Grade	Etablissement	Qualité
GUETARNI Islem Hadj Mohamed	MCB	IMSI-Université D'Oran 2	Président
TITAH Mawloud	MAA	IMSI-Université D'Oran 2	Encadreur
HACHEMI Khalid	Pr	IMSI-Université D'Oran 2	Co-encadreur
REGUIEG Yassaad Sadek	MAA	IMSI-Université D'Oran 2	Examineur

Année 2019/2020

Remerciements

Ce travail est l'aboutissement d'un dur labeur et de beaucoup de sacrifices; nos remerciements vont d'abord au bon DIEU le Créateur de l'univers qui nous a doté d'intelligence, et nous a maintenu en santé pour mener à bien cette année d'étude.

Nous offrons premièrement de sincères et chaleureux remerciements à notre encadreur de recherche **Mr TITAH Mawloud**. La confiance qu'il nous a accordée ainsi que le soutien moral. Nous saluons son amabilité, sa patience, sa disponibilité, sa souplesse d'esprit et son savoir. C'est certes avec joie et fierté que nous déposons aujourd'hui ce mémoire, mais aussi avec un brin de nostalgie que nous terminons ce programme d'études.

Nous tenons également à exprimer notre gratitude à **Pr HACHEMI Khalid, professeur** au département de la maintenance en instrumentation, qui s'est montré très patient comme un Co-encadreur.

Nous adressons aussi des remerciements à **Mr GUETARNI Islem Hadj Mohamed**, doctorant au département de l'hygiène et la sécurité industrielle qui a accepté de présider le jury de soutenance, pour tout ce qu'il a pu nous apprendre ; qu'il trouve ici l'expression de notre profonde et sincère reconnaissance.

Nous dédions un merci particulier à **Mr REGUIEG Yassaad** pour nous avoir fait l'honneur d'accepter d'examiner ce travail.

Un grand merci à toute l'équipe de service Covid au niveau d'**EHU** Oran .

Enfin, nous ne pouvons passer outre notre reconnaissance envers nos parents, nos sœurs et frères. Leur présence, leur écoute, leur confiance en nous et leur soutien constant nous assurent des bases solides nous permettant de persévérer et de nous surpasser.

Dédicace

Gloire soit rendu au Dieu tout puissant, le très miséricordieux pour tous ses bienfaits dont il m'a comblé et de m'avoir donné le courage et la force pour réaliser ce modeste travail que je dédie à :

Les deux bougies qui s'allument que pour éclaircir mon chemin : Mes très chers parents.

Mes très chères sœurs Anfel, Soumia, et Takwa.

A mon précieux frère Rayen.

A mon âme-sœur Nadour.N.

A ma copine au cœur en or Nesrine .

A mon binôme Belakhdar.K et mes chers amis dont je ne saurais citer car la liste est longue.

Aya.

Dédicace

Tout d'abord, je tiens à remercier DIEU De m'avoir donné la force et le courage de mener à bien ce modeste travail.

Je tiens à dédier cet humble travail à :

A la mémoire de ma mère que dieux l'accueillit dans son vaste paradis.

A ma deuxième mère, qui m'a entouré d'amour, d'affection et qui fait tout pour ma réussite, Ma grand-mère Yamin, que Dieu la protège.

Mon très cher père qui m'a aidé de devenir ce que je suis aujourd'hui.

Mes bien chères sœurs Amel et Djemila.

Mon bras droit tout au long de ce travaille Manel.

A mon binôme Aya et mes chers amis ,Amine,Brahim et Nesrine et tous les autres qui on était la pour moi.

Karim.

Résumé

Un grand nombre de langages de représentation des connaissances apparus offrant différents niveaux d'expressivité et de raisonnement. Le langage d'ontologie Web OWL a été recommandé par le W3C, car il ajoute une puissance d'expressivité considérable au web sémantique. Mais, OWL a des restrictions expressives qui sont très lourdes dans certaines applications.

Établir un diagnostic médical fiable requiert l'identification de la maladie d'un patient sur la base de l'observation de ses signes et symptômes. Par ailleurs, les ontologies constituent un formalisme adéquat et performant de représentation des connaissances biomédicales. Cependant, les ontologies classiques ne permettent pas de représenter les connaissances liées au processus du diagnostic médical : connaissances probabilistes et connaissances imprécises et vagues.

Nous proposons des méthodes générales de représentation des connaissances afin de construire HDO. Ces méthodes permettent de représenter :

Les connaissances imprécises et vagues par la discrétisation des concepts.

Les connaissances probabilistes (les sensibilités et les spécificités des signes pour les maladies, et les prévalences des maladies pour une population donnée).

Les connaissances liées au processus du diagnostic médical par des règles SWRL.

Un moteur d'inférences adductif et probabiliste a été conçu et développé. Ces méthodes ont été testées à l'aide de dossiers patients réels.

Dans ce travail, nous présentons un processus de développement d'une ontologie OWL enrichie par les règles SWRL. Le domaine choisi est médical. Pour raisonner sur cette ontologie, nous avons utilisé les outils standards du web sémantique : protégé-OWL, et Hermit sur les règles SWRL.

Mot clés : web sémantique, ontologie , OWL, règles SWRL, W3C, HDO, protégé.

Abstract

Many knowledge representation languages appeared with different levels of expressiveness and reasoning. The Web Ontology Language OWL has been recommended by the W3C, it adds considerable expressive power for the semantic web. However, OWL has expressive limitations that are very heavy in some applications.

Making a reliable medical diagnosis requires the identification of the patient's disease based on the observation of signs. Moreover, ontologies provide an adequate and efficient formalism for medical knowledge representation. However, classical ontologies do not allow representing knowledge associated with medical reasoning such as probabilistic, imprecise, or vague knowledge.

In the current work, general knowledge representation methods are proposed. They aim at building HDO. They allow representing:

Imprecise or vague knowledge by discretizing concepts. Probabilistic knowledge (sensitivity, specificity and prevalence).

medical reasoning and reasoning on the absent signs by SWRL rules. An abductive reasoning engine and a probabilistic reasoning engine were designed and implemented. The methods were evaluated by use of real patient records.

In this work, we present a process of developing an OWL ontology enriched by SWRL rules. The medical field is selected. To reason about this ontology, we used standard tools of the Semantic Web: Protégé-OWL and Hermit for SWRL rules

Key words: Semantic Web, Ontology, OWL, SWRL rules, W3C, HDO.

Liste des Abréviations

OWL : Ontology Web Langage.

RDF: Resource Description Framework.

SWRL: Semantic Web Rule Language.

SQWRL: sematic query enhanced web rule language.

XML: eXtensible Markup Langage.

SPRQL: sparql protocol and RDF query language.

HDO: Hospital Department Ontology.

DL: Description Logic.

W3C: World Wide Web Consortium.

RDF(S): Resource Description Framework Schema.

URI: Uniform Resource Identifiers.

URL: Uniform Resource Locator.

RIF: Rule InterchangeFormat.

RuleML: Rule Markup Language.

ARN: AcideRibonucléique.

ORL : Oto-Rhino-Laryngologie.

VNI : Ventilation Non Invasive.

SAMU : Service Aide Médicale Urgent.

JESS : JavaExpert System Shell.

OMS : Organisation Mondiale de la Santé.

SIM : Systèmed'InformationHospitalier.

PCR : Polymerase Chain Reaction.

Sommaire

INTRODUCTION GENERALE	3
INTRODUCTION.....	6
LE WEB SEMANTIQUE.....	6
ARCHITECTURE ET LANGAGES DU WEB SEMANTIQUE	7
<i>La couche URI et UNICODE</i>	9
<i>La couche XML</i>	9
<i>La couche RDF</i>	10
<i>La couche ontologie</i>	10
<i>La couche Logique</i>	10
<i>Couche règles</i>	10
LOGIQUES DE DESCRIPTION.....	11
L'INTERPRETATION SELON L'HYPOTHESE DU MONDE OUVERT	13
<i>Couche preuve et confiance</i>	13
LES LANGAGES DU WEB SEMANTIQUE	14
XML (eXtensibleMarkupLanguage).....	14
RDF	14
RDFS (Resource Description Framework Schéma) –méta-modèle	15
Langage OWL	16
RIF (Rule Interchange Format).....	18
INGENIERIE ONTOLOGIQUE	19
CONSTITUANTS D'UNE ONTOLOGIE	19
Concept	20
Relation	21
Axiome	21
Instance	22
UTILISATION DES ONTOLOGIES	22
<i>La communication</i>	22
<i>L'interopérabilité entre les systèmes</i>	22
<i>Les typologies d'ontologies</i>	23
CONSTRUCTION D'ONTOLOGIE	24
<i>La conceptualisation</i>	24
<i>L'ontologisation</i>	24
RAISONNEMENT SEMANTIQUE SUR UNE ONTOLOGIE OWL	25
INFERENCE DANS LES LOGIQUES DE DESCRIPTION	25
<i>Au niveau factuel (ABox)</i>	27
OUTILS POUR LE WEB SEMANTIQUE	28
EDITEURS D'ONTOLOGIE	28
1.12.1Protégé.....	28
INTEGRATION D'UN LANGAGE DE REGLE AVEC UN LANGAGE D'ONTOLOGIE	28
APPROCHE HOMOGENE	29
LANGAGE SWRL	29

EDITION DES REGLES SWRL	30
14 . CONCLUSION.....	31
2 .Chapitre 2.....	32
<i>Conception d’une ontologie à base des services Hospitalier.....</i>	32
INTRODUCTION	33
	OBJECTIF 34
ARCHITECTURE GENERALE DU SYSTEME.....	34
FONCTIONNEMENT DU SYSTEME.....	36
CONCEPTION D’UNE ONTOLOGIE DANS LE DOMAINE MEDICAL.....	37
CYCLE DE VIE D’UNE ONTOLOGIE	37
METHODOLOGIES DE CONSTRUCTION D’ONTOLOGIES	38
VERS UN PROCESSUS DE CONSTRUCTION D’UNE ONTOLOGIE OWL	39
<i>Spécification</i>	39
	PROCESSUS 40
CONCEPTUALISATION	41
FORMALISATION	41
2.8.1 <i>Partie terminologiqueTBox.....</i>	42
2.8.2 <i>Partie assertion elle ABox.....</i>	43
IMPLEMENTATION	43
TESTS ET EVOLUTION	43
LES ONTOLOGIES EXISTANTES DANS LE DOMAINE MEDICAL.....	44
LE BESOIN D’UNE ONTOLOGIE HOSPITALIER.....	45
CONSTRUCTION DE L’ONTOLOGIE HDO DANS LE DOMAINE HOSPITALIER.....	46
DOMAINE D’APPLICATION	46
UNE CONSULTATION MEDICALE.....	47
	<i>Spécification 49</i>
CONCEPTUALISATION	50
<i>Construction de glossaire de termes.....</i>	50
<i>Classification de concepts</i>	54
<i>Dictionnaire de concepts.....</i>	56
<i>Tableau de relations binaires</i>	59
<i>Tableau des attributs.....</i>	60
CONCLUSION.....	62
ETUDE DE CAS	64
DOMAINE D’APPLICATION : COVID-19.....	64
MODES DE TRANSMISSION	64
SYMPTOMES ET SIGNES EVOCATEURS	64
PARCOURS DE LA PRISE EN CHARGE DU PATIENT POTENTIELLEMENT ATTEINT PARLE COVID-19	65

CONCEPTUALISATION.....	69
<i>Dictionnaire de concepts (COVID—19)</i>	<i>70</i>
<i>Tableau de relations binaires (COVID-19)</i>	<i>71</i>
REPRESENTATION SEMANTIQUE DES REGLES AVEC LE LANGAGE SWRL	72
<i>Parcours d'un Patient consultant en Première Ligne</i>	<i>72</i>
<i>Parcours d'un Patient consultant au urgences</i>	<i>73</i>
<i>Regles decisionnel COVID-19.....</i>	<i>73</i>
SEMANTIC QUERY ENHANCED WEB RULE LANGUAGE: SQWRL.....	75
IMPLEMENTATION DE L'ONTOLOGIE	75
OUTILS D'IMPLEMENTATION	76
PROTEGE ET PLUGINS	76
	<i>Plugins utilisés</i>
	77
CREATION DE L'ONTOLOGIE AVEC PROTEGE OWL	78
5.2.1 <i>Création de classes et hiérarchie de classes</i>	<i>78</i>
5.2.2 <i>Création de propriétés</i>	<i>80</i>
<i>Création d'attributs.....</i>	<i>81</i>
<i>Création d'instance.....</i>	<i>82</i>
<i>Les règles SWRL</i>	<i>83</i>
<i>Exécution de requêtes SQWRL</i>	<i>85</i>
CONCLUSION GENERALE ET PERSPECTIVES	87

Liste des Figures

Figure1.1. XML, RDF et OWL : les 3 couches du WS [1]...	7
Figure1.2. les couche du web sémantique [1]	8
Figure1.3 La grammaire des expressions conceptuelles selon AL [4].....	12
Figure1.4.les sous langages d'OWL [8].	16
Figure1. 5. Complexité des dialectes OWL et langages connexes [9].....	18
Figure1. 6. Condition en RIF et la correspondance en règle RIF Horn [10].....	18
Figure 1.7 : construction d'une ontologie opérationnelle	25
Figure1. 8. Architecture du système d'inférence [18].....	26
Figure 1.9. Réduction des problèmes d'inférence d'une TBox à des problèmes de subsomption.....	27
Figure 1.10. Réduction des problèmes d'inférence d'une TBox à des problèmes de satisfiabilité.....	27
Figure 1.11. Augmentation du niveau d'expressivité avec SWRL [34]	30
Figure 2.12. Architecture du système de raisonnement.....	35
Figure2.13.Fonctionnement du système de raisonnement.....	36
Figure2.14.Cycle de vie d'une ontologie [13].....	37
Figure 2.15 : Un Flux De Travail Du HDO	40
Figure2.16.Description de concepts et rôles	42
Figure 2.17. Définition d'un rôle.....	42
Figure2.18. Définition des axiomes... ..	43
Figure 2.19. conception d'une ontologie dans le domaine hospitalier	55
Figure2.20.Démarche d'une consultation médicale	47
Figure2. 21. Parcours du patient Consultant en première ligne.....	67
Figure2. 22 .Parcours du patient consultant au urgences dans les établissements de santé COVID-19... ..	68
Figure 3.23. Interface de PROTEGE OWL	76
Figure3.24. L'éditeur de règles SWRL... ..	77
Figure3.25. Création de classes.....	78
Figure 3.26.Structures des classes principaux et hiérarchie des classes	79
Figure 3.27. Création de propriétés	79
Figures3.28.Ontographe des propriétés.....	81

Figure 3.29. Créations d'attributs.....	82
Figures3.20. création des instances	83
Figures 3.31. Les règles SWRL.....	84
Figures 3.32. Editeur des Règles SWRL.....	84
Figure 3.33. Les règles SWRL.....	85
Figures 3.34. Résultat d'exécution d'une règle SQWRL.....	86

Liste des Tables

Tableau 1.1. Syntaxe et sémantique des différents constructeurs.....	13
Tableau2.2 Syntaxe du langage SHIQ	41
Tableau2.3. Glossaire de termes... ..	50
Tableau2.4 Table dictionnaire des concepts... ..	56
Tableau2.5 Table des relations binaires	59
Tableau2.6 Table des attributs	60
Tableau 3.7 Glossaire des termes (COVID-19).....	69
Tableau 3.8 Dictionnaire des concepts	70
Tableau 3.9 relations binaires (COVID-19)....	71
Tableau 3.10 Les règle SWRL.....	73

Introduction générale

Le monde entier fait face à une crise sanitaire sans précédent due à la pandémie de Covid-19. La progression rapide des cas depuis février 2020 constitue un défi majeur pour le système de santé mondiale. Dans ce contexte, l'hôpital devient saturé de malades, avec une situation critique et très compliquée à gérer. Les services hospitaliers sont perturbés ; se traduisant par, le manque de lits de réanimation, le manque du personnel médical, le manque des moyens de tests (scanner, PCR,..) et aussi des médicaments nécessaires comme l'oxygène.

Le système d'information joue un rôle important pour la gestion de la crise, le processus de partage d'information en tout moment et en temps réel pour éliminer ses perturbations des services hospitaliers, devient primordial.

Selon l'Organisation Mondiale de la Santé (OMS) : « *On entend par qualité des soins la mesure dans laquelle les services de santé pour les individus et les populations augmentent la probabilité d'obtenir les résultats sanitaires escomptés et sont conformes aux connaissances professionnelles à bases factuelles dont on dispose* » [1].

L'objectif de cette étude, est d'améliorer le Système d'Information Hospitalier (SIH) dans le cas des crises par un processus de partage des connaissances entre les services hospitaliers, utilisant la technologie de web sémantique. Le Web Sémantique a pour but d'ajouter des informations caractéristiques aux informations existantes pour aider à l'automatisation des services.

Dans le cadre de la pandémie, nous avons proposé une ontologie à base des services hospitaliers appelés HDO (Hospital Departments Ontology), nous avons défini les principaux concepts des services et leurs relations, une conception des règles SWRL, ainsi que l'interrogation de l'ontologie utilisant les requêtes SQWRL. L'objectif est de faciliter le partage des connaissances et d'informations entre les services hospitaliers, et faciliter la réaffectation du personnel et des équipements entre les services. Par exemple, les services de médecine interne prennent en charge la grande majorité des patients hospitalisés nécessitant des interactions critiques avec les services des urgences, Cette gestion requiert d'importants efforts d'organisation et de

coordination avec les autres acteurs des services, afin d'assurer l'hospitalisation de l'afflux de patients atteints de COVID-19 tout en garantissant la prise en charge équitable des patients atteints d'autres pathologies.

Après cette introduction générale de notre travail, le reste de ce mémoire est structuré comme suit :

Chapitre 1: présente la notion de web sémantique, principes et composantes principales, la notion de l'ontologie, les différentes classifications, méthodologies et outils nécessaires pour son développement.

Chapitre 2: Dans ce chapitre nous présentons notre conception de l'ontologie HDO, les concepts principaux, les relations entre concepts et leurs attributs.

Chapitre 3: nous appliquons notre ontologie proposée dans un service Covid-19 pour la réaffectation des ressources, décrivons une implémentation de l'ontologie présentée dans le chapitre précédent. La création de l'ontologie est faite en utilisant l'éditeur Protégé, la connaissance de domaine est modélisée par les règles SWRL et l'interrogation de notre ontologie par les requêtes SWRL. Enfin, nous terminons par une conclusion générale.



Chapitre 01 :

**Technologie de web sémantique à
base des ontologies**

Introduction

Le Web sémantique (plus techniquement appelé « le Web de données ») permet aux machines de comprendre la sémantique, la signification de l'information sur le Web. Il étend le réseau des hyperliens entre des pages Web classiques par un réseau de lien entre données structurées permettant ainsi aux agents automatisés d'accéder plus intelligemment aux différentes sources de données contenues sur le Web et, de cette manière, d'effectuer des tâches (recherche, apprentissage, etc.) plus précises pour les utilisateurs. Le terme a été inventé par Tim Berners-Lee, co-inventeur du Web et directeur du W3C, qui supervise l'élaboration des propositions de standards du Web sémantique.

La plupart du temps, lorsque l'on prononce le terme de Web sémantique, on parle des différentes technologies qui se cachent derrière. Parmi les plus connues, on peut citer RDF (Resource Description Framework) qui correspond à un modèle d'information, et les formats d'échanges de données en RDF pour communiquer entre différentes applications.

Dans le domaine du Web sémantique, la sémantique des données est décrite par des ontologies avec des langages prévus pour fournir une description formelle de concepts, termes ou relations d'un domaine quelconque.

Le web sémantique

L'expression de Web Sémantique, énoncée par [1] fait d'abord référence à la vision du Web du futur comme un vaste espace d'échange de ressources supportant la collaboration entre humains et machines en vue d'exploiter plus efficacement de grands volumes d'informations et de services variés disponibles sur le Web.

L'objectif du Web sémantique est de rendre explicite le contenu sémantique des ressources dans le Web (documents, pages web, services, etc.). Les machines et les agents logiciels pourraient -comprendre les contenus décrits dans les ressources et faciliter les tâches de traitement des informations de façon plus automatique et plus efficace. Avec cette idée le moteur de recherche sémantique peut savoir et trouver exactement les documents contenant l'information grâce aux sémantiques attachées dans ces documents

Architecture et langages du web sémantique :

La vision courante du Web sémantique proposée par [1] peut être représentée dans une architecture en plusieurs couches différentes (**Figure 2**), dont les trois premières sont considérées comme couche de base (**Figure 1**).

Chaque niveau repose sur les résultats définis au niveau inférieur, c'est-à-dire que chaque niveau est progressivement plus spécialisé et plus complexe que le niveau précédent.

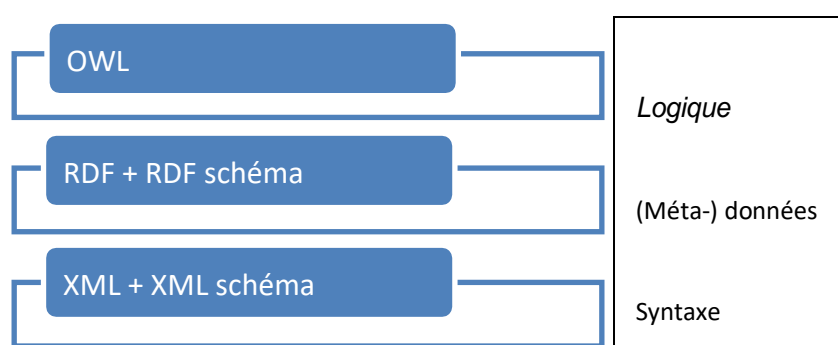


Figure1.1. XML, RDF et OWL : les 3 couches du WS [1]

Les couches inférieures sont stabilisées à l'heure actuelle. Par contre, la faisabilité des couches supérieures ne semble toujours pas claire.

Le Web Sémantique recommande un ensemble d'outils syntaxiques spécifiques aux traitements des ressources numériques. Il repose sur une architecture pyramidale composée de différents langages de représentation des connaissances contenues dans le Web. Cette architecture propose une combinaison de principes, eux-mêmes convertis en langages informatiques. Chaque langage de couche supérieure vient spécifier les couches de niveau inférieur. En effet, chacun de ces langages de représentation vient compléter les informations du langage précédent, ajoutant au fur et à mesure divers formalismes pour le traitement des données. Ces couches sémantiques de complexités croissantes viennent ainsi graduellement spécifier les informations inhérentes aux ressources par rapport à un domaine et à un usage donné.

Grâce aux efforts établis pour réaliser la vision du Web sémantique et au grand consensus dans la communauté d'inclure les règles avec l'ontologie dans l'architecture du Web sémantique, un

diagramme de couche plus raffiné a résultat [1], ce diagramme est montré dans la partie (b) de la Figure 2.

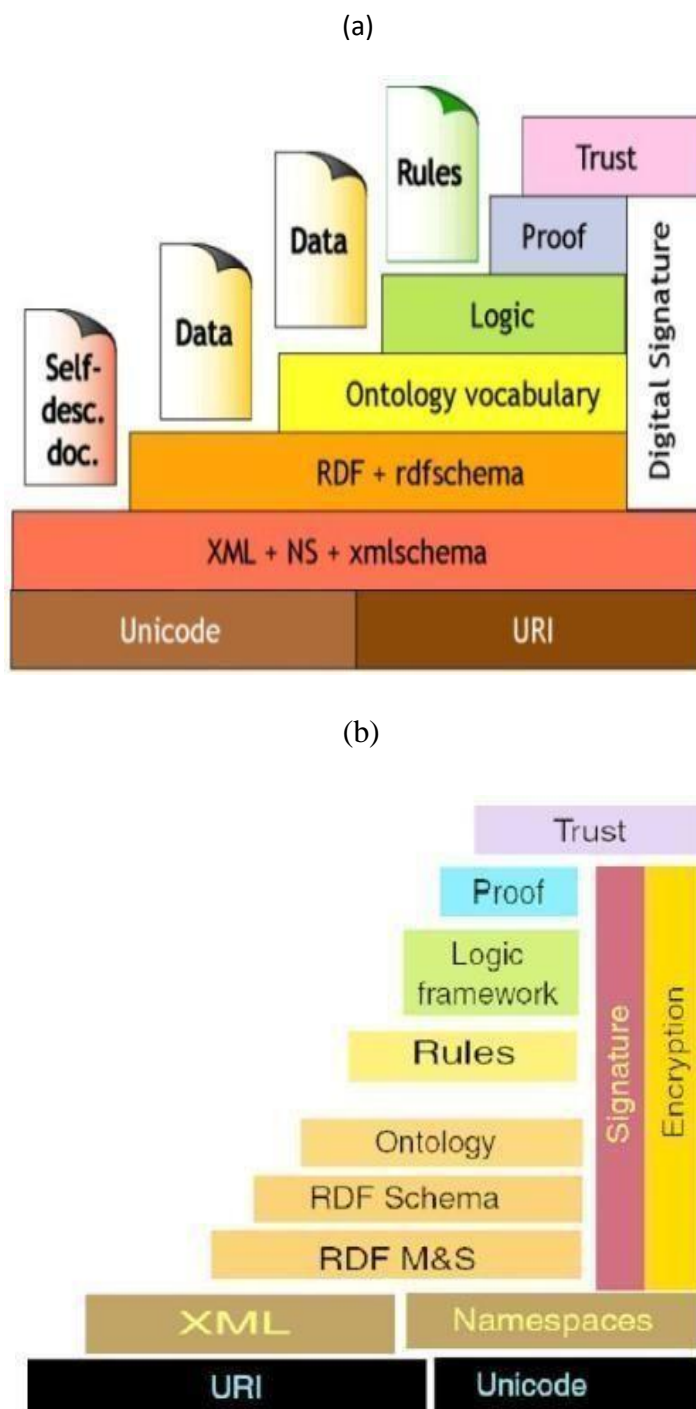


Figure1.2. les couches du web sémantique [1]

La couche qui a actuellement atteint une maturité suffisante est la couche d'ontologie sous la forme du langage OWL. La prochaine étape dans le développement du web sémantique est la réalisation des couches de règles, logique et preuve qui seront développées en dessus de la couche d'ontologie. Pour mieux comprendre l'utilité de la couche de règles qui présente un formalisme fondé sur la programmation logique dans l'infrastructure du web sémantique et plus particulièrement dans les applications de recherches d'informations, nous exposons en détail les différentes couches de cette architecture.

La couche URI et UNICODE

Cette Architecture repose sur URI¹, (Uniform Resource identifier) qui permet d'attribuer un identifiant unique à une ressource assurant ainsi la localisation des différentes ressources sur la Toile. Ce protocole d'identification des ressources est commun au Web hypertexte, c'est la couche la plus stable de l'architecture.

La couche XML

XML² est la syntaxe sur laquelle repose l'ensemble de l'architecture ascendante du Web Sémantique. Ce langage va permettre d'apposer aux documents des métadonnées via des balises sémantiques (également appelées marqueurs ou annotations sémantiques). Ces métadonnées décrivent le contenu des documents, les rendant ainsi interprétables par les systèmes informatiques. Les balises XML vont permettre de construire un index, une base de données formulée à partir des concepts contenus dans les ressources. En ce sens, le balisage Sémantique est comparable à l'indexation documentaire, comme elle, la qualification des ressources demande un important effort d'analyse et de catégorisation des informations pertinentes.

¹Uniform Resource Identifier

²eXtensibleMarkupLanguage

La couche RDF

Les couches RDF³ et RDF Schéma⁴ sont considérées comme les premières fondations de l'interopérabilité sémantique. Elles permettent de décrire les taxonomies des concepts et des propriétés (avec leurs signatures). RDF fournit un moyen d'insérer de la sémantique dans un document, l'information est conservée principalement sous forme de déclarations RDF. Le schéma RDFS⁵ décrit les hiérarchies des concepts et des relations entre les concepts, les propriétés et les restrictions domaine/co-domaine pour les propriétés. Les trois langages XML, RDF et RDFS seront présentés plus en détail dans la partie suivante.

La couche ontologie

La couche « Ontology vocabulary », dont le langage le plus usité à l'heure actuelle est OWL . Le langage OWL consiste à conceptualiser de manière formelle un monde de connaissance de sorte à pouvoir générer des règles d'inférences exploitables par les machines. OWL étend les capacités de représentation de RDF au moins tout autant que sa complexité.

Couche règles

Les **règles** sont un élément clé de la vision du Web sémantique, la couche Règles offre la possibilité et les moyens de l'intégration, de la dérivation, et de la transformation de données provenant de sources multiples, etc.

La couche Logique

La couche logique se trouve au-dessus de la couche Ontologie. Certains considèrent ces deux couches comme étant au même niveau, comme des ontologies basées sur la logique et permettant des axiomes logiques. Elle s'appuie sur des règles d'inférences qui permettent d'inférer de nouvelle. Une large variété de logiques a été conçue jusqu'à présent. Étant le formalisme le mieux apprécié dans la représentation de la connaissance, la logique de description est celle qui est généralement la plus adoptée pour la représentation des règles d'inférences.

Le développement des **LD** a été fortement influencé par les travaux sur la logique des prédicats, les

schémas (frames) introduits par [2] et les réseaux sémantiques développés par [3].

Logiques de description

Logiques de description (LD), une famille de langages de représentation de connaissances utilisés pour la représentation et le raisonnement sur des connaissances d'un domaine d'application d'une manière structurée et formelle.

La modélisation des connaissances d'un domaine avec les LD se réalise en deux niveaux : le niveau terminologique ou TBox, et le niveau factuel ou ABox. Alors une base de connaissances K est une paire (T, A) où T est un ensemble d'axiomes —terminologiques (TBox), A est un ensemble d'axiomes —assertionnels (ABox).

Les entités de base qui sont définies et manipulées dans une logique de description sont les **concepts**, les **rôles** et les **individus**. Un concept permet de représenter un ensemble d'individus, tandis qu'un rôle représente une relation binaire entre individus. Un concept correspond à une entité générique d'un domaine d'application et un individu à une entité particulière, instance d'un concept.

Donc une ontologie est définie par l'expression suivante : $O = \{C, R, H, rel, A\}$ où

C : concept, R : relation binaire, H : hiérarchie, rel : la relation de subsomption, A : axiomes introduisant les concepts et relations.

Il existe plusieurs langages de description de concepts et de rôles. Nous décrivons la logique de description minimale **AL** (*Attributive Language*) introduite par [04] et qui représente la base des logiques de description.

La figure ci-dessous (3) illustre les constructeurs offerts par **AL** pour l'édification de concepts composés.

$C, D \longrightarrow$	A	(concept atomique)
	\top	(le concept universel)
	\perp	(le concept le plus spécifique)
	$\neg A$	(la négation atomique)
	$C \sqcap D$	(l'intersection)
	$\exists R.T$	(quantification existentielle limitée)
	$\forall R.C$	(quantification universelle complète)

Figure 3 : La grammaire des expressions conceptuelles selon AL [4]

- Le concept TOP (\top) dénote le concept le plus général. L'extension de (\top) inclut tous les individus possibles.
- Le concept BOTTOM (\perp) le concept le plus spécifique. L'extension de (\perp) est l'ensemble vide.
- Le constructeur $\neg A$ est utilisé pour évoquer la négation d'un concept atomique, c'est à-dire les individus qui n'appartiennent pas au concept atomique A .
- Le constructeur $C \sqcap D$ permet de faire la conjonction de deux concepts composés, ce qui représente l'ensemble des individus qui sont à la fois membres du concept C et du concept D .
- Le quantificateur existentiel non typé $\exists R.T$ désigne l'ensemble des individus, membres du domaine d'un rôle R dans T .
- Le quantificateur universel $\forall R.C$ désigne l'ensemble des individus du domaine d'un rôle R qui sont en relation, par le biais de R , avec un individu du concept C .

Le langage $\mathcal{AL} = \{\top, \neg, \perp, A, C \sqcap D, \forall r, C, r\}$ peut être enrichi de plusieurs constructeurs qui permettant de former toute une famille de logiques de description.

Le tableau⁶ ci-dessous (**Tableau 1**) illustre la syntaxe et sémantique des différents constructeurs

⁶http://fr.wikipedia.org/wiki/Logique_de_description

Tableau 1.1. Syntaxe et sémantique des différents constructeurs

Lettre	Constructeur	Syntaxe	Sémantique
$\mathcal{A}\mathcal{C}$	nom de concept	C	$C^{\mathcal{I}}$
$\mathcal{A}\mathcal{C}$	top	\top	$\Delta^{\mathcal{I}}$
\mathcal{C}	négation de concepts non nécessairement primitifs	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$\mathcal{A}\mathcal{C}$	conjonction	$C_1 \sqcap C_2$	$C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$
\mathcal{U}	disjonction	$C_1 \sqcup C_2$	$C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}}$
$\mathcal{A}\mathcal{C}$	quantificateur universel	$\forall R.C$	$\{d_1 \in \Delta^{\mathcal{I}} \mid \forall d_2 \in \Delta^{\mathcal{I}}.(R^{\mathcal{I}}(d_1, d_2) \rightarrow d_2 \in C^{\mathcal{I}})\}$
\mathcal{E}	quantificateur existentiel typé	$\exists R.C$	$\{d_1 \in \Delta^{\mathcal{I}} \mid \exists d_2 \in \Delta^{\mathcal{I}}.(R^{\mathcal{I}}(d_1, d_2) \wedge d_2 \in C^{\mathcal{I}})\}$
\mathcal{N}	restriction de nombre	$(\geq n R)$ $(\leq n R)$	$\{d_1 \in \Delta^{\mathcal{I}} \mid \{d_2 \mid R^{\mathcal{I}}(d_1, d_2)\} \geq n\}$ $\{d_1 \in \Delta^{\mathcal{I}} \mid \{d_2 \mid R^{\mathcal{I}}(d_1, d_2)\} \leq n\}$
\mathcal{Q}	restriction de nombre qualifiée	$(\geq n R.C)$ $(\leq n R.C)$	$\{d_1 \in \Delta^{\mathcal{I}} \mid \{d_2 \mid R^{\mathcal{I}}(d_1, d_2), d_2 \in C^{\mathcal{I}}\} \geq n\}$ $\{d_1 \in \Delta^{\mathcal{I}} \mid \{d_2 \mid R^{\mathcal{I}}(d_1, d_2), d_2 \in C^{\mathcal{I}}\} \leq n\}$
\mathcal{O}	un-de	$\{a_1, \dots, a_n\}$	$\{d \in \Delta^{\mathcal{I}} \mid d = a_i^{\mathcal{I}} \text{ pour un } a_i\}$
\mathcal{B}	role filler	$\exists R.\{a\}$	$\{d \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(d, a^{\mathcal{I}})\}$
$\mathcal{A}\mathcal{C}$	nom de rôle	R	$R^{\mathcal{I}}$
\mathcal{R}	conjonction de rôles	$R_1 \sqcap R_2$	$R_1^{\mathcal{I}} \cap R_2^{\mathcal{I}}$
\mathcal{I}	rôles inverses	R^{-1}	$\{(d_1, d_2) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(d_2, d_1)\}$
\mathcal{H}	hiérarchie des rôles	$R_1 \sqsubseteq R_2$	$R_1^{\mathcal{I}} \subseteq R_2^{\mathcal{I}}$
\mathcal{R}^+	transitivité des rôles	R^+	Plus petite relation transitive contenant $R^{\mathcal{I}}$

L'interprétation selon l'hypothèse du monde ouvert

Dans les LD, l'interprétation des connaissances respecte l'hypothèse du monde ouvert. Contrairement à la sémantique des bases de données traditionnelles, l'absence d'information représente l'ignorance plutôt qu'une information négative. Pour enrichir la comparaison avec les bases de données, répondre à une requête pour un système bâti sur les LD nécessite d'effectuer un raisonnement logique souvent plus complexe qu'une simple recherche pour vérifier la présence d'une information, car le système doit souvent considérer plusieurs interprétations possibles, une conséquence de l'hypothèse du monde ouvert [5].

Couche preuve et confiance

Au sommet de la pyramide, les couches « preuve » et « confiance » qui bien que séparées dans les strates de l'architecture du Web Sémantique sont souvent associées. Ces couches permettent l'authentification et la validation des ressources du Web. Le Web étant un espace virtuel où chacun peut déverser des ressources de tout ordre, il est nécessaire de pouvoir contrôler la validité de leur contenu, notamment à l'aide de la signature électronique. L'utilité de ces couches peut paraître quelque peu superflue, cependant la démocratisation des principes du Web Sémantique

peut conduire des sites peu scrupuleux à biaiser les métadonnées. Il suffit pour cela d'ajouter des métadonnées décrivant des informations populaires chez les utilisateurs afin de s'assurer une meilleure visibilité sur la Toile. C'est d'ailleurs une pratique courante. A travers ces dernières couches le Web Sémantique fait en quelque sorte signer un contrat de bonne conduite aux diffuseurs des ressources numériques.

Les langages du web sémantique

Le Web sémantique propose des langages spécialement conçus pour les données : RDF (Resource Description Framework), OWL (Ontology Web Language), et XML (eXtensibleMarkupLanguage). HTML décrit les documents et les liens entre eux.

XML (eXtensibleMarkupLanguage)

Le **XML**, désigne un langage informatique (ou métalangage pour être plus précis) utilisé, entre autres, dans la conception des sites Web et pour faciliter les échanges d'informations sur Internet. Ce langage de description a pour mission de formaliser des données textuelles. Il s'agit, en quelque sorte, d'une version améliorée du langage HTML avec la création illimitée de nouvelles balises.

XML est un langage à balisage extensible et il est considéré comme une Spécification pour les documents lisible par les machines. Il est utilisé pour encoder les langages du Web sémantique. Le balisage signifie que certaines suites des caractères du document peuvent contenir de l'information indiquant le rôle du contenu du document. XML se sert de balises (tags, par exemple <nom> ou <adresse>) pour décrire un classement des données du document et sa structure logique. Cependant, le caractère extensible indique la différence importante avec d'autres langages précédents qui est aussi la caractéristique essentielle du XML.

RDF

RDF (Resource description Framework [6] est une infrastructure permettant d'encoder, échanger, et réutiliser des métadonnées sur le web.

RDF nous permet de faire des déclarations sur les ressources. Le format de ces déclarations est simple. Une déclaration a toujours la structure suivante :

< sujet > < prédicat > < objet >

Une déclaration RDF exprime une relation entre deux ressources. Le sujet et l'objet représentent les deux ressources en relation ; le prédicat représente la nature de leur relation. La relation est formulée de manière directionnelle (depuis le sujet vers l'objet) et est appelée dans RDF une propriété. Puisque les déclarations RDF consistent en trois éléments, elles sont appelées triplets.

Un triplet RDF est similaire à la déclaration < sujet — prédicat — objet >, où :

- a) **Ressource** (*Sujet*) : Cela peut être n'importe quel objet référencé par une URI, qu'il concerne le web (Page HTML, document PDF, fichier multimédia...), ou non (Personne, Région, Etc.).
- b) **Propriété** (*prédicat*) : Critère, caractéristique, attribut ou relation qui peut décrire la ressource (titre, couleur, taille, auteur, etc.). Une propriété ses valeurs permises et ses relations avec les autres propriétés.
- c) **Valeur** (*objet*) : C'est la valeur qui sera affectée à la propriété de la ressource. Cette affectation peut être soumise à certaines restrictions.

RDFS (Resource Description Framework Schéma) –méta-modèle

Le langage RDFS a été développé en se basant sur RDF [6]. Le modèle des données RDF ne précise que le mode de description des données mais ne fournit pas la déclaration des propriétés spécifiques au domaine ni la manière de définir ces propriétés avec d'autres ressources. RDFS a pour but d'étendre RDF en décrivant plus précisément les ressources utilisées pour étiqueter les graphes. Pour cela, il fournit un mécanisme permettant de spécifier les classes dont les instances seront des ressources, comme les propriétés.

RDFS s'écrit toujours à l'aide de triplets RDF en utilisant deux propriétés fondamentales **subClassOf** et **type** pour représenter respectivement les relations de subsomption entre classes et les relations d'instanciation entre instances et classes.

Langage OWL

OWL offre aux machines de plus grandes capacités d'interprétation du contenu Web que celles permises par XML, RDF et RDF schéma (RDFS), grâce à un vocabulaire supplémentaire et une sémantique formelle. Inspiré des logiques de descriptions (et successeur de DAML+OIL), OWL fournit un grand nombre de constructeurs permettant d'exprimer de façon très fine les classes de manière plus complexe correspondant aux connecteurs de la logique de description équivalente (intersection, union, restrictions diverses, etc.), les propriétés des classes définies (telles que la disjonction), la cardinalité (par exemple "exactement un"), plus des types des propriétés (propriétés d'objet ou d'annotation...), des caractéristiques des propriétés (par exemple la symétrie, la transitivité), et des classes énumérées.

Le langage OWL se compose de trois sous-langages (**Figure 4**) offrant une expressivité croissante [8]: OWL-Lite, OWL-DL et OWL-Full.

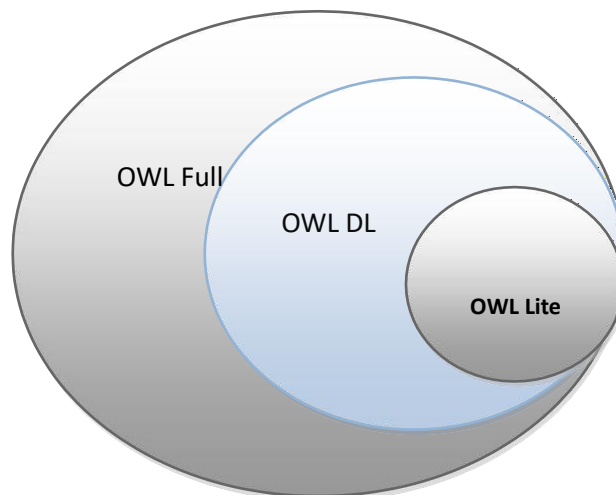


Figure1.4. les sous langages d'OWL [8]

a) Le langage de base **OWL-lite** pour la description des hiérarchies de classification et des contraintes simples. Ce sous-langage ne contient qu'un sous-ensemble réduit des constructeurs disponibles. Il a la complexité formelle la plus basse et l'expressivité minimale dans la famille OWL. Il est suffisant pour représenter des thésaurus et d'autres taxonomies ou des hiérarchies de classification avec des contraintes simples.

- Il convient aux utilisateurs qui ont principalement besoin d'une hiérarchie de classification et de contraintes simples.
 - Il reprend tous les constructeurs de RDF (c'est-à-dire fournit des mécanismes permettant de définir un individu comme instance d'une classe, et de mettre des individus en relation),
 - Il utilise les mots-clés de RDFS (RDFS:subclassof, RDFS:property, RDFS:subpropertyof, RDFS:range, RDFS:domain), avec la même sémantique,
 - Il supporte les contraintes de cardinalité, mais ne permet d'utiliser que les valeurs 0 ou 1.
- b) **OWL-DL** pour une description d'une expressivité maximale tout en prenant en compte la logique de description afin d'offrir des propriétés de calcul nécessaires aux systèmes de raisonnement.
- Il convient aux utilisateurs qui veulent le maximum d'expressivité,
 - Il reprend tous les constructeurs d'OWL lite,
 - Il permet tout entier positif dans les contraintes de cardinalité,
 - Il tire son nom de sa correspondance avec les logiques de descriptions.
 - La grande complexité de ce langage semble rendre nécessaire une approche heuristique.

OWL-FULL pour la meilleure expressivité dans la description des ontologies, mais sans offrir aucune garantie de calcul.

- Il reprend tous les constructeurs d'OWL DL,
- Il reprend tout tous les constructeurs de RDF schema,
- Il permet d'utiliser une classe en position d'individu dans les constructeurs.

OWL 2 est devenu une recommandation du W3C le 27 octobre 2009. OWL2 est compatible avec OWL1, et peut être utilisé avec des informations écrites en RDF. Les ontologies OWL 2 sont principalement échangées en documents RDF.

On trouve aussi une diversité de dialectes du langage OWL, la figure ci-dessous (**Figure 5**) représente un diagramme de complexité des différents dialectes d'OWL, et les langages connexes vis-à-vis leur disposition envers les LD et LP based-sémantiques.

Les langages qui ne sont pas encore standardisés sont représentés avec des cadres en pointillés, Tandis que le rectangle du langage OWLIM en rouge est jugé comme le plus complexe des dialectes d'OWL.

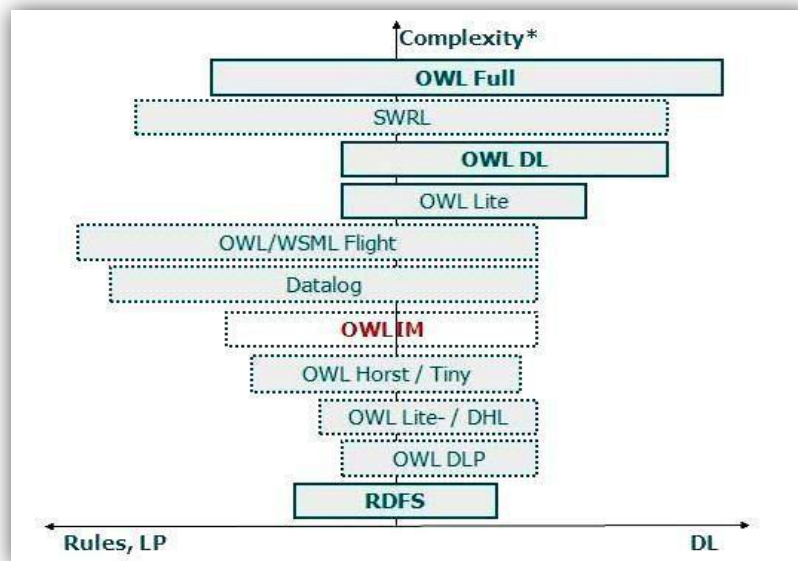


Figure1. 5. Complexité des dialectes OWL et langages connexes [9]

RIF (Rule Interchange Format)

RIF [10] est un formalisme proposé par le W3C permettant de fournir l'interopérabilité entre les langages de règles en général et ceux utilisés en particulier pour le Web. Le noyau de ce langage, RIF Core, correspond à la logique de Horn. Il offre certaines extensions inspirées des langages à objets et de frames, ainsi que les URIs, les types de données de XML Schema...etc. La partie principale de RIF Core est le langage de condition (Condition Language). Ce langage définit la syntaxe et la sémantique des règles de RIF ainsi que la syntaxe pour des requêtes. D'autre part, la spécification de RIF repose sur certains types de règles : les règles de production, la programmation logique, les règles basées sur la logique du premier ordre, les règles réactives ou les règles normatives... Nous montrons ci-dessous (Figure 6) une condition représentée en RIF et la correspondance en règle RIF Horn.

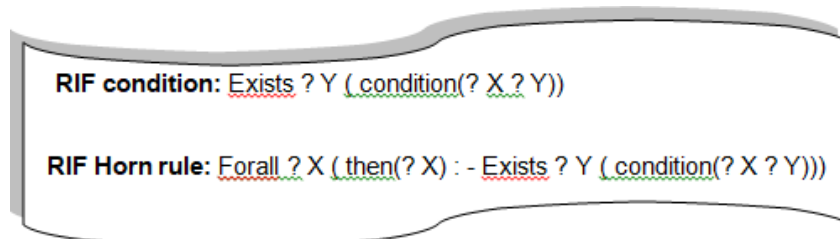


Figure1. 6. Condition en RIF et la correspondance en règle RIF Horn [10]

Ingénierie ontologique

Le besoin de représenter les connaissances dans les systèmes informatiques a conduit certains chercheurs à la naissance des ontologies.

Le terme *ontologie* est initialement emprunté de la philosophie « Etude de l'être en tant qu'être » signifiant —explication systématique de l'existence. Une ontologie est similaire à un dictionnaire ou un glossaire mais avec une structure détaillée et grande qui permet aux machines de traiter son contenu.

Une des définitions les plus célèbres et la plus utilisée est celle de [11] : « *Une ontologie est une spécification explicite d'une conceptualisation* ».

L'ontologie est une *spécification* parce qu'elle représente la conceptualisation dans une forme concrète. Elle est *explicite* parce que tous les concepts et les contraintes utilisés sont explicitement définis. Une *conceptualisation* est une abstraction du monde que nous souhaitons représenter dans un certain but. La conceptualisation est le résultat d'une analyse ontologique du domaine étudié.

Une ontologie exprime la conceptualisation explicitement dans un langage formel.

Une définition explicite et formelle permet aux agents de raisonner et d'inférer de nouvelles connaissances.

N. GUARINO affine la définition de **T. GRUBER** en considérant les ontologies comme des spécifications partielles et formelles d'une conceptualisation [12].

Les ontologies sont formelles car exprimées sous forme logique, et partielles car une conceptualisation ne peut pas toujours être entièrement formalisée dans un cadre logique du fait d'ambiguïtés ou du fait qu'aucune représentation de leur sémantique n'existe dans le langage de représentation d'ontologies choisi [13].

Constituants d'une ontologie

Avec l'aide d'une interprétation en termes d'un modèle de l'ontologie, les connaissances du domaine sont représentées par une sémantique formelle. D'après [14], et [15], il est clair d'une ontologie est constituée de concepts et de relations entre ceux-ci. S'il existe quelques différences concernant la structure, les composants d'une ontologie sont les mêmes. Les

connaissances du domaine traduites par une ontologie sont véhiculées normalement par Les concepts, relations, axiomes, et instances :

Concept

Un concept peut représenter un objet, une notion, une idée. Un concept peut être divisé en trois parties : un terme (ou plusieurs), une notion et un ensemble d'objets. Le **terme** est un élément lexical qui permet d'exprimer le concept en langue naturelle, il peut admettre des synonymes. La **notion** également appelée *intension* du concept, contient la sémantique du concept, exprimée en termes de propriétés et attributs, et de contraintes. L'ensemble **d'objets** appelé *extension* du concept, regroupe les objets manipulés à travers le concept ; ces objets sont appelés instances du concept.

De fait, les concepts manipulés dans un domaine de connaissance sont organisés au sein d'un réseau de concepts liés par des propriétés conceptuelles. Les propriétés portant sur des concepts sont :

L'abstraction

Un concept est abstrait si toute instance de ce concept est aussi instance d'un de ses Concepts fils. Par exemple, dans une hiérarchie comportant les concepts *homme* et *femme*, fils du Concept *humain*, le concept *humain* est abstrait.

La subsomption

Un concept C1 subsume un concept C2 si toute propriété sémantique de C1 est aussi une propriété sémantique de C2, c'est-à-dire si C1 est plus spécifique que C2. L'extension d'un concept subsumé est forcément plus réduite que celle du concept qui le subsume. Son intension est par contre plus riche. La subsomption sert à la hiérarchisation de l'ensemble des concepts de l'ontologie. Par exemple, *homme* subsume *humain*.

L'équivalence: deux concepts sont équivalents s'ils ont la même extension. Par exemple, *étoile du matin* et *étoile du soir*.

La disjonction : on parle aussi d'incompatibilité) : deux concepts sont disjoints si leurs extensions sont disjointes. Par exemple, *homme* et *femme*. Si tous les concepts fils d'un concept abstrait sont disjoints deux à deux, ils forment une **partition** du concept abstrait.

Relation

Les relations représentant des types d'interactions entre les concepts. Une relation permet de lier des instances de concepts, ou des concepts génériques. Elles sont caractérisées par un terme (voir plusieurs) et une signature qui précise le nombre d'instances de concepts que la relation lie, leurs types et l'ordre des concepts, c'est-à-dire la façon dont la relation doit être lue.

Tout comme les concepts, les relations peuvent être spécifiées par des propriétés. Les propriétés intrinsèques d'une relation sont :

Les propriétés algébriques: symétrie, réflexivité, transitivité, antisymétrie, anti réflexivité.

La cardinalité: il s'agit du nombre possible de relations de ce type pouvant exister entre les mêmes concepts (ou instances de concept). Les relations portant une Cardinalité représentent souvent des attributs. Par exemple, une *pièce* a au moins une *porte*, un *humain* a entre zéro et deux *jambes*.

Les propriétés liant deux relations sont principalement:

L'incompatibilité: Deux relations sont incompatibles si elles ne peuvent lier les mêmes instances de concepts. Par exemple, les relations *fils-de* et *père-de* sont incompatibles

L'inverse : Deux relations binaires sont inverses l'une de l'autre si, quand l'une lie deux instances i_1 et i_2 , l'autre lie i_2 et i_1 . Par exemple, les relations *fils-de* et *père-de* sont inverses l'une de l'autre.

L'exclusivité: deux relations sont exclusives si, quand l'une lie des instances de concepts, l'autre ne lient pas ces instances, et vice-versa. L'exclusivité entraîne L'incompatibilité. Par exemple, l'appartenance et la non appartenance sont exclusifs.

Axiome

Les axiomes constituent des assertions (ou des prédicats) acceptées comme vraies qui s'appliquent sur les classes ou les instances des classes de l'ontologie et qui permettent de restreindre les interprétations possibles d'une ontologie et/ou de déduire de nouveaux faits à partir des faits connus.

Instance

Instances (ou objets ou individus) représentent des éléments spécifiques d'un concept (ou d'une classe).

Les instances peuvent être associées à un identifiant unique qui Utilisation des ontologies.

Utilisation des ontologies :

Les ontologies sont utilisées dans plusieurs domaines, les plus répandus sont :

La communication

Une ontologie facilite la communication en fournissant une spécification explicite d'un domaine qui représente un modèle normatif. De plus, les ontologies permettent d'assurer la consistante et d'enlever l'ambiguïté dans les descriptions des connaissances concernant un domaine spécifique.

Enfin, les ontologies peuvent intégrer différentes perspectives des utilisateurs. Quand les utilisateurs (qui ont différentes perspectives d'un domaine) partagent une ontologie, ils ont une perspective standard.

L'interopérabilité entre les systèmes

Les ontologies permettent de faciliter l'interopérabilité en intégrant les connaissances concernant différents domaines dont l'objectif est de décrire un domaine unifié ou accomplir une tâche commune. Elles permettent aussi d'intégrer les différents vocabulaires concernant certains domaines. Pour ce faire, les ontologies de ces domaines doivent être intégrées par les méthodes d'intégration d'ontologies afin de partager un même vocabulaire

- a) **Ingénierie des systèmes** : Le développement des systèmes basé sur les ontologies a donné un profit à l'ingénierie de systèmes qui peut être résumé commesuit
- b) **Réutilisabilité** : l'ontologie encode les informations relatives à un domaine (y compris les composants logiciels) de sorte que le partage et la réutilisation sontpossibles.
- c) **Acquisition des connaissances** : l'ontologie guide l'acquisition desconnaissances.
- d) **Sûreté** : l'ontologie rend possible l'automatisation du processus de

vérification de consistance.

- e) **Spécification** : l'ontologie aide le processus d'identification des besoins et la définition des spécifications des systèmes [16].

Les typologies d'ontologies

Pour la typologie des ontologies, on peut distinguer différents niveaux d'ontologies selon le but pour lequel elles sont conçues. La classification de [17] repose sur deux critères : le sujet et la structure d'une conceptualisation. Concernant le sujet de la conceptualisation, les auteurs distinguent :

- **Les ontologies de domaine** : les plus connues, elles expriment des conceptualisations spécifiques à un domaine, elles sont réutilisables pour des applications sur ce domaine.
- **Les ontologies d'application** : elles contiennent des connaissances du domaine nécessaires à une application donnée ; elles sont spécifiques et non réutilisables.
- **Les ontologies génériques** : appelées aussi ontologies de haut niveau, elles expriment des conceptualisations très générales tels que le temps, l'espace, l'état, le processus, les composants, elles sont valables dans différents domaines; les concepts figurant dans une ontologie du domaine sont subsumés par les concepts d'une ontologie générique, la frontière entre les deux étant floue.
- **Les ontologies de représentation ou méta-ontologies** : indiquent des formalismes de représentation de la connaissance ; les ontologies génériques ou du domaine peuvent être écrites en utilisant des primitives d'une telle ontologie. D'autre part, les ontologies sont aussi classées en plusieurs niveaux d'expression [17] selon leur utilisation. Quatre catégories principales qui vont des ontologies hautement informelles jusqu'aux ontologies rigoureusement formelles:
 - **Très informelle** : l'ontologie est exprimée en langage naturel.

- **Semi-informelle** : l'ontologie est exprimé sous une forme restreinte et structurée de langage naturel pour augmenter la clarté et pour réduire l'ambiguïté.
- **Semi-formelle** : l'ontologie est exprimée en langage formel.
- **Rigoureuse formelle** : l'ontologie est défini avec une sémantique formelle, permettant le théorème et la preuve.

Construction d'ontologie

La construction d'une ontologie intervient donc après qu'un travail de conceptualisation ait été mené à bien. Ce travail consiste à identifier, au sein d'un corpus, les connaissances spécifiques au domaine de connaissance à représenter, et conventionnellement reconnues comme Relevant de ce domaine. « *A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose* » [11]

Le processus général de construction d'ontologies opérationnelles (**Figure 7**) peut donc être découpé en 3 phases [13] :

- a) **La conceptualisation** : identification des connaissances contenues dans un corpus représentatif du domaine. Ce travail doit être mené par un expert du domaine, assisté par un ingénieur de la connaissance.
- b) **L'ontologisation** : formalisation, autant que possible, du modèle conceptuel obtenu à l'étape précédente. Ce travail doit être mené par l'ingénieur de la connaissance, assisté de l'expert du domaine.

Afin de respecter les objectifs généraux des ontologies, T. GRUBER propose 5 critères permettant de guider le processus d'ontologisation [11] :

- La clarté et l'objectivité des définitions, qui doivent être indépendantes de tout choix d'implémentation.
- La cohérence (consistance logique) des axiomes
- L'extensibilité d'une ontologie, c'est-à-dire la possibilité de l'étendre sans modification.

- La minimalité des postulats d'encodage, ce qui assure une bonne portabilité.
- La minimalité du vocabulaire, c'est-à-dire l'expressivité maximum de chaque terme.

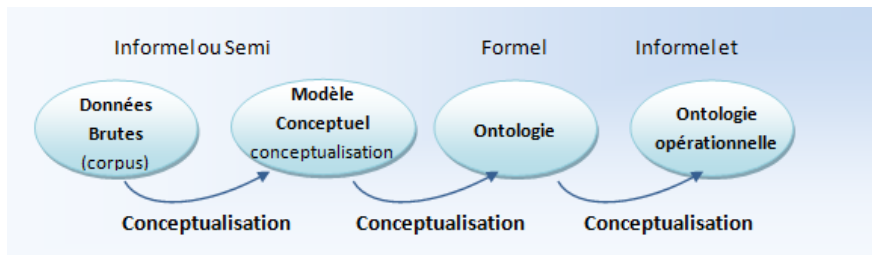


Figure 1.7 : construction d'une ontologie opérationnelle

L'opérationnalisation :

Transcription de l'ontologie dans un langage formel et opérationnel de représentation de connaissances, pour permettre à une machine, via cette ontologie, de manipuler des connaissances du domaine. Ce travail doit être mené par l'ingénieur de la connaissance.

Les ontologies sont considérées comme des spécifications partielles et formelles d'une conceptualisation [13]. Elles sont formelles car exprimées sous un formalisme doté d'une sémantique formelle, et partielle car 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.

Raisonnement sémantique sur une ontologie OWL

La représentation des connaissances par les ontologies peut s'accompagner des mécanismes de raisonnement. L'**inférence** concerne la manipulation des connaissances déjà acquise pour produire de nouvelles connaissances. C'est un terme général dont les mots raisonnement, déduction, induction sont des cas spéciaux.

Il utilise des mécanismes d'inférence permettant la résolution des problèmes pour lesquels il n'existe pas de procédures explicites dans le programme.

1.11.1 Inférence dans les logiques de description

La plupart des raisonneurs sont conçus pour raisonner sur les logiques de descriptions, mais acceptent en entrée des fichiers OWL. L'inférence s'effectue au niveau terminologique ou assertionnel. **La Figure 8** présente l'architecture proposée pour l'inférence basée sur la logique de description.

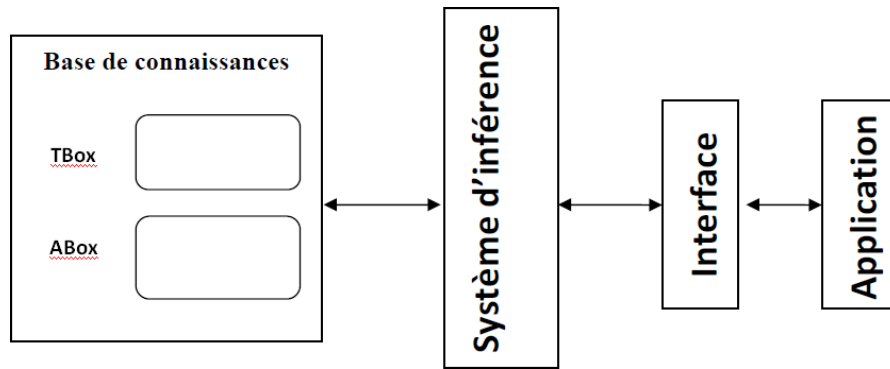


Figure1. 8. Architecture du système d'inférence [18].

Dans les LDs, l'inférence s'effectue au niveau terminologique ou factuel. Les inférences de bases sont :

La satisfiabilité : Un concept peut-il avoir des instances ? La satisfiabilité va permettre d'évaluer la cohérence d'une construction ontologique.

La subsumption : Un concept en subsume-t-il un autre ? La subsumption va permettre de valider une classification. Le mécanisme de classification va permettre de replacer un concept ou une propriété dans les hiérarchies de concepts et de propriétés, aidant en cela à la construction et à la maintenance évolutive des ontologies.

La classification : Où placer un concept dans la hiérarchie?

Au niveau terminologique

Quatre principaux problèmes d'inférence se présentent au niveau terminologique [19] :

La satisfiabilité : Un concept C d'une terminologie T peut être satisfait si et seulement s'il existe un modèle M de T tel que $C^M = \emptyset$.

La subsumption : Un concept C est subsumé par un concept D pour une terminologie si et seulement si $C^M \subseteq D^M$ pour tout modèle M de T .

L'équivalence : Un concept C est équivalent à un concept D pour une terminologie et seulement si $C^M \equiv D^M$ pour chaque modèle M de T .

La disjonction : Des concepts C et D sont disjoints pour une terminologie T si et seulement si $C^M \cap D^M = \emptyset$; pour chaque modèle M de T .

Les moteurs d'inférence actuels tirent généralement profit du fait que les quatre types de problèmes d'inférence peuvent être réduits à des problèmes de subsumption ou à de satisfiabilité. Il a

été prouvé par [20] que les moteurs d'inférence nécessitent souvent qu'un seul algorithme pour raisonner au niveau terminologique. D'ailleurs, les deux grandes classes d'algorithmes de raisonnement pour les logiques de description correspondent aux façons de réduire respectivement des problèmes d'inférence à des problèmes de subsumption (**Figure 9**) et de satisfiabilité (**Figure 10**) [19].

C est insatisfiable	\iff	C est subsumé par \perp
C et D sont équivalents	\iff	C est subsumé par D , et D par C
C et D sont disjoints	\iff	$C \sqcap D$ est subsumé par \perp

Figure 1.9. Réduction des problèmes d'inférence d'une TBox à des problèmes de subsumption

C est subsumé par D	\iff	$C \sqcap \neg D$ est insatisfiable
C et D sont équivalents	\iff	$C \sqcap \neg D$ et $\neg C \sqcap D$ sont insatisfiables
C et D sont disjoints	\iff	$C \sqcap D$ est insatisfiable

Figure 1.10. Réduction des problèmes d'inférence d'une TBox à des problèmes de satisfiabilité.

Au niveau factuel (ABox)

Le niveau factuel comprend quatre principaux problèmes d'inférence [19]

Cohérence : Une ABox A est cohérente par rapport à une TBox T si et seulement s'il existe un modèle M de A et T .

Vérification d'instance : ce problème consiste à vérifier par inférence qu'une assertion $S(a)$ est vraie pour tout modèle d'une ABox A et d'une TBox T .

Vérification de rôle : ce problème consiste à vérifier par inférence si une relation $R(a, b)$ est vraie pour tout modèle d'une ABox A et d'une TBox T .

Problème de récupération : ce problème consiste à déduire pour une ABox A et un concept C d'une terminologie T tous les individus d'un modèle M de T .

Outils pour le web sémantique :

Editeurs d'ontologie :

Il existe un grand nombre d'outils de construction d'ontologies, dont nous présentons seulement: Protégé [20] :

Protégé :

Protégé-OWL est un éditeur qui permet de construire une ontologie pour un domaine donné, de définir des formulaires d'entrée de données, et d'acquérir des données à l'aide de ces formulaires sous forme d'instances de cette ontologie. Protégé est également une librairie java qui peut être étendue pour créer de véritables applications à bases de connaissances en utilisant un moteur d'inférence pour raisonner et déduire de nouveaux faits par application de règles d'inférence aux instances de l'ontologie et à l'ontologie elle-même (méta-raisonnement). Des « plugin » pour les langages RDF, Daml+Oil et OWL ont été développés pour protégé, aussi pour éditer les règles SWRL [21]. Ces « plugin » permettent d'utiliser protégé comme éditeur d'ontologies pour ces différents langages, de créer des instances et les sauver dans les formats respectifs.

Il est également possible de raisonner sur les ontologies en utilisant un moteur d'inférence général tel que Jess [22], ou des outils d'inférence spécifiques au web sémantique basés sur des logiques de description tels que Racer [23]. Ces deux outils peuvent être facilement intégrés à protégé.

Par ailleurs, une des particularités de protégé-OWL est qu'il fournit aux

Utilisateurs la possibilité de formuler des requêtes de manière très simple, pour l'interrogation de l'ontologie.

Intégration d'un langage de règle avec un langage d'ontologie :

L'intégration d'un langage de règle avec un langage d'ontologie exige la définition d'un Nouveau langage, sa syntaxe et sémantique et le développement des algorithmes de raisonnement pour le nouveau langage [32]. Les propositions existantes pour l'intégration des langages de règles avec des langages d'ontologie doivent être classifiées par le degré d'intégration [33].

Approche homogène :

Dans cette approche, les règles et les ontologies sont incorporées dans le même langage logique L (par exemple le langage SWRL ou DLP) sans faire une distinction à priori entre les prédicats de règles et les prédicats d'ontologie. Les ontologies et les règles sont traitées d'une manière homogène, comme un nouveau langage logique. L'idée générale est que les règles peuvent utiliser des prédicats unaires et binaires de l'ontologie c'est-à-dire les classes et les propriétés ainsi, un nouveau raisonneur est nécessaire, capable de gérer le nouveau langage homogène qui émerge. Nous allons présenter dans la suite le langage SWRL qui est un exemple de cette approche.

Le résultat d'une intégration homogène est un langage, tel que SWRL. Un système de raisonnement sur une ontologie OWL enrichie par des règles SWRL peut lui-même être homogène.

Langage SWRL :

SWRL [29] est un langage de règles pour le web sémantique combinant le langage OWL-DL et le langage RuleML (RuleMarkupLanguage (Unary/BinaryDatalog)). SWRL est un langage qui enrichit la sémantique d'une ontologie définie en OWL. SWRL permet contrairement à OWL de manipuler des instances par des variables (?x, ?y, ?z).

SWRL ne permet pas de créer des concepts ni des relations. Il permet simplement d'ajouter des relations suivant les valeurs des variables et la satisfaction de la règle.

OWL permet de spécifier des propriétés algébriques pour les relations. Mais il lui manque surtout des possibilités pour encoder des connaissances plus générales, relatives en particulier à la composition des relations. Ainsi, SWRL constitue un pas vers le rapprochement entre les langages OWL et les règles logiques. SWRL intègre une syntaxe abstraite de haut niveau pour les règles de Horn dans le langage OWL.

OWL et SWRL sont les langages fondamentaux du Web sémantique. OWL est un langage d'ontologie développé pour construire des ontologies fournissant un haut niveau d'expressivité pour le contenu du Web. SWRL est basée sur OWL, il garde la puissance d'OWL DL mais au prix de la décidabilité. La figure 1.11 montre que l'ajout des règles à l'ontologie fournit une couche supplémentaire d'expressivité. SWRL permet aux utilisateurs d'écrire des règles pour raisonner sur les individus et pour en déduire de nouvelles connaissances sur ces individus.

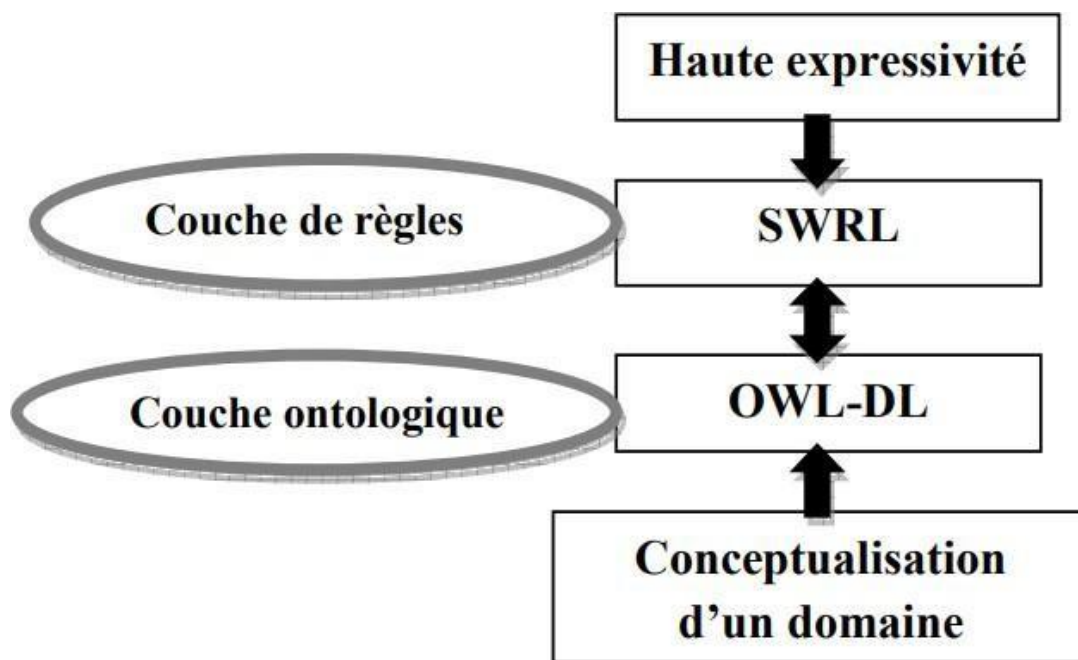


Figure 1.11. Augmentation du niveau d'expressivité avec SWRL [34]

La syntaxe

La syntaxe SWRL définit une règle qui est une relation d'implication entre un antécédent (corps) et un conséquent (tête). Si les conditions spécifiées dans l'antécédent sont vérifiées c'est-à-dire si l'on a pu démontrer la partie corps, alors les conditions spécifiées dans le conséquent le sont aussi (antécédent \Rightarrow conséquent).

L'antécédent et le conséquent d'une règle sont des conjonctions d'atomes. Les atomes peuvent avoir les formes:


- $C(x)$ où C est une description OWL, x est soit une variable, un individu OWL ou encore des valeurs de données (data values) de OWL.
- $P(x, y)$ où P est une propriété OWL (Object_property ou data_type_property), x est soit une variable ou des individus OWL.

Edition des règles SWRL

Une des étapes clés de l'interopérabilité de règles sur le Web est le langage SWRL, qui vise à être le langage de règles standard du web sémantique. Un des avantages de Protégé est de pouvoir attacher un grand nombre de plugins venant notamment enrichir la représentation de l'ontologie et permettant sa validation. L'éditeur de règles SWRL [35] opère dans Protégé OWL. Il fournit une interface très interactive pour l'édition des règles prenant en charge l'ensemble des fonctionnalités du langage SWRL.

14. Conclusion

Nous avons dressé au cours de ce chapitre un rapide aperçu sur les notions d'ontologie et le web sémantique. Nous nous sommes attachés à définir les ontologies en présentant leur composant, leur utilisation, leur construction, ainsi que le mode de raisonnement sur ces ontologies, et les outils disponible pour édition et l'inférence.



***Chapitre 2 : Conception
d'une Ontologie à base des
Services Hospitaliers
(HDO)***

Introduction :

La recherche d'informations sur le Web concerne de multiples utilisateurs et notamment du monde médical.

Les besoins majeurs attendus en médecine du Web Sémantique sont de pouvoir trouver facilement des informations médicales sur le Web, partager ces informations grâce au Web et ainsi pouvoir les exploiter pour l'aide à la décision.

Ce chapitre présente notre contribution au problème posé par ce mémoire, à savoir la conception et le raisonnement sur une ontologie hospitalière. Pour la modélisation de cette ontologie, nous avons choisi des langages très répandues : OWL-DL étendu par des règles dans le langage SWRL.

L'architecture du système de raisonnement sémantique dans les ontologies que nous avons adopté est donnée dans la figure 12.

Cette architecture repose sur le principe d'avoir deux modules d'inférences, s'exécutant en parallèle, en échangeant leurs conclusions : un classifieur pour raisonner sur l'ontologie OWL, et un moteur de règles, qui admet en entrée les règles SWRL.

Nous devons construire l'ontologie hospitalière. Une ontologie dans le domaine médical, spécialement dans l'hospitalisation pour structurer les connaissances, les organiser, et plus précisément raisonner dessus. Pour y arriver, nous devons d'abord définir un processus de développement d'ontologie de domaine définie par le langage OWL. Pour ce faire, nous nous sommes basés sur le travail de [24].

Ensuite nous donnerons la Représentation sémantique des règles avec le langage SWRL qui va augmenter l'expressivité de notre ontologie, et qui permet d'inférer de nouvelles connaissances.

Objectif

L'objectif principal de notre travail est de concevoir une ontologie OWL, et

Ensuite l'enrichir de règles de production afin de pouvoir faire des raisonnements et inférences et déduire de nouvelles informations.

Pour accomplir cet objectif, il faut mettre en œuvre les sous objectif suivants :

- Description de l'architecture du système de raisonnement et d'inférence.
- Conception de l'ontologie **HDO (Hospital Départements Ontology)** , et décrire le domaine par le biais des concepts et relations.
- Rédiger les règles SWRL, que vont pouvoir ajouter de nouvelles assertions à l'ontologie.

Une fois, notre ontologie conceptualisée, elle sera aussitôt éditée par protégé et ensuite vérifiée par le système de raisonnement

Architecture générale du système :

Le raisonnement sur une ontologie OWL enrichie par les règles SWRL nécessite un raisonneur qui prend en charge à la fois le langage OWL et les règles SWRL. Il existe quelques raisonneurs, qui acceptent en entrée les règles SWRL, mais le raisonnement ne sera pas complet.

Architecture du système

L'architecture de ce système de raisonnement est illustrée par la figure 12.

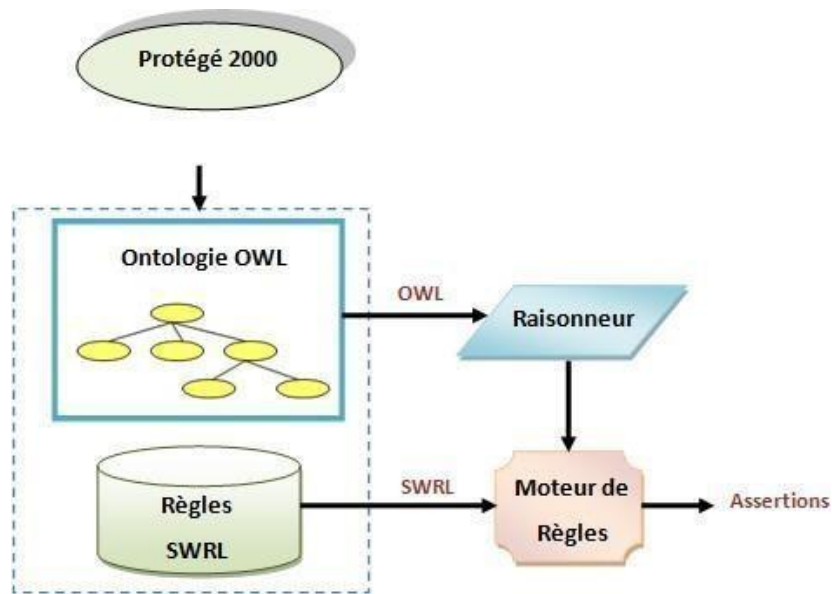


Figure 2.12. Architecture du système de raisonnement

L'architecture du système proposé est constituée des composants suivants:

(1):L'ontologie : elle est constituée de trois parties:

- Base terminologique : qui fournit les définitions logiques des concepts (classes), des relations (propriétés), et des axiomes (les conditions nécessaires etsuffisantes)
- Base assertionelle : qui contient les individus (instances de classes) et les instances des relations qui lesrelient.
- Base de règles : qui fournit des règles étendant l'ontologie pour augmenter son expressivité.

(2) : Leraisonneur : Tel que Racer ou pellet pour effectuer des raisonnements sur la section OWL del'ontologie.

(3) : Lemoteur de règles : Tel que Jess pour raisonner sur le fragment de règlemodélisé avecSWRL.

Fonctionnement du système :

La figure 12 ci-dessous explique le fonctionnement du système de raisonnement :

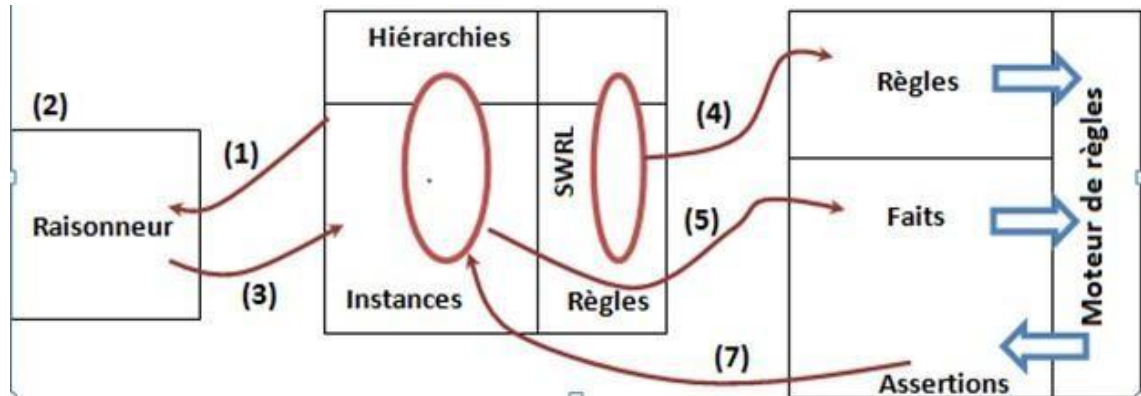


Figure2.13.Fonctionnement du système de raisonnement

Le raisonnement sur une ontologie hybride s'effectue en suivant les étapes suivantes :

- (1) : Chargement de l'ontologie OWL au niveau du raisonneur(pellet)
- (2) : Exécution du raisonneur. Les inférences de base sont établies, et des résultats sont générés.
- (3) : Mise à jour de la base assertionnelle de l'ontologie par les résultats obtenus dans (2).
- (4) : Chargement des règles SWRL au niveau du moteur de règles (Jess). Ces règles se placent dans la base de règles du moteur de règles.
- (5) : Chargement des fragments d'OWL de l'ontologie, et se placent dans la base de faits du moteur d'inférence.
- (6) : Exécution du moteur de règles, et génération de nouveaux faits
- (7) : Mise à jour de l'ontologie par les nouvelles assertions obtenues dans(6).

Conception d'une ontologie dans le domaine médical

Cycle de vie d'une ontologie

Les ontologies étant destinées à être utilisées comme des composants logiciels dans des systèmes répondant à des objectifs opérationnels différents, leur développement doit s'appuyer sur les mêmes principes que ceux appliqués en génie logiciel. Différentes méthodes de construction ont été proposées (METHONTOLOGY, TOVE, ...), nous allons en parler en détail ci-dessous.

Un cycle de vie inspiré du génie logiciel est présenté dans [13]. Il comprend une étape initiale d'évaluation des besoins, une étape de construction, une étape de diffusion, et une étape d'utilisation. Après chaque utilisation significative, l'ontologie et les besoins sont réévalués et l'ontologie peut être étendue et, si nécessaire, en partie reconstruite. Le cycle de vie d'une ontologie est schématisé par la figure 14 suivante :

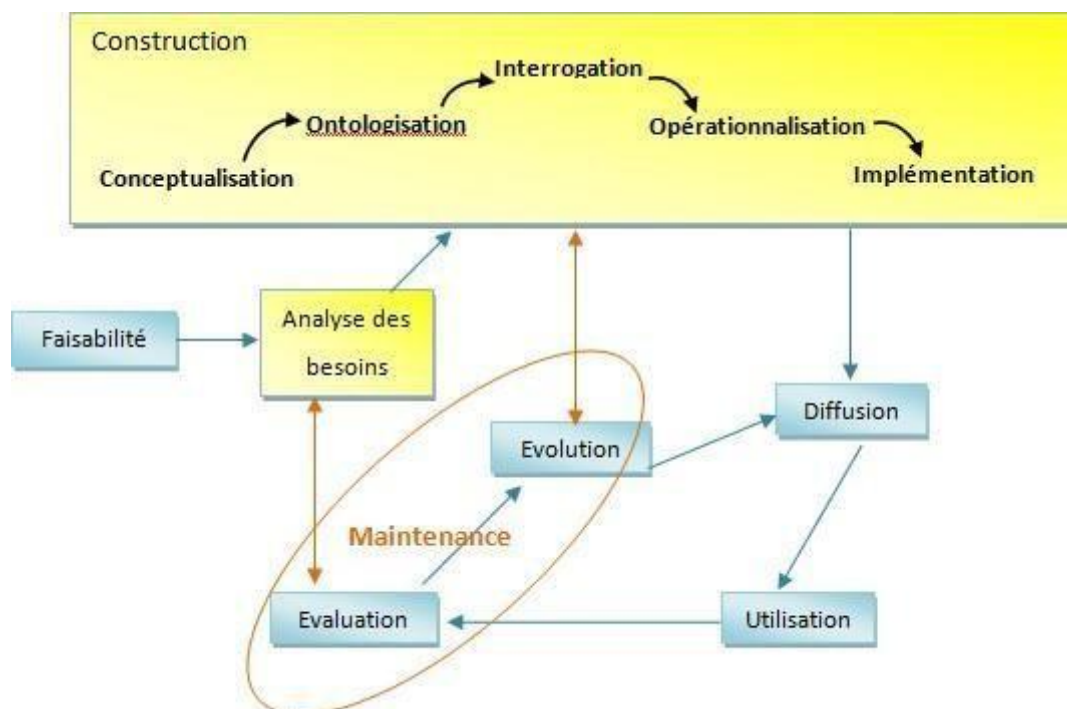


Figure2.14.Cycle de vie d'une ontologie [13]

Méthodologies de construction d'ontologies

Une méthodologie étant considérée comme un ensemble de principes de construction systématiquement reliés, appliqués avec succès par un auteur dans la construction d'ontologies. Les méthodologies peuvent porter sur l'ensemble du processus et guider l'otologiste dans toutes les étapes de la construction. A l'heure actuelle, il existe plus de trente méthodologies de conception d'ontologies. Nous citons seulement quelques unes :

- METHONTOLOGY [25] : Élaborée en 1998 par A. GOMEZ-PEREZ, elle couvre tout le cycle de vie d'une Ontologie.
- M. USCHOLD et M. KING [26] ont également proposé une méthodologie générale, inspirée de leur expérience de construction d'ontologies dans le domaine de la gestion des entreprises.
- La méthodologie présentée par M. GRUNINGER et M.S. FOX [27] est elle aussi issue d'une expérience de construction d'ontologie sur ce domaine.
- B. BIÉBOW propose à la fois une méthode de construction de ressources terminologiques et ontologiques à partir de textes et un outil, TERMINAE, implémentant cette méthode.
- La méthodologie ONTOSPEC de G.KASSEL constitue une aide à la structuration des hiérarchies de concepts et de relations durant la phase d'ontologisation.
- C'est également le cas des principes différentiels énoncés par B. BACHIMONT [28] et des critères de classification des propriétés, proposés par N. GUARINO et C. WELTY.

Vers un processus de construction d'une ontologie OWL

Nous avons proposé un processus de construction d'une ontologie d'application partant de connaissances brutes et arrivant à une ontologie d'application opérationnelle représentée par le langage OWL. Ce processus est composé de cinq étapes [24] :

- Spécification des besoins.
- Conceptualisation.
- Formalisation.
- Implémentation.
- Test & évolution de l'ontologie.

Spécification

Cette phase est inspirée du processus proposé de [24] qui consiste à établir un document formel de spécification des besoins représenté dans le langage RDF. Ce dernier permet de décrire l'ontologie à construire à travers les cinq aspects (i.e. propriétés) suivants :

1. **Le domaine de connaissance** : déterminer aussi précisément que possible le domaine que va couvrir l'ontologie.
2. **L'objectif** : le but de l'ontologie à créer pour le domaine considéré.
3. **Les utilisateurs** : identifier au maximum les futurs utilisateurs de l'ontologie à créer.
4. **Les sources d'informations** : déterminer les sources d'informations d'où les connaissances seront obtenues, par exemple, les experts du domaine, les documents techniques, etc., sont des sources de connaissance.
5. **La portée de l'ontologie** : déterminer à priori la liste des termes (les plus importants) pour le domaine à représenter.

Processus

Notre processus comprend quatre composants clés (voir la figure 15) :

- (1) La base de connaissances : C'est l'élément le plus important du processus, la connaissance du domaine (Domaine Hospitalier) et une base des règles SWRL.
- (2) Le système de partage d'information.
- (3) Le moteur d'inférence : c'est le module de raisonnement du système, par exemple, le moteur Raisonner Hermit1.4.3.456 dans cette ontologie, un moteur de raisonnement qui peut vérifier la cohérence de l'ontologie développée pour éliminer les erreurs, est également un élément important.
- (4) L'interface de requête SQWRL.

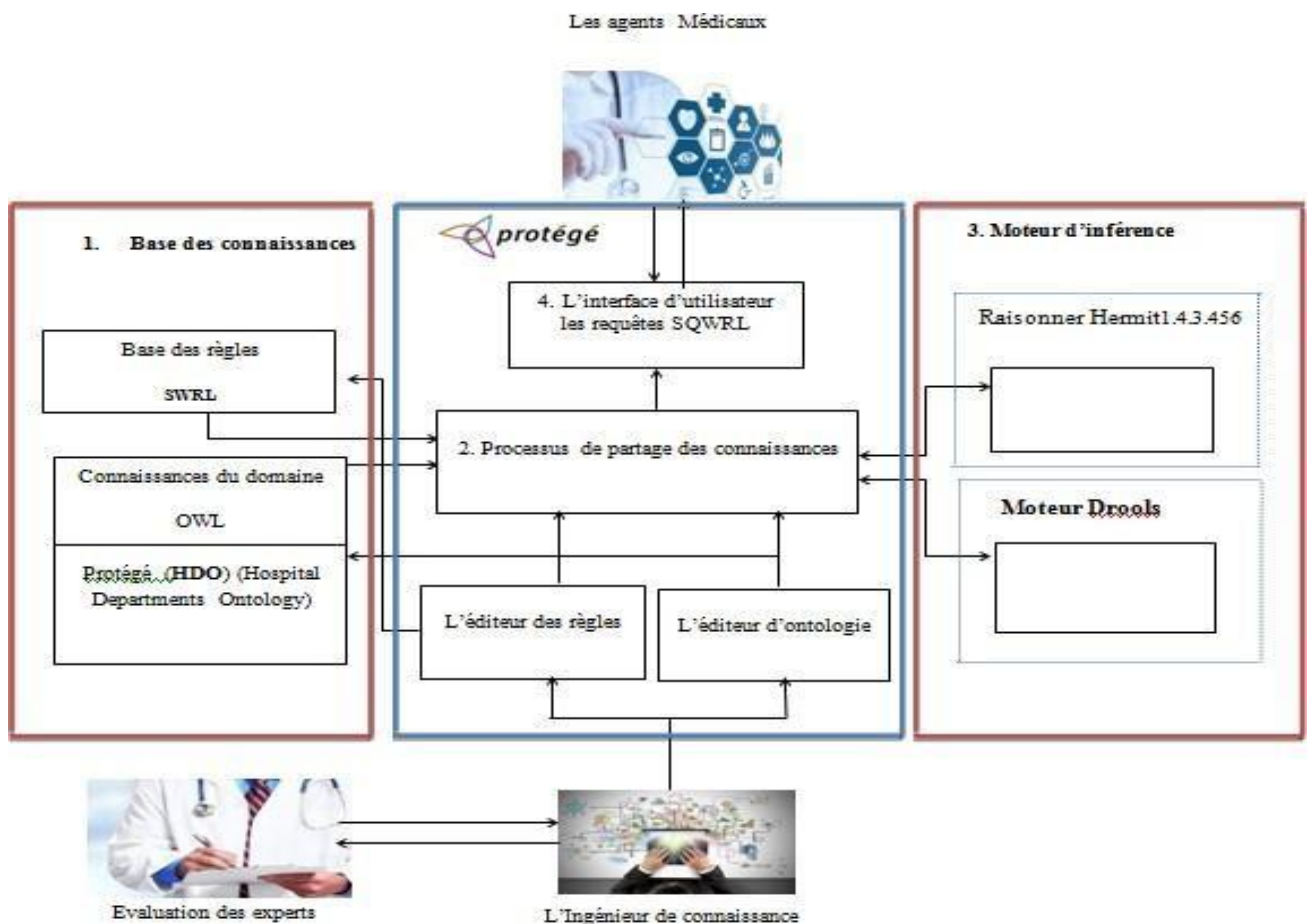


Figure 2.15 : Un Flux De Travail Du HDO

Conceptualisation:

C'est l'étape la plus importante dans le processus de construction de l'ontologie. Elle est inspirée de la méthodologie METHONTOLOGY qui consiste à identifier et à structurer, à partir des sources d'informations, les connaissances du domaine. Elles permettent d'aboutir à un ensemble de représentations intermédiaires semi-formelles indépendamment des langages de formalisations à utiliser pour représenter l'ontologie. A la fin de cette phase, nous obtenons une ontologie conceptuelle. Pour cela, on distingue les principales tâches suivantes :

- Construction du glossaire de termes.
- Construction du diagramme de relations binaires et de classification des concepts
- Construction du dictionnaire de concepts(DC).
- Décrire les relations dans une table de relations binaires.
- Spécifier des contraintes sur les attributs dans une table d'attributs.
- Spécifier des axiomes sur les concepts dans une table d'axiomes logiques.
- Décrire les instances des concepts dans une table d'instances.

La description de chaque étape est définie au fur et à mesure de la construction de l'ontologie HDO .

Formalisation

Cette phase consiste à formaliser l'ontologie conceptuelle obtenue dans la phase précédente afin de faciliter sa représentation ultérieure dans un langage complètement formel et opérationnel. Notre choix est porté sur le formalisme de représentation de la logique de description en s'appuyant sur sa syntaxe de type SHIQ qui présente une logique de description très expressive et qui offre un certain nombre de constructeurs pour décrire les concepts.

Tableau2.2 Syntaxe du langage SHIQ

<i>Syntaxe</i>	<i>Interprétation</i>
T	Le concept le plus général (TOP)
\perp	Le concept le plus spécifique (Bottom)
C	Le nom d'un concept
R	Le nom d'un rôle (une relation binaire ou bien un attribut)

A	Le nom d'un individu
$C1 \cap C2$	Conjonction de concepts pour définir de nouveaux concepts
$C1 \cup C2$	Disjonction de concepts pour définir de nouveaux concepts
$\neg C$	Utilisé pour définir le complément d'un concept
$\forall R.C$	Définit le co-domaine du rôle R
$\exists R.C$	Il y a au moins un objet relié par le rôle R au concept C
$(\geq nR.C)$ $(\leq nR.C)$	Cardinalité Minimum/Maximum (n est un nombre entier non négatif).
R^-	Le rôle inverse

La logique de description est constituée de deux parties : une partie terminologique (TBox) permettant de décrire les concepts et les rôles et d'une partie assertionnelle (ABox) décrivant les instances.

2.8.1 Partie terminologique TBox

```

<Concept> → <concept-primitif> | <T> | <⊥> |
           <Concept> ∩ <Concept> | <Concept> ∪ <Concept> |
           ¬<Concept> | ∀<role>. <Concept> | ∃<role>. <Concept> |
           ≥n<role>. <Concept> | ≤n<role>. <concept> |
<role> → < rôle-primitif> <role> → <role> ∪ <role> | <role> ∩ <role>

```

Figure 2.16. Description de concepts et rôles

$R : (C1, C2)$ où R est un nom de rôle, C1 est un concept source et C2 est un concept cible **$R^- := R1$**
où R1 est le rôle inverse pour R

Figure 2.17. Définition d'un rôle

Ce qui montre dans la grammaire précédente que les concepts et les rôles sont exprimés de manière déclarative. Cette dernière comprend des définitions de concepts et de rôles ainsi que des inclusions de concepts. Une définition de rôle est peut être définie aussi en fonction des concepts qui l'utilisent. Nous pouvons utiliser l'une des formes suivantes (Figure 17):

2.8.2 Partie assertion elle ABox

Elle définit l'ensemble des axiomes utilisés pour décrire des situations concrètes. Pour chaque individu et pour chaque relation qui relie deux individus, nous décrivons les deux formes suivantes (Figure 18) :

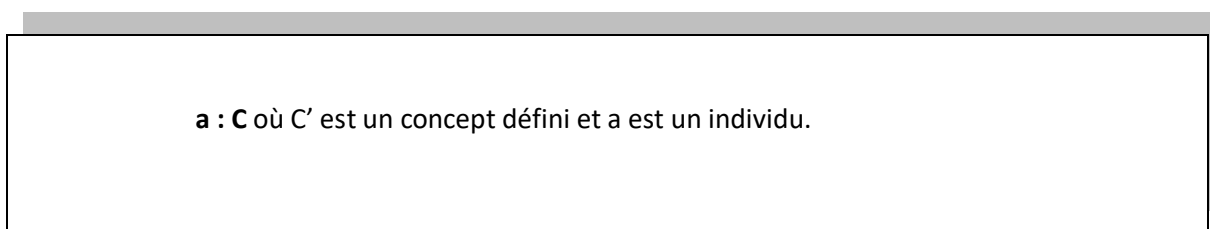


Figure2.18. Définition des axiomes

Implémentation

L'ontologie que nous avons obtenue dans la phase précédente est appelée une ontologie formelle. Le but de cette étape sera donc de coder l'ontologie formelle en OWL qui dispose de fonctionnalités sémantiques plus riches que ses prédécesseurs comme RDFS et DAML+OIL. A la fin de cette phase, nous aurons une ontologie opérationnelle.

Afin de faciliter le processus de codification, nous utilisons PROTEGE-OWL disposant d'une interface modulaire, permettant l'édition, la visualisation, le contrôle (vérification des contraintes) d'ontologies, issu du modèle des frames et contient des classes (concepts), des slots (propriétés) et des facettes (valeurs des propriétés et contraintes), ainsi que des instances des classes et des propriétés.

Tests et évolution

Cette étape consiste à exploiter les services d'inférence fournis par la logique de description afin de supporter le processus de construction et d'améliorer la qualité de l'ontologie. Pour ce faire, nous proposons l'utilisation de l'outil RACER, un système de la logique de descriptions. Ce dernier, permet de lire un document au format OWL (ontologie OWL) et de le

Représenter sous forme d'une base de connaissances LD et de fournir des services d'inférence pour les niveaux TBox et ABox. Cette étape sert aussi à suivre l'évolution de l'ontologie, c'est-à-dire les nouveaux concepts à ajouter dans la partie terminologique (TBox) de l'ontologie. Une classification a lieu chaque fois qu'une définition de concept est nouvellement créée. Le mécanisme de raisonnement de base des logiques de description est la classification de concepts. Elle est réalisée par un algorithme de classification, appelé « le classifieur ». Le classifieur utilise la description d'un nouveau concept pour le placer à l'endroit correspondant dans la hiérarchie. Afin de trouver la place appropriée au nouveau concept, l'algorithme de classification détermine les relations de subsumption entre ce concept et les autres; Ces relations peuvent être spécifiées directement, trouvées par transitivité ou calculées à partir de la sémantique des conditions des rôles. La recherche de la place correcte pour le nouveau concept comporte trois étapes :

- (1) : La recherche des subsumant les plus spécifiques SPS (concepts qui subsument le concept à classer et dont les fils ne le subsument pas)
- (2) : La recherche des subsumés les plus généraux SPG (concepts subsumés par le concept à classer et dont les pères ne sont pas subsumés par lui).
- (3) : Insertion du nouveau concept dans la hiérarchie.

Les ontologies existantes dans le domaine médical

Dans [36], les auteurs ont introduit une approche basée sur l'ontologie, appelée système d'aide à la décision médicale (MDSS), qui recueille les informations les plus pertinentes de l'utilisateur en fournissant un questionnaire personnalisé en fonction du profil du patient, puis fournit aux patients des conseils personnalisés.

Dans [37], les auteurs ont produit un système basé sur une ontologie formelle qui intègre les informations patient et les données d'alertes extraites des dossiers de santé électroniques afin de classer l'importance des alertes. Cela peut être très utile lorsqu'un grand nombre d'alertes sont générées et envoyées aux médecins, ce qui rend très difficile pour les médecins de gérer toutes ces alertes simultanément.

Dans [38], les auteurs ont produit un système de collecte d'informations (IGS) qui recueille les informations les plus pertinentes liées au patient, ce qui améliore l'approche classique en

personnalisant l'entretien avec chaque patient en fournissant un questionnaire basé sur une ontologie de domaine.

Le besoin d'une ontologie hospitalier

À l'origine, l'ontologie vient de la discipline philosophique qui traite de la nature et de la structure de la réalité. Il a été initialement proposé pour la représentation des connaissances de manière détaillée. Les concepts d'ontologie sont repris et mis en œuvre dans divers systèmes informatiques. La santé est un aspect important de la vie qui nécessite une attention particulière pour apporter des solutions aux problèmes actuels afin de préserver la vie. La santé n'est pas seulement une infirmité ou une absence de maladie, mais un état de plein bien-être mental et physique. Il existe de nombreuses informations relatives à la santé dans l'environnement sanitaire, mais les connaissances pratiques font défaut. En termes de système de santé, même avec beaucoup de données, il n'y a pas d'outils efficaces pour partager les connaissances entre les différents services, mais avec l'aide de systèmes basés sur l'ontologie, les services de santé peuvent être grandement améliorés. Dans ce chapitre, nous présentons une ontologie à base des services hospitaliers.

Construction de l'ontologie HDO dans le domaine Hospitalier :

Dans cette section, nous construisons notre ontologie qui concerne le domaine médical. Nous allons d'abord cerner le domaine d'application qui le domaine hospitalier . Après, nous suivrons les étapes du processus de construction d'ontologie développé dans la section précédente.

2.13.1 Domaine d'application :

Un hôpital est un établissement de soins où un personnel soignant peut prendre en charge des personnes malades ou victimes de traumatismes trop complexes pour être traités à domicile ou dans le cabinet de médecin.

Dans la plupart des pays développés, par rapport au domicile et au cabinet du médecin, le centre hospitalier présente l'avantage d'avoir :

- une hygiène assurée par un personnel de nettoyage formé.
- un accueil permanent et une surveillance continue par du personnel hospitalier médical et paramédical (infirmier ou infirmières, aide-soignant) .
- des équipes de soignants, disposant de compétences particulières (médecins spécialistes) et du matériel (plateau technique) nécessaire à des examens et soins plus poussés qu'au cabinet du médecin (dont en général des blocs opératoires) .
- d'une Pharmacie à Usage Intérieur ayant des dispositifs médicaux et des spécialités pharmaceutiques spécifiques.

En revanche, la présence et le passage de patients porteurs de nombreuses pathologies, et l'usage chronique de médicaments et biocides expose à un risque d'infection nosocomiale.

Certains hôpitaux ont un service des urgences, voire un service mobile d'urgence et de réanimation (SMUR)

Une Consultation médicale :

L'examen médical obéit à des règles simples qui doivent être toujours appliquées, pour élaborer un diagnostic et remettre un traitement si nécessaire. L'examen se déroule comme suit, figure 20 :

- L'interrogatoire
- Examen clinique
- Diagnostic clinique
- Examen paraclinique
- Diagnostic positif
- Traitement
- Evolution et surveillance

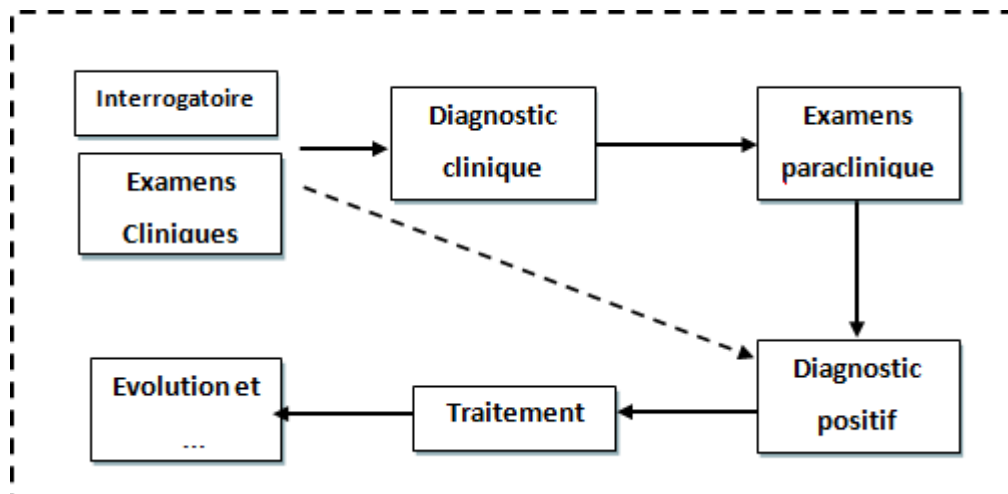


Figure 2.20. Démarche d'une consultation médicale

- **Interrogatoire** : Il s'agit de recueillir des informations de la patiente. L'interrogatoire a trois objectifs principaux :
 - (1) Déterminer le ou les motifs de la consultation,
 - (2) Répertorier les antécédents personnels et familiaux du patient,

(3) Faire décrire la symptomatologie fonctionnelle du patient.

- **Diagnostic clinique :** Le diagnostic clinique apporte une aide précieuse au médecin pour le diagnostic et le suivi de nombreuses pathologies et maladies.
- **Examen para clinique:** Après l'examen clinique, les examens complémentaires permettent d'orienter le diagnostic. Ils ne doivent être demandés qu'après avoir fait un examen clinique minutieux pour confirmer le diagnostic clinique. En fonction de l'hypothèse la plus probable, on doit procéder par des examens : biologiques, radiologiques, autres.
- **Diagnostic positif :** Ce sont des argumentations cliniques et para cliniques permettant d'éliminer les hypothèses incertaines et de confirmer un diagnostic positif.
- **Traitement :** est l'acte de remédier d'un problème de santé.
- **Examen clinique :** permet au médecin ou au clinicien (infirmière, infirmière clinicienne, infirmière praticienne) d'aboutir à un diagnostic. C'est un examen physique à la recherche de signes physiques.

Il doit être systématique et aussi complet que possible. Il permet, en retrouvant ou pas certains signes cliniques, d'apporter des preuves pour appuyer une hypothèse diagnostique.

- **Evolution et surveillance** : la surveillance après le traitement a pour but de déceler les rechutes afin que le traitement ait le plus de chance possible d'obtenir un résultat significatif.

Spécification

Pour commencer le développement de l'ontologie, nous entamons d'abord la phase de spécification qui consiste à établir un document de spécification des besoins. Nous dériverons l'ontologie à construire à travers les cinq aspects suivants:

- **Le domaine de connaissance** : domaine hospitalier
- **Les utilisateurs** : médecins et agents de la santé et ces différents services
- **Les sources d'informations** : les sources d'informations sur lesquelles nous nous sommes basés pour arriver à Conception de l'ontologie en premier lieu les experts du Domaine, mais aussi d'après des livres de médecine [29], et des encyclopédies médicales [30].
- **La portée de l'ontologie** : Cet aspect consiste à déterminer à priori la liste des termes de l'ontologie d'application (les plus importants), parmi ces termes, nous pouvons citer : structure-santé, maladie, personne, Traitement,....etc

Conceptualisation

Une fois la majorité des connaissances acquise, on doit les organiser et les structurer en utilisant des représentations intermédiaires semi-formelles qui sont faciles à comprendre et indépendantes de tout langage d'implémentation. Cette phase contient plusieurs étapes qui sont :

- Construction du glossaire défermes
- Construction du diagramme de relations binaires
- Dictionnaire de concepts
- Tableaux des relations binaires
- Tableaux des attributs
- Tableaux des instances

Construction de glossaire de termes

Ce glossaire recueille et décrit tous les termes qui sont utiles et potentiellement utilisables dans l'ontologie finale. Le tableau ci-dessous (Tableau 3) fournit une liste détaillée des différents termes utilisés dans l'ontologie :

Tableau2.3. Glossaire de termes

Thème	Description
Departements and Units	Ce sont des structures agrégées constituant l'organisation d'un hôpital. Un service peut être constitué d'une seule unité fonctionnelle si cette dernière ne présente aucune complémentarité directe avec d'autres unités de même discipline médicale ou s'il n'existe pas d'unité fonctionnelle ayant la même activité
Equipment	L'équipement médical est constitué de

	l'appareillage destiné à aider le diagnostic et le traitement de problèmes médicaux. Il est en général conçu selon des règles rigoureuses de sécurité.
Department of cancerology	Le département prend en charge les patients atteints de tumeurs solides ou hématologiques, dans les différentes phases de la maladie (traitement, suivi post thérapeutique, phase palliative) et ce, pour l'ensemble des organes.
Humanresources	Tous humain qui fait partie de l'ontologie .
Department of medical imaging and nuclear medicine	il fournit des services médicaux aux patients atteints de divers troubles en utilisant diverses modalités d'imagerie, telles que la tomodensitométrie, l'IRM, l'échographie et la tomographie par émission de positrons. La radiologie interventionnelle est également menée pour une thérapie mini-invasive ainsi que pour le diagnostic
Pharmacy	Faire l'approvisionnement et la dispensation des produits de santé (médicaments et dispositifs médicaux) pour les patients hospitalisés et ou ambulatoires
Departement of surgery	Le service de chirurgie polyvalente assure la prise en charge chirurgicales de plusieurs spécialités
Head and Neck department	Il fournis des services medicauxau personnes souffrant de troubles de l'oreille, du nez, de la gorge, de la tête et du cou .
Nursing Administration	L'administration infirmière peut être mieux définie comme la gestion stratégique du personnel infirmier, des soins aux patients et des ressources

	de l'établissement grâce à l'appui de politiques de réglementation
Outpatient Departments	Un service ambulatoire ou une clinique externe est la partie d'un hôpital conçue pour le traitement des patients ambulatoires, des personnes ayant des problèmes de santé qui se rendent à l'hôpital pour un diagnostic ou un traitement, mais qui n'ont pas à ce moment besoin d'un lit ou d'être admises pour des soins de nuit
Check up center	Le bilan est un examen physique général, qui donne une évaluation générale de l'état de santé d'une personne en scannant les maladies possibles qui n'ont pas causé de plaintes jusqu'à présent.
Critical health Unit	également appelé unité de thérapie intensive ou unité de traitement intensif (UIT) ou unité de soins intensifs (CCU), est un service spécial d'un hôpital ou d'un établissement de soins de santé qui fournit des soins intensifs.
Dialysis center	unité de dialyse Une salle ou une partie de celle-ci

	pour la dialyse de patients atteints d'une maladie rénale avancée.
Doctor	Un médecin travaille dans une structure de santé, examine des patients, prescrit des traitements et dépiste des maladies.
Patient	Un patient doit au moins se rendre dans une structure de santé, et être examiné par un médecin.
Traitement	Traitement peut être préventif ou curatif.
Maladie	Peut être infectieuse, endocrinienne, cardiaque, tumorale, gynécologique, dermatologique, ou hématologique.
Nurse	Personne qui, par profession, soigne des malades et s'en occupe, sous la direction des médecins.
Midwife	Personne ayant pour rôle la surveillance, les soins et le conseil des femmes pendant l'accouchement et dans les suites de couches.

Classification de concepts

La hiérarchie de classification de concepts démontre l'organisation des Concepts de l'ontologie en un ordre hiérarchique qui exprime les relations sous classe – super classe. En utilisant la relation « Sous classe de » entre les classes pour définir leurs classifications, la classe C1 est une sous classe de la classe C2 si et seulement si toute instance de la classe C1 est une instance de la classe C2, par exemple la classe (Equipment) est une sous classe de la classe(ressources).

Conception d'une ontologie dans le domaine Hospitalier :

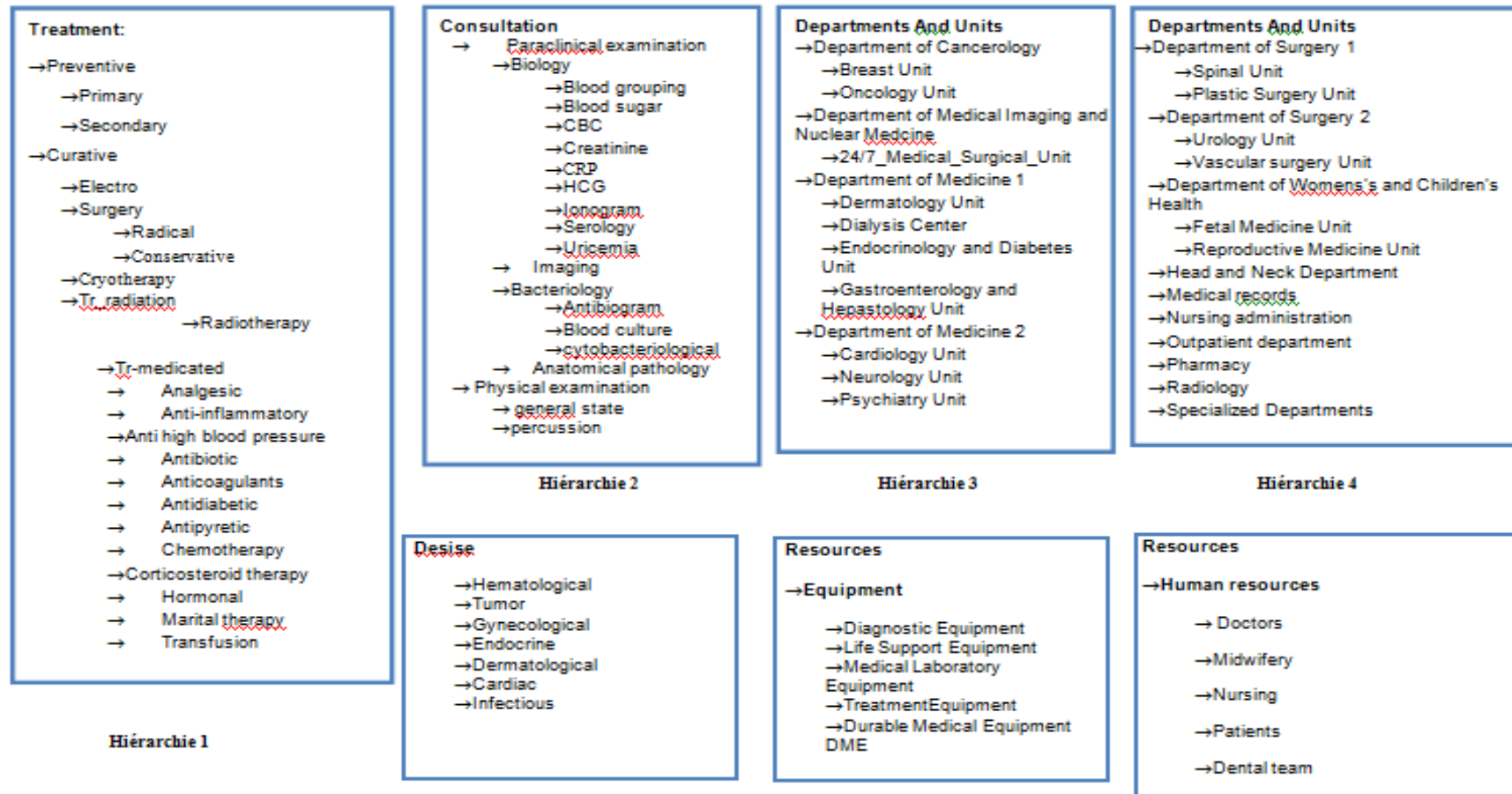


Figure 2.19. Conception d'une ontologie dans le domaine hospitalier

Dictionnaire de concepts

Dans cette étape nous allons donner une description formelle des concepts qui ont été présentés dans la hiérarchie des classes. Ce processus correspond à la création du dictionnaire de concepts accordé à METHONTOLOGY. Dans ce dictionnaire, nous définissons pour chaque concept : les instances, les attributs, les relations dont la source est ce concept, le tableau 4 représente le dictionnaire des concepts pour l'ontologie définie par la hiérarchie précédente.

Tableau2.4 Table dictionnaire des concepts.

Concepts	Description	Atributs	Instances
Department and Units	Services hospitalier	Dénomination Adresse Tel Fax	
Equipments	outils médicales	Type Durabilité	Dialysis machine Hospitalbeds
Patient		Nom Prénom Nom-JF	

	-	Tel Adresse Age Etat	
Doctors	médecin	Grade-médecin Tel Fax Adresse Email Spécialité-médecin	
Desises	maladie	Nom-maladie Pronostic Type-maladie	
Midwifery	Sage-femme	Grade SF	midwife
Tr-medicated	Remède, médication	Nom-traitement Nom-commercial Forme Dosage	Paracétamol Voltaire

		Posologie Mode-admin Durée-traitement Type-Med Effet secondaire	
Tumor	cancer	Name-disease Beginning-disease History Prognosis Complication Type-disease	

Tableau de relations binaires

Les relations binaires sont représentées sous forme de propriétés ou attributs qui lient un concept à un autre, ce sont des ‘attributs de type instance’ : c'est-à-dire les attributs ayant pour type de valeur Instance [31].

Pour chaque relation dont la source est dans l'arbre de classification de concepts, nous définissons : son nom, le nom du concept source, le nom du concept cible, la cardinalité et le nom de la relation inverse; le tableau 5 illustre la spécification des relations binaires entre les différentes hiérarchies pour notre ontologie.

Tableau2.5 Table des relations binaires.

Nom de Relation	Concept source	Conceptible	Cardinalité
Countains	Départements and units	resources	(1, n)
Uses	Doctor	equipment	(1, n)
Treated With	patient	equipment	(1, n)
Treats	Doctors	patient	(1, n)
Cares About	Nurse	Patient	(1, n)
Gets	Patient	Treatment	(1, 1)
Needs	Patient	antibiotic	(1, n)

Diagnosed	Patient	Covid-19	(1, n)
sort	Healthcare Professione 1	patient	(1, n)

Tableau des attributs

Les attributs sont des propriétés qui prennent leurs valeurs dans les types prédéfinis (String, Integer, Boolean, Date...). Par exemple le concept patienta comme attributs : Name, Age, Adress...

Pour chaque attribut apparaissant dans le dictionnaire de concepts nous spécifions: son nom et son type Le tableau 6 spécifie ces informations pour chaque attribut.

Tableau2.6 Table des attribut

Nom de l'attribut	Type
Name	string
Last Name	string

Adress	nmtoken
Number	Integer
Email	nmtoken
Age	Integer
State	string
Treatment-name	string
Commercial-name	nmtoken

Conclusion :

Nous avons donné un processus de construction de l'ontologie médical HDO qui est basé sur la méthodologie « METHONTOLOGY ». Cette dernière est constituée de cinq phases successives. La spécification, première phase de ce processus, a permis l'analyse des différentes sources de données concernant le domaine de l'ontologie à construire, et d'établir un document de spécification pour celle-ci, exprimé en langage naturel. Ce document décrit le domaine, la portée, les utilisateurs et les caractéristiques de l'ontologie.

Nous avons abouti à un ensemble de représentations intermédiaires semi- formelles, et nous avons obtenu une ontologie conceptuelle. Les résultats de l'étape de conception vont servir pour l'opérationnalisation de cette ontologie.



Application et implémentation

De l'ontologie HDO

Au service COVID-19

Etude de cas

Depuis la déclaration du premier cas d'infection SARS-Cov2 le 08 Décembre 2019, la situation a évolué vers la déclaration par l'OMS de la pandémie le 11 Mars 2020.

Le Covid-19 est un virus à ARN simple brin enveloppé. Il est responsable d'infections Respiratoires principalement basses à manifestations variables. Des formes graves se présentent avec un tableau d'insuffisance respiratoire aiguë pouvant évoluer vers un syndrome de détresse respiratoire aiguë.

Domaine d'application : COVID-19

Modes de transmission :

La transmission est interhumaine type air et gouttelettes. La contamination se fait principalement par voie aérienne.

Le dépôt des gouttelettes sur les surfaces inertes est un moyen de transmission par les mains contaminées.

La projection directe de grosses gouttelettes sur les muqueuses du visage, de la sphère ORL et les conjonctives et par les mains contaminées sont les principaux modes de contamination. Il est à noter un risque élevé de transmission chez les consommateurs de Narguilé La transmission aérienne en milieu de soins par petites gouttelettes semble possible à l'occasion de manœuvres de réanimation (intubation, ventilation non invasive VNI, traitement par aérosols, aspiration trachéale). La majorité des infections à Covid-19 ne présentent pas de signes de gravité en particulier chez les enfants et les moins de 30 ans. Toutefois, les enfants sont considérés comme un vecteur viral.

Symptômes et signes évocateurs:

La présentation clinique typique est celle d'un syndrome grippal. Les symptômes décrits comprennent :

- Des signes respiratoires : une toux, une dyspnée, des difficultés respiratoires
- Une fièvre (toutefois, son absence n'élimine pas le diagnostic en présence d'une forte suspicion par ailleurs)
- Des frissons
- Une asthénie, une fatigabilité
- Des céphalées
- Des arthro-myalgies

Les signes suivants sont moins fréquents :

- Une sècheresse de la gorge
- Une rhinorrhée
- Des douleurs thoraciques
- Une diarrhée
- Des nausées et des vomissements

L'absence de ces signes n'élimine pas la contamination.

Parcours de la prise en charge du patient potentiellement atteint par le COVID-19 :

les instances officielles de la Santé ont présenté à ce sujet des algorithmes, capables de diagnostiquer et prédire l'évolution de la maladie, et qui résument le parcours des patients potentiellement atteints.

Pour le parcours du patient consultant en première ligne, dans le public ou le privé, le professionnel de la santé doit effectuer un tri initial, et puis mesurer la température du patient en lui posant un questionnaire.

Il s'agit d'une « anamnèse poussée », une collecte d'informations utiles pour le diagnostic, à travers des questions sur l'historique du patient, séjour à l'étranger ou probabilité de contagion horizontale/ transversale. « L'anamnèse » retrace également les antécédents médicaux du patient pour pouvoir le classer après.

il faut se protéger et protéger les collaborateurs si le cas est suspect, et le médecin effectuera l'examen clinique complet, en vérifiant l'existence d'une détresse respiratoire par la prise de la saturation pulsée en oxygène (SpO2) ».

En cas de suspicion de Covid-19, et si le patient présente une forme sévère ou grave des symptômes indiqués, le corps médical doit appeler le SAMU sur le 190, pour probablement l'hospitaliser. Le transfert du malade dans une zone d'isolement, accompagné par le médecin et l'infirmier, serait obligatoire.

D'autre part, s'il présente une forme légère des symptômes, une quarantaine à domicile et des conseils lui seront recommandés.

Si l'anamnèse et la prise de température concluent à un cas non suspect COVID 19, le médecin effectue une consultation normale.

Parcours du patient consultant en première ligne (Publique et privée)

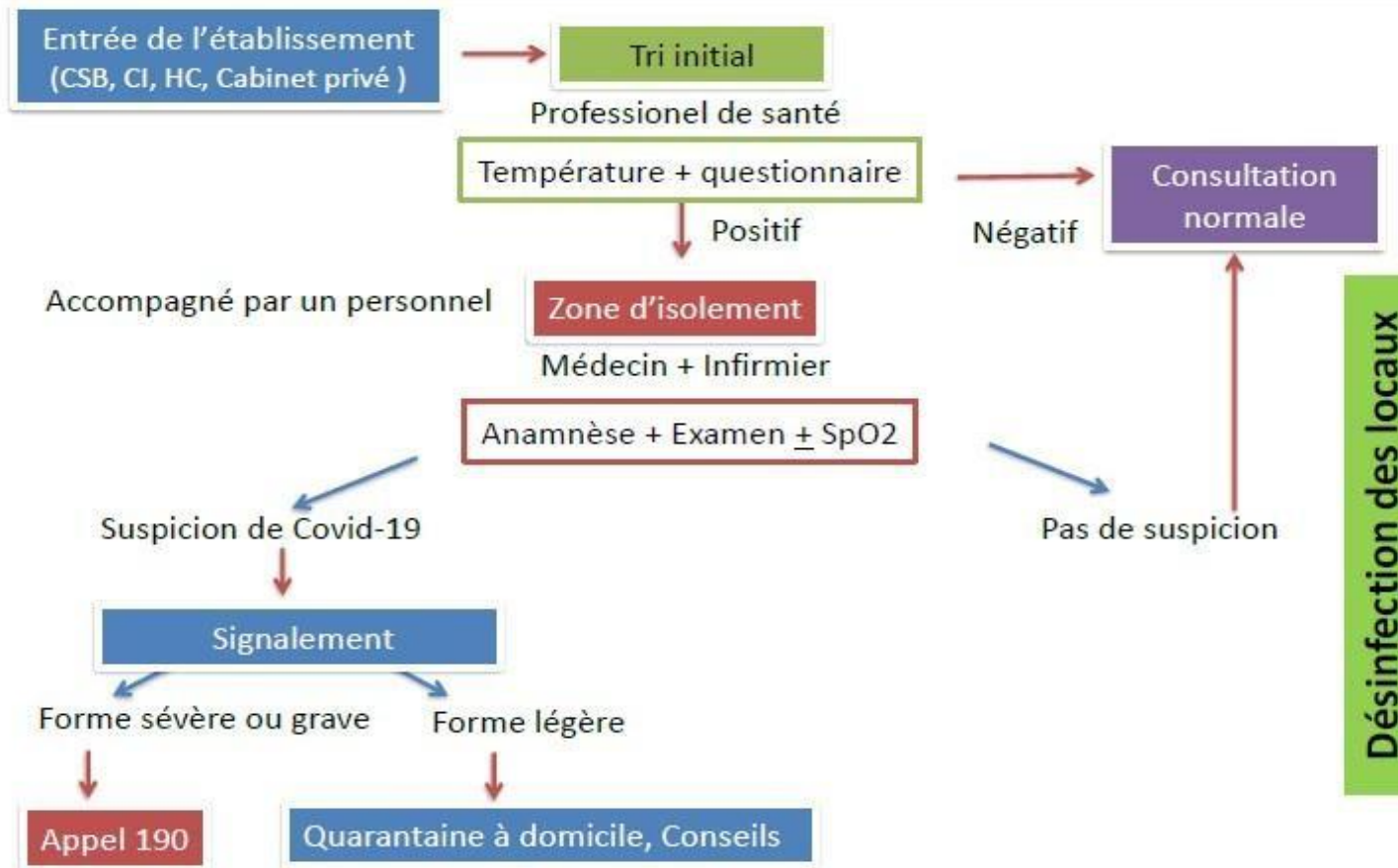


Figure 21 : Parcours du patient Consultant en première ligne

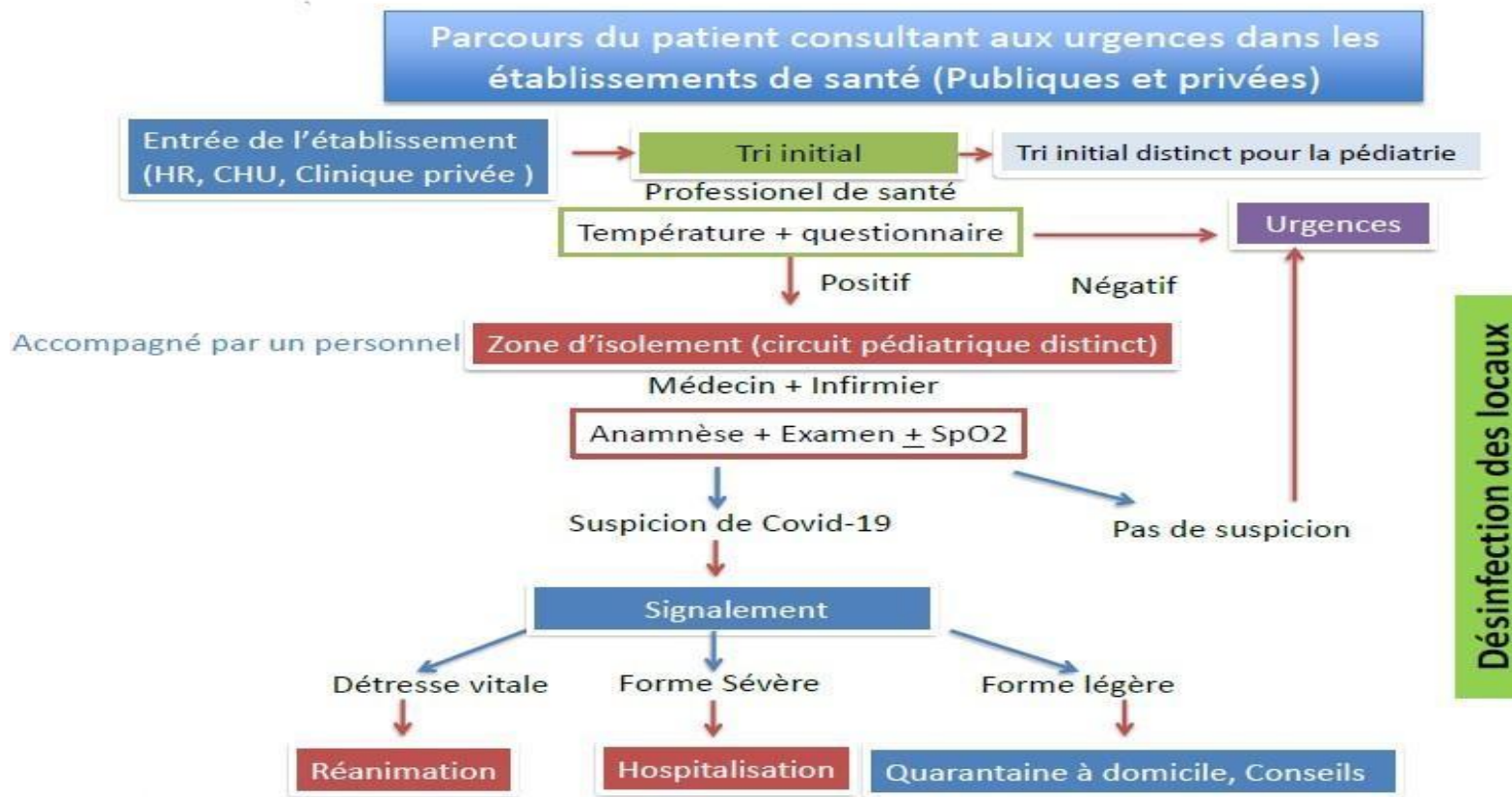


Figure 22 : Parcours du patient consultant au urgences dans les établissements de santé COVID-19

Conceptualisation

Construction de glossaire de termes :

Ce glossaire recueille et décrit tous les termes qui sont utiles et potentiellement utilisables dans cette partie de l'ontologie . Le tableau ci-dessous (Tableau 6) fournit une liste détaillée des différents termes utilisés dans l'ontologie :

Tableau 3.7 Glossaire des termes (COVID-19)

Thème	Descriptio n
Covid-19	
Symptoms	Un symptôme est un signe, un trouble observable par un patient, qui est la manifestation d'une maladie et de son évolution
fever	La fièvre est une élévation de la température corporelle interne. On parle de fièvre au-delà de 38 °C.
PCR Test	Le test PCR permet de savoir si une personne est contaminée par le SARS-CoV-2 au moment où le test est réalisé
diagnostic	identification d'une maladie à partir de ses symptômes
Typical scan	scanners thoraciques
quarantain	a quarantaine consiste à isoler des personnes, des animaux, ou des

e	végétaux durant un certain temps, en cas de suspicion de maladies contagieuses,
---	---

Dictionnaire de concepts (COVID—19)

Tableau 3.8 Dictionnaire des concepts

Nom de concept	Concepts synonymes	Attributs
Patient	Humain	Name Age Adress Temperature
Healthcare Professionel	Clinicien, thérapeute	Nom Prénom
Doctor	Docteur, clinicien, thérapeute.	Grade-doctor numberl Fax Adress Email Speciality-doctor

PCR	Test	Positif Negatif
Covid-19	maladie	Severe forme Normal forme
Therapy protocol	traitement	

Tableau de relations binaires (COVID-19)

Tableau 3.9 relations binaires (COVID-19)

Nom de Relation	Concept source	Concept cible	Cardinalité
Check	Healthcare professionnel	patient	(1, n)
Sort	Healthcare professionnel	patient	(1, n)
Gets	patient	reported	(n, 1)
Call	Healthcare professionnel	190	(n, 1)
Pass	patient	RTPC	(1, 1)
Does	Patient	scan	(n, 1)
Diagnosed with	patient	COVID-19	(n, 1)
Get considered as	Patient	Respiratory desis	(n, 1)
Starting	doctor	Therapy protocol	(n, 1)
Repass	Patient	PCR	(1, n)

Représentation sémantique des règles avec le langage SWRL

L'ontologie est formulée dans le langage de représentation de connaissance OWL cette implémentation est réalisée par l'outil Protégé et validée par le raisonneur Pellet. Le langage OWL pose certains problèmes à cause de quelques limitations en particulier dans la définition des propriétés. Pour résoudre ce problème, SWRL, une extension à OWL, a été développé pour rendre le langage encore plus expressif pour la description de ces propriétés.

L'enrichissement de l'ontologie par des règles consistant à injecter un système à base de règles dans la base de connaissance, permet d'ajouter via les règles SWRL des axiomes sur les propriétés décrites dans le fragment d'ontologie.

Dans notre étude de cas , nous avons enregistré une série de règles qui enrichie les modèles ontologiques de la description des connaissances dans le domaine hospitalier. Ces règles essentiellement représentent les connaissances de base acquises par les experts dans le domaine grâce aux entrevues et des questionnaires dans le processus d'enregistrement et d'analyse du système. Ces règles peuvent être divisées en 3 catégories:

Parcours d'un Patient consultant en Première Ligne :

Ces règles permet de diriger la patient dans son parcours de consultation en première ligne .Par exemple, une telle règle est la suivante:

```
Healthcareprofessional(?h) ^ Patients(?p) ^ check(?h, ?p) ^ Temperature(?p, "positif") → isolate(?h, ?p)
```

Cette règles présente le rôle du professionnel de santé, il vérifie la température du patient si elle est élevé le patient doit être amené en isolation .

Parcours d'un Patient consultant au urgences

Dans cette catégorie représente les règles qui dirige u patient en parcours de consultation d'urgence.

Un exemple de ces règles est le suivant :

emergency(?e) ^ Patients(?p) ^ Doctors(?d) ^ gets(?p, ?d) ^ examination(negatif) → goesTo(?p, ?e)

Règles décisionnelles COVID-19 :

Ces règles aident à prendre la bonne décision de traitement selon le cas du patient Voici un exemple de regles :

Patients(?p) ^ COVID-19(severeForme) ^ hospitalisation(?h) -> goesTo(?p, ?h) Dans cette règle si un patient est en état grave il doit être hospitaliser.

Tableau 3.10 Les règles SWRL

N°	Règle SWRL
01	emergency(?e) ^ Healthcare_professionel(?h) ^ Patients(?p) ^ check(?h, ?p) ^ Temperature(?p, "negatif") -> goesTo(?p, ?e)
02	Patients(?p) ^ PCR(?T) ^ PCR(negatif) ^ with(?p, negatif) ^ RespiratoryDesis(?R) -> GetConsideredAs(?p, ?R)
03	Patients(?p) ^ COVID-19(?c) ^ diagnosedWith(?p, ?c) ^ TherapyProtocol(?tp) -> starting(?p, ?tp)
	Patients(?p) ^ COVID-19(?c) ^ diagnosedWith(?p, ?c) ^ TherapyProtocol(?T) -> starting(?p, ?T)

04	
05	emergency(?e) ^ Patients(?p) ^ Doctors(?d) ^ gets(?p, ?d) ^ examination(negatif) -> goesTo(?p, ?e)
06	Patients(?p) ^ COVID-19(severeForme) ^ hospitalisation(?h) -> goesTo(?p, ?h)
07	suggestiveSymptomes(?s) ^ Patients(?p) ^ has(?p, ?s) ^ PCR(?t) -> pass(?p, ?t)
08	suggestiveSymptomes(?s) ^ Patients(?p) ^ dontHave(?p, ?s) ^ RespiratoryDesis(?r) -> GetConsideredAs(?p, ?r)
09	Patients(?p) ^ PCR(positif) ^ with(?p, positif) ^ COVID-19(?c) -> diagnosedWith(?p, ?c)
10	Patients(?p) ^ PCR(?T) ^ PCR(negatif) ^ with(?p, negatif) -> repass(?p, ?T)
11	Patients(?p) ^ PCR(?T) ^ PCR(positif) ^ with(?p, positif) ^ COVID-19(?c) -> diagnosedWith(?p, ?c)
12	Healthcare_professionel(?h) ^ Patients(?p) ^ check(?h, ?p) ^ Temperature(?p, "positif") -> isolate(?h, ?p)

13	<code>Consultation(?c) ^ Patients(?p) ^ Doctors(?d) ^ gets(?p, ?d) ^ examination(negatif) -> gets(?p, ?c)</code>
14	<code>Patients(?p) ^ COVID-19(positif) ^ COVID-19(severeForme) ^ Healthcare_professionnel(?h) -> call(?h, ?p)</code>

Semantic Query Enhanced Web Rule Language: SQWRL

SQWRL « *Semantic Query-Enhanced Web Rule Language* » est un langage de requête fondé sur le langage *SWRL* pour les ontologies faites en *OWL* (O'Connor & Das, 2009). Il fournit un ensemble de fonctions semblable à du « *Sql* » pour extraire des connaissances d'une ontologie en *OWL*. Ces requêtes peuvent être écrites via la *SWRLTab* de PROTÉGÉ.

Implémentation De l'Ontologie

Après avoir conçu l'ontologie, il nous reste maintenant à l'implémenter. Pour cela, nous disposons de nombreux langages et outils qui permettent d'implémenter l'ontologie proposée.

Notre choix porte sur le langage *OWL* qui est un langage de codification utilisé pour implémenter l'ontologie en *OWL* (un langage de définition d'ontologie pour le Web), et cela pour toutes les fonctionnalités sémantiques que permet *OWL* et qui sont plus riches que celles des langages *RDFS* & *DAML+OIL*.

Ainsi, nous utiliserons *SWRL* comme langage de règles, pour définir des règles non exprimable en *OWL*, ces dernières sont dites « sucre syntaxique » facile à écrire et apporteront une plus haute expressivité.

Pour l'implémentation du système de raisonnement, nous avons choisi des outils standards du Web sémantique : Protégé-*OWL* et ses plugins.

Outils d'implémentation

Protégé et plugins

PROTEGE-OWL est une interface modulaire, développée au Stanford Medical Informatics de l'Université de Stanford⁷, permettant l'édition, la visualisation, le contrôle (vérification des contraintes) d'ontologies, l'extraction d'ontologies à partir de sources textuelles, et la fusion semi-automatique d'ontologies.

PROTEGE-OWL autorise la définition de méta-classes, dont les instances sont des classes, ce qui permet de créer son propre modèle de connaissances avant de bâtir une ontologie. Son interface, très bien conçue, et son architecture logicielle permettant l'insertion des plug-ins.

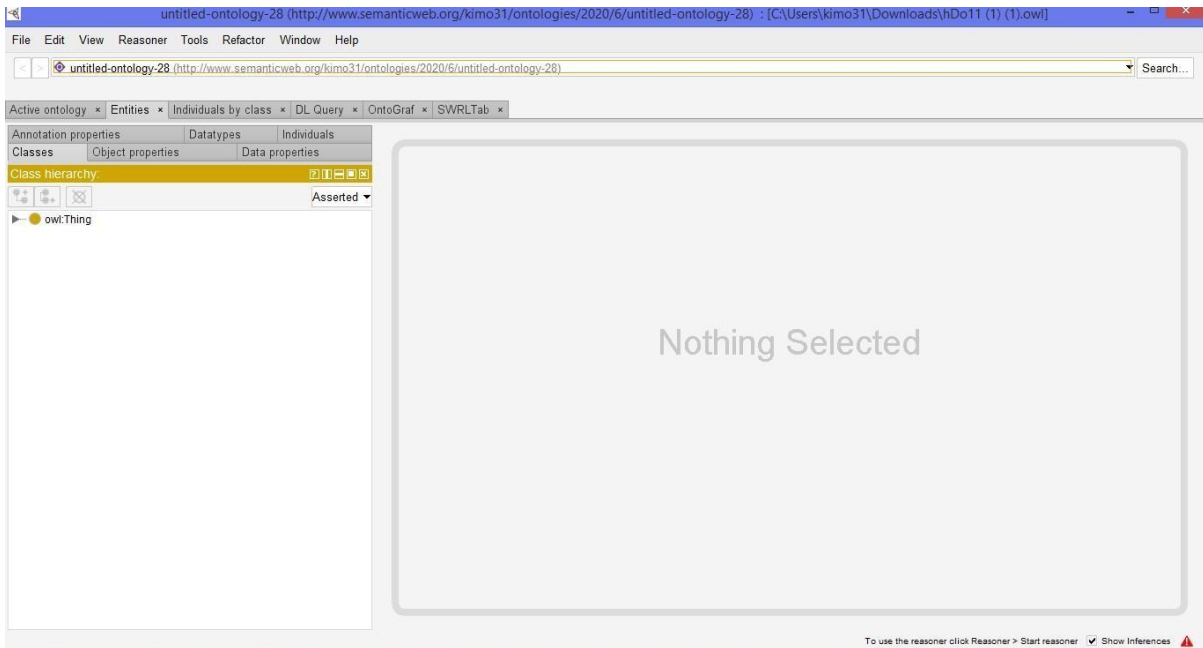


Figure 3.23. Interface de PROTEGE OWL

Plugins utilisés

SWRL Tab

L'éditeur SWRL est un plugin de Protégé OWL permettant l'édition interactive des règles SWRL. Les utilisateurs peuvent créer, modifier, lire et écrire les règles SWRL.

La figure montre que l'éditeur SWRL permet aux utilisateurs de saisir des règles complètement sous forme de texte. Toutefois, il permet également aux utilisateurs de sélectionner des classes OWL, propriété et individus de la base de connaissances (Ontologie) chargée et les insérer dans la règle en cours de création.

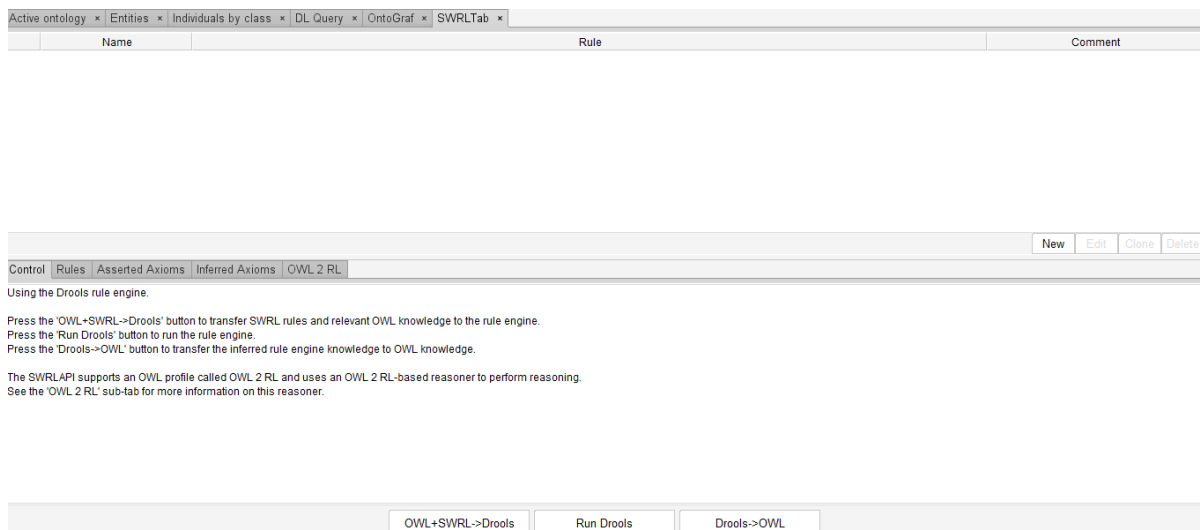


Figure 3. 24. L'éditeur de règles SWRL

Création de l'ontologie avec PROTEGE OWL

Création de classes et hiérarchie de classe

Nous commencerons tout d'abord par la création des concepts spécifiés dans l'étape de conceptualisation. PROTEGE OWL nous offre également un moyen de construire la hiérarchie de concepts, la figure présente comment procède-t-on pour créer un concept.

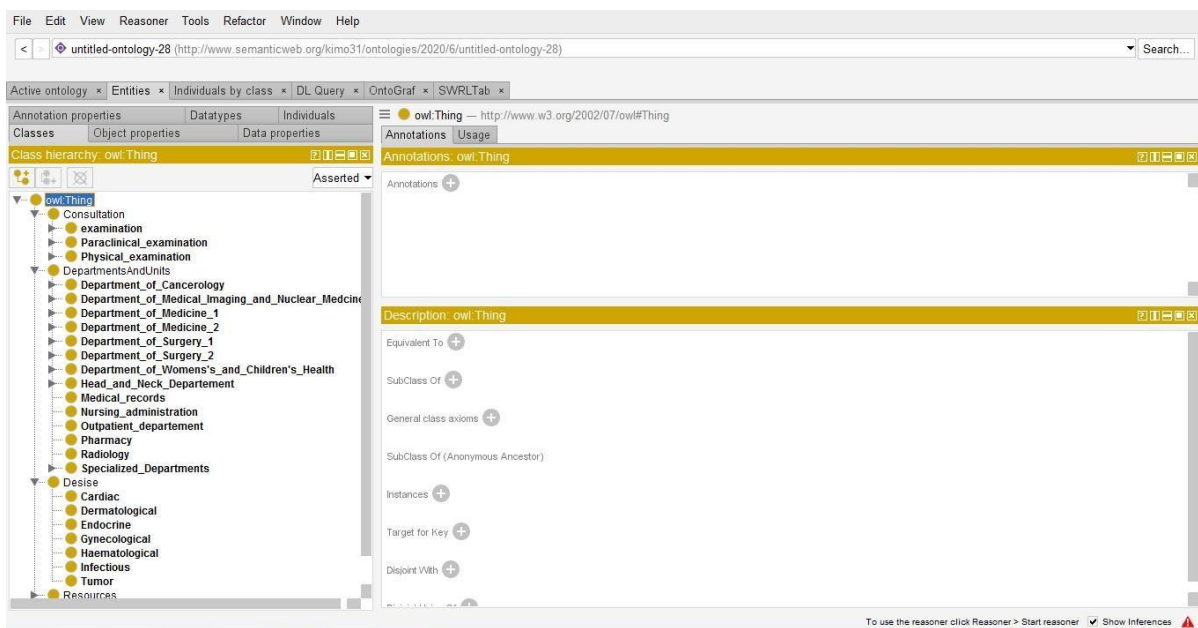


Figure 3.25. Création de classes

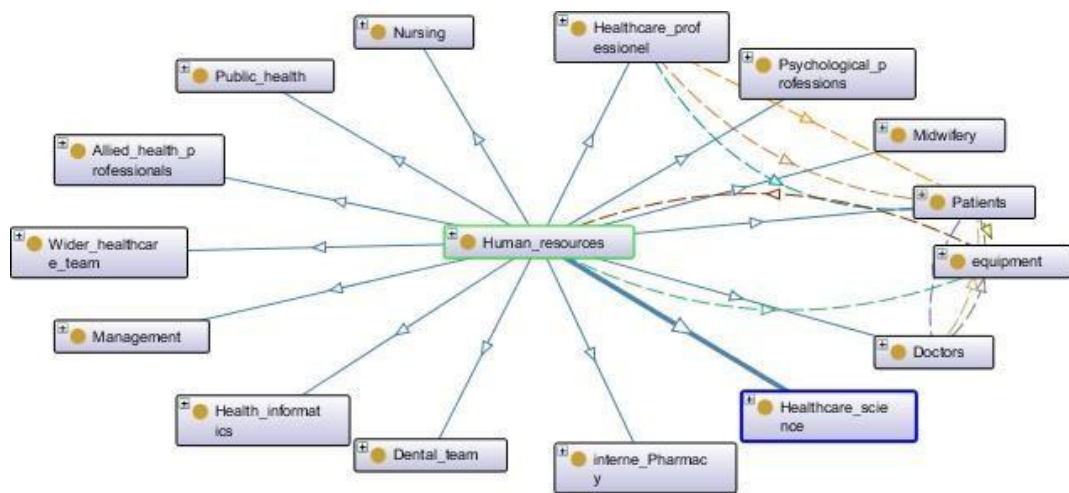
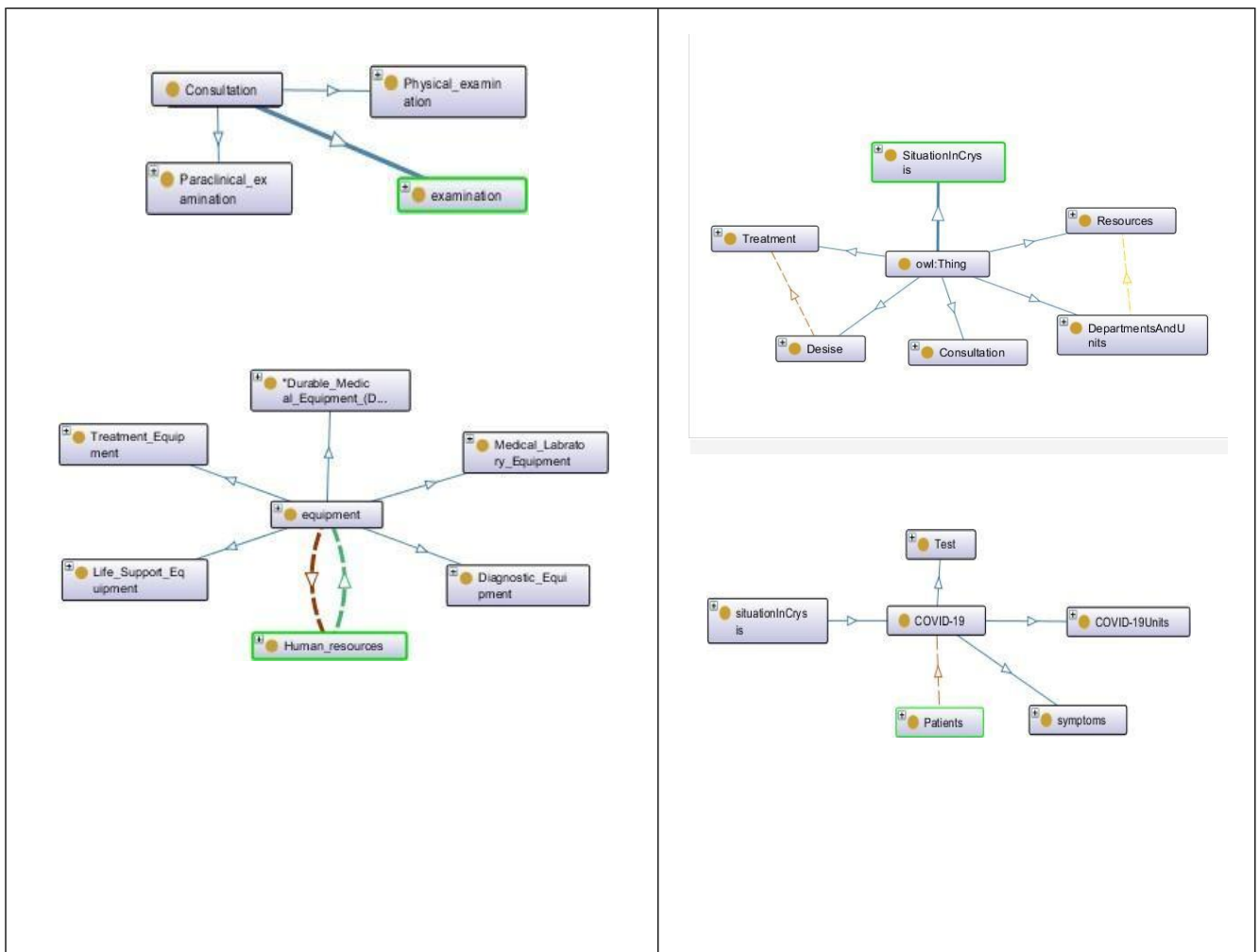


Figure 3.26. Structures des classes principaux et hiérarchie des classes

Création de propriétés :

Après avoir construit les concepts, nous allons maintenant créer les propriétés pour chacun d'eux, les attributs vont être créés sous PROTEGE OWL par *'dataProperty'* et les relations par *'objectProperty'*. Les propriétés d'une classe sont les propriétés héritées de sa superclasse, plus ses propres propriétés privées.

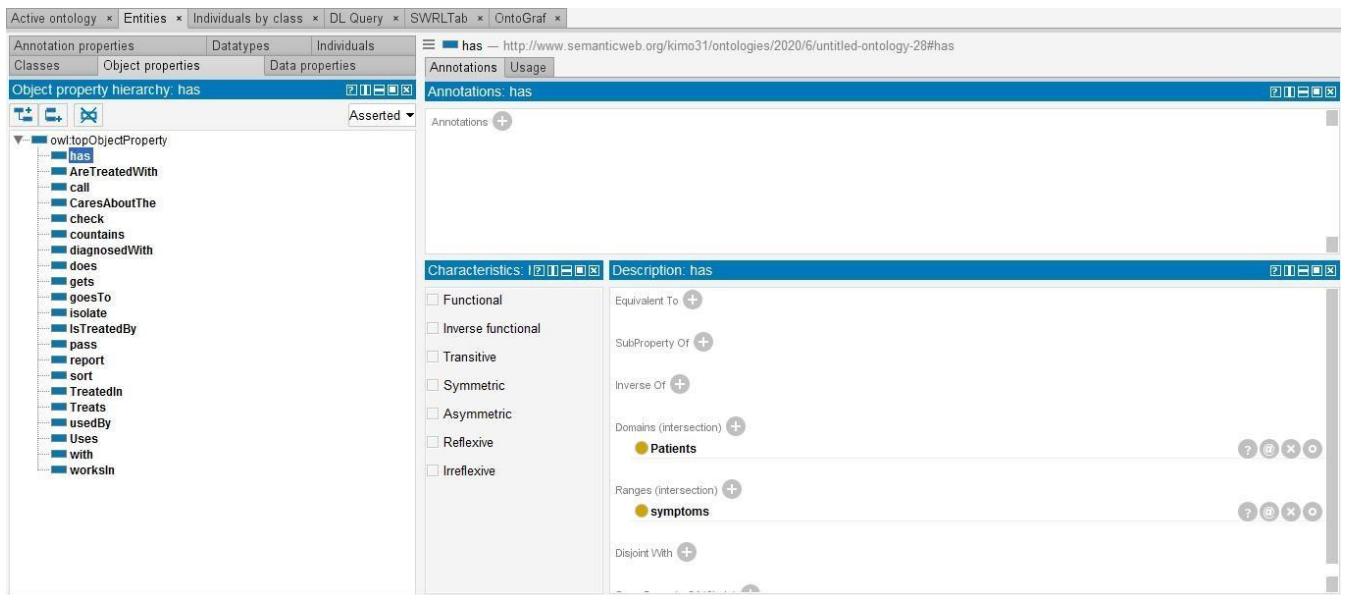
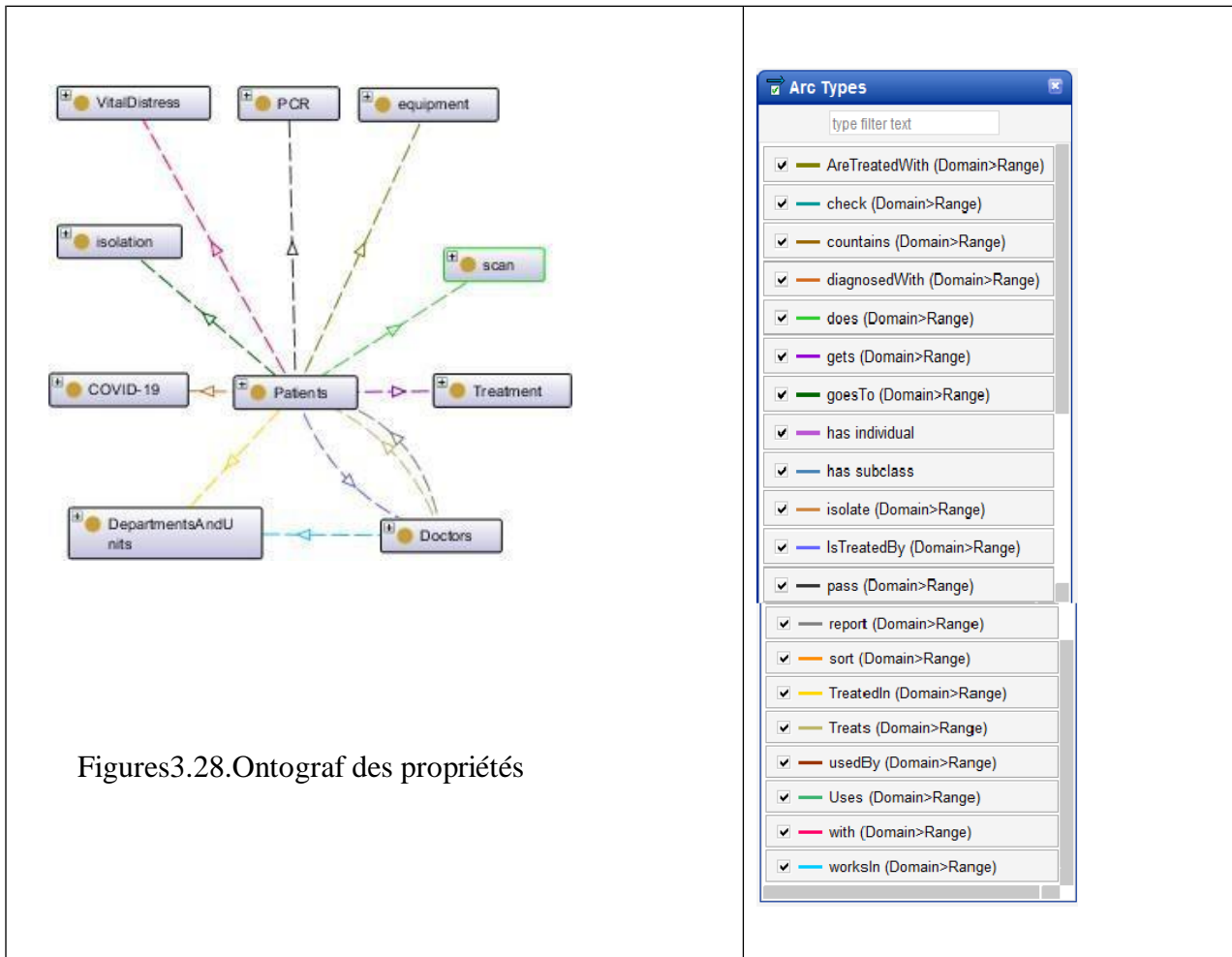


Figure 3.27. Création de propriétés



Création d'attributs

La création des attributs se fait en cliquant au-dessus le bouton *'dataTypeProperty'* : ce qui génère la fenêtre comme la montre la figure ci-dessous ; qui permet de fournir les champs pour remplir le nom, le domaine, le type de valeurs de l'attribut et le type XML associé

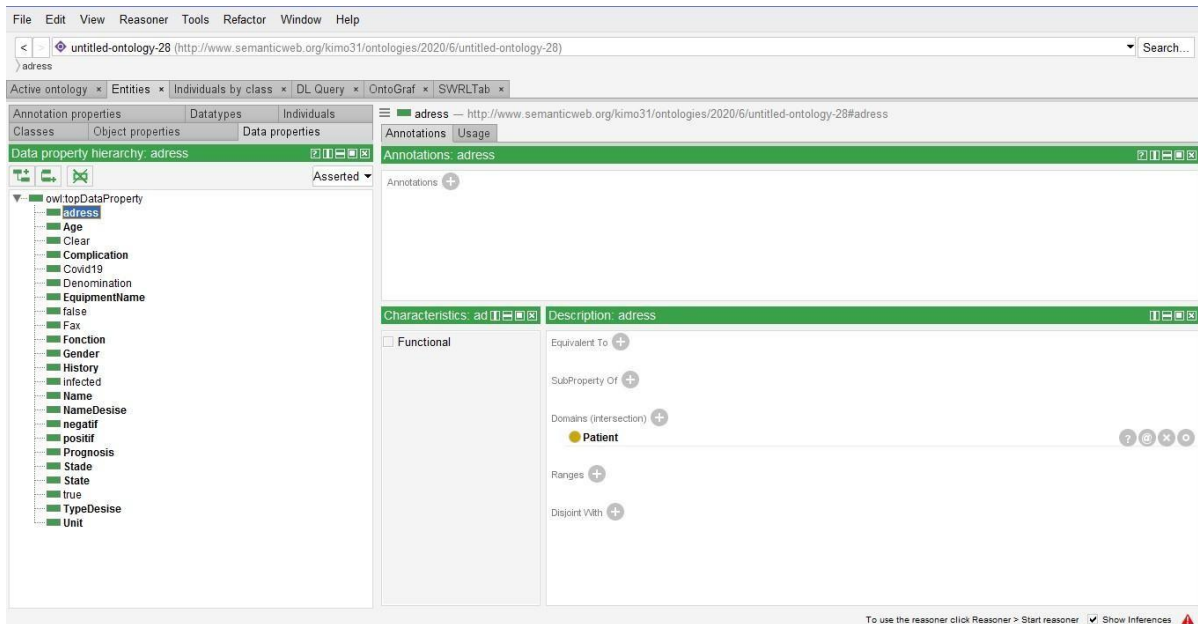


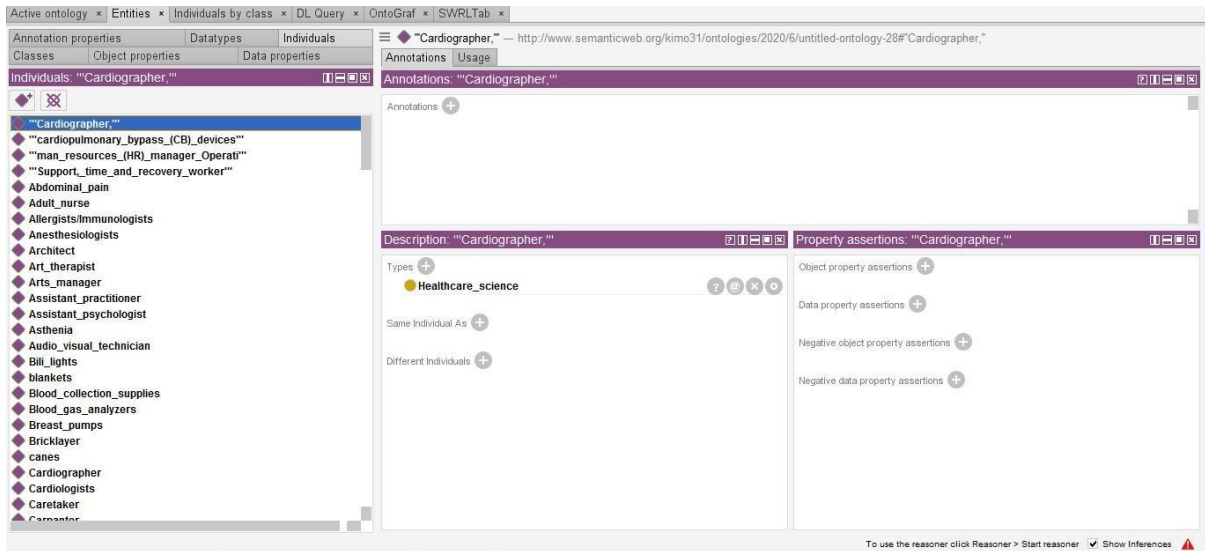
Figure 3.29. Créations d'attributs

Création d'instance :

La dernière étape consiste à créer les instances des classes dans la hiérarchie. Définir une instance individuelle d'une classe exige

1. choisir une classe
2. créer une instance individuelle de cette classe
3. la renseigner avec les valeurs des attributs.

Par exemple, la figure montre la création d'un individu de la classe Healthcare science



Figures3.30. création des instances

Les règles SWRL

Les règles sont nécessaires pour exprimer des dépendances entre des concepts ou des propriétés, ou des contraintes de cohérence.

Grace à SWRL Tab, les règles SWRL sont facilement intégrées à l'ontologie OWL. Les règles peuvent être entré directement du clavier. Cependant, l'utilisateur peut sélectionner les entités OWL directement du panneau des icônes.

Name	Rule	Comment
swrl-1	emergency(?e) ^ Healthcare_professional(?h) ^ Patients(?p) ^ check(?h, ?p) ^ Temperature(?p, "negatif") -> goesTo(?p, ?e)	check patient Temperature
swrl-10	Patients(?p) ^ COVID-19(positif) ^ COVID-19(NormalForm) ^ QuarantineAtHome(?q) -> does(?p, ?q)	
swrl-11	Patients(?p) ^ COVID-19(positif) ^ COVID-19(severeForme) ^ Healthcare_professional(?h) -> call(?h, ?p)	
swrl-12	Consultation(?c) ^ Patients(?p) ^ Doctors(?d) ^ gets(?p, ?d) ^ examination(negatif) -> gets(?p, ?c)	
swrl-13	suggestiveSymptomes(?s) ^ Patients(?p) ^ dontHave(?p, ?s) ^ RespiratoryDesis(?r) -> GetConsideredAs(?p, ?r)	
swrl-14	Patients(?p) ^ PCR(positif) ^ with(?p, positif) ^ COVID-19(?c) -> diagnosedWith(?p, ?c)	
swrl-15	Patients(?p) ^ PCR(?T) ^ PCR(negatif) ^ with(?p, negatif) -> repass(?p, ?T)	
swrl-19	Patients(?p) ^ PCR(?T) ^ PCR(positif) ^ with(?p, positif) ^ COVID-19(?c) -> diagnosedWith(?p, ?c)	
swrl-2	Patients(?p) ^ PCR(?T) ^ PCR(negatif) ^ with(?p, negatif) ^ RespiratoryDesis(?R) -> GetConsideredAs(?p, ?R)	patient PCR negatif
swrl-3	Patients(?p) ^ COVID-19(?c) ^ diagnosedWith(?p, ?c) ^ TherapyProtocol(?tp) -> starting(?p, ?tp)	therapy protocol
swrl-4	emergency(?e) ^ Patients(?p) ^ Doctors(?d) ^ gets(?p, ?d) ^ examination(negatif) -> goesTo(?p, ?e)	patient examination
swrl-5	Patients(?p) ^ COVID-19(severeForme) ^ hospitalisation(?h) -> goesTo(?p, ?h)	patient severeForme
swrl-6	suggestiveSymptomes(?s) ^ Patients(?n) ^ has(?n, ?s) ^ PCR(?n) -> nass(?n, ?n)	Patient with suggestive symptoms

Control Rules Asserted Axioms Inferred Axioms OWL 2 RL

Using the Drools rule engine.

Press the 'OWL+SWRL->Drools' button to transfer SWRL rules and relevant OWL knowledge to the rule engine.
 Press the 'Run Drools' button to run the rule engine.
 Press the 'Drools->OWL' button to transfer the inferred rule engine knowledge to OWL knowledge.

The SWRLAPI supports an OWL profile called OWL 2 RL and uses an OWL 2 RL-based reasoner to perform reasoning.
 See the 'OWL 2 RL' sub-tab for more information on this reasoner.

Figures 3.31. Les règles SWRL

Edit ✖

Name	swrl-4
Comment	patient examination
Status	Ok

`emergency(?e) ^ Patients(?p) ^ Doctors(?d) ^ gets(?p, ?d) ^ examination(negatif) -> goesTo(?p, ?e)`

Cancel
Ok

Figures 3.32. Editeur des Règles SWRL

Exécution de requêtes SQWRL

Deux mécanismes sont fournis par SWRLAPI pour exécuter des requêtes SQWRL: (1) une API Java qui fournit une interface de type JDBC, appelée API de requête SQWRL, qui peut être utilisée pour exécuter des requêtes et récupérer des résultats de requête dans des applications Java, et (2) une interface utilisateur graphique appelée l' onglet Requête SQWRL qui prend en charge les requêtes interactives et l'affichage des résultats.

The screenshot shows the SWRLAPI interface with a table of queries and a 'Run' button. The table has three columns: Name, Query, and Comment. The queries listed are:

Name	Query	Comment
Sqwrl 1	Patients(?p) ^ Name(?p, ?N) ^ Age(?p, ?Ag) ^ address(?p, ?Ad) ^ State(?p, ?St) -> sqwrl:select(?p, ?N, ?Ag, ?Ad, ?St)	Selection of Positive patients
Sqwrl2	Patients(?p) ^ Name(?p, ?N) ^ Age(?p, ?Ag) ^ address(?p, ?Ad) ^ State(?p, "Sever") -> sqwrl:select(?p, ?N, ?Ag, ?Ad)	selection ser case
Sqwrl3	Patients(?p) ^ Name(?p, ?N) ^ Age(?p, ?Ag) ^ swrlb:greaterThan(?Ag, 50) ^ address(?p, ?Ad) ^ State(?p, ?St) -> sqwrl:select(?p, ?N, ?Ag, ?Ad, ?St)	patients oled than 50
equipments 1	Life_Support_Equipment(?LSE) -> sqwrl:select(?LSE)	list of life support equipments
equipments 2	DiagnosticEquipment(?DE) -> sqwrl:select(?DE)	diagnostic equipments
equipments 3	Medical_Labratory_Equipment(?MLE) -> sqwrl:select(?MLE)	MLE
equipments 4	DurableMedicalEquipment(?DME) -> sqwrl:select(?DME)	DME
equipments 5	Treatment_Equipment(?TE) -> sqwrl:select(?TE)	Treatment equipments

Below the table, there are buttons for 'New', 'Edit', 'Clone', and 'Delete'. The 'SQWRL Queries' section is active, showing 'OWL 2 RL' and 'Sqwrl3' tabs. The main content area contains instructions: 'The SWRLAPI supports an OWL profile called OWL 2 RL and uses an OWL 2 RL-based reasoner to perform querying. See the 'OWL 2 RL' subtab for more information on this reasoner. Executing queries in this tab does not modify the ontology. Using Drools for query execution. Select a SQWRL query from the list above and press the 'Run' button. If the selected query generates a result, the result will appear in a new sub tab. The SWRLAPI supports an OWL profile called OWL 2 RL and uses an OWL 2 RL-based reasoner to perform querying. See the 'OWL 2 RL' subtab for more information on this reasoner. Executing queries in this tab does not modify the ontology. Using Drools for query execution. See the Sqwrl3 tab to review results of the SQWRL query. The query took 5516 milliseconds. 2 rows were returned.' A 'Run' button is highlighted at the bottom.

Figure 3.33. Les Requêtes SQWRL

Résultat d'une requête SQWRL qui représente le nombre de patient qui ont plus de 50 ans dans la figure (34).

SQWRL Queries OWL 2 RL Sqwr13						
P	N	Ag	Ad	St		
.Patient2	Laid Larbi	76**xsd:int	"Les castors oran**xsd:NMTOKEN	Sever		
.patient3	Fatma Mestri	80**xsd:int	Es senia	Sever		

Figures 3.34. Résultat d'exécution d'une Requête SQWRL

Conclusion générale et perspectives

Dans ce mémoire, nous avons présenté une application de la technologie du web sémantique dans le système hospitalier. Nous avons développée l'ontologie HDO (Hospital Departments Ontology), une ontologie à base des services hospitaliers qui facilite Le processus de partage de connaissances entre les différents services tels que, le services des urgences, le service de médecine interne,etc.

Ce processus est inspiré principalement de la méthodologie de construction d'ontologies «METHODOLOGY». Il est constitué de cinq phases successives.

Pour la représentation de connaissances du domaine, nous avons utilisé le formalisme ontologique OWL, qui est le plus compétent pour la définition des concepts hospitaliers. Nous avons implémenté notre ontologie utilisant Protégé, qui est l'outil le plus utilisé dans le développement des ontologies.

Pour éditer les règles SWRL conçues, nous avons utilisé l'éditeur SWRL (plugin de Protégé OWL) qui fournit une interface très interactive pour l'édition des règles prenant en charge l'ensemble des fonctionnalités du langage SWRL.

Pour l'interrogation de notre ontologie nous avons utilisé les requêtes SQWRL pour faciliter le partage d'information entre les employés des différents services et la connaissance du domaine.

Nous avons appliqué l'ontologie HDO dans le service Covid-19 Afin d'assurer l'hospitalisation de l'afflux de patients atteints de COVID-19 tout en garantissant la prise en charge équitable des patients atteints d'autres pathologies, pour soulager les services de réanimation, des places en soins intensifs ont été créées.

Les perspectives de ce mémoire concernant l'application d'ontologie HDO seraient de passer de domaines d'application spécifiques à des domaines plus généraux. Notons que Comme nous l'avons mentionné précédemment, l'ontologie HDO a été appliqué dans le service Covid-19 à l'hôpital EHU d'Oran.

Bibliographie

- [1] T. Berners-Lee, J. Hendler, O. Lassila, The Semantic Web, Scientific American Magazine, 284 (5), (pp. 34-43), (2001)
- [2] M. Minsky, A framework for representing knowledge. Dans Haugleland, J. Mind Design, The MIT Press, (pp. 95-128), (1981)
- [3] R. Quillia, Mn, M. Minsky, Semantic memory. Semantic Information proceeding Cambridge , Mass MIT press. (1968)
- [4] M. Shmidt-Schaub, G.Smolka. Attributive concept descriptions with complements. Artificial Intelligence 48 (1), (pp. 1- 26), (1991)
- [5] F. Baade, W. Nutt. Basic description logics. In Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, Peter F. Patel-Schneider, editors, *The Description Logic Handbook: Theory, Implementation, and Applications*., Cambridge University Press, (pp. 43–95), (2003)
- [6] Ora Lassila , Ralph R. Swick, Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation, World Wide Web Consortium, Cambridge (MA), (1999).
- [7] L. Deborah, McGuinness, Frank van Harmelen, OWL Web Ontology Language Overview *W3C Recommendation, World Wide Web Consortium, (2004)*.
- [8] D. L. McGuinness, F. van Harmelen, Owl web ontology language overview. World Wide Web Consortium, Recommendation REC-owlfeatures- 20040210. (2004)
- [9] RDF(S), Rules, and OWL Dialects. <http://www.ontotext.com/rdfs-rules-owl>
- [10] RIF – Rule Interchange Format Working group, (2008). http://www.W3.org/2005/rules/wiki/RIF_Working_Group
- [11] T. Gruber, A translation approach to portable ontology specifications. Knowledge Acquisition, (pp 199-220), (1993).
- [12] N. Guarino, Understanding, building, and using ontologies: A commentary to "Using Explicit Ontologies in KBS Development", by van Heijst, Schreiber, and Wielinga. International Journal of Human-Computer Studies, Vol. 46, (pp. 293-310), (1997)
- [13] F. Furst, L'ingenierie ontologique. Rapport de recherche IRIN N° 02-07, (2002).
- [14] H. Mihoubi, Une approche declarative de traduction d'ontologies. These Grenoble : université Stendhal Grenoble 3, (2000).
- [15] Actes du colloque, Terminologies et intelligence Artificielle._

Bibliographie

- [16] M.Uschold, M.Grüniger; ONTOLOGIES: Principles, Methods and Applications. Knowledge Engineering Review, (1996).
- [17] G. V. Heijst, A. Th. Schreiber, B.J. Wielinga, *Using explicit ontologies in KBS development*. Int. J. Human.-Computer. Studies. 46(2): (pp.183-292), (1997)
- [18] A.Napoli, Une introduction aux logiques de description, Rapport technique-INRIA, (1997)
- [19] P.Fournier-viger, Un modèle de représentation des connaissances à trois niveaux de sémantique pour les systèmes tutoriels intelligents, Thèse de maîtrise, Sherbrooke, Québec, Canada. (2005)
- [20] <http://protege.stanford.edu>
- [21] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosz, M. Dean, SWRL: A Semantic Web Rule Language Combining OWL and RuleML, W3C Member , (2008), from <http://www.w3.org/Submission/SWRL/>.com
- [22] H. Knublauch, M. O'Connor, S. Tu and M. Musen. Writing Rules for the Semantic Web Using SWRL and Jess. 8th International Protege Conference, Protégé with Rules Workshop, Madrid, Spain, (2005)
- [23] 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).
- [24] M.Hemam. Un processus de developpement d'ontologie dans le cadre du web sémantique, Centre Universitaire Larbi Ben M'hidi - Oum El Bouaghi- Institut des sciences exactes. (2005).
- [25] M. Fernandez, A. Gomez-Perez, N. Juristo, METHONTOLOGY : From Ontological Arts towards Ontological Engineering. In proceeding of the AAAI97 Spring symposium Series on Ontological Engineering, Stanford, USA, (pp. 33- 40), (1997).
- [26] M. Uschold, M.King, Towards a Methodology for building Ontologies. In proceeding of the Workshop on Basic Ontological Issues in Knowledge Sharing at the International Joint Conference on Artificial Intelligence (IJCAI'1995), Montreal, Canada,(1995)
- [27] M. Fox, M. Gruninger, Enterprise Modeling. In AI Magazine, Fall 1998, vol 19, n° 3, (pp. 109- 121), (1993)

Bibliographie

- [28] B. Bachimont, Modélisation linguistique et modelisation logique des ontologies : l'apport de l'ontologie formelle. In Actes des journées francophones d'ingénierie des connaissances (IC'2001) , Grenoble : presse universitaire Grenoble, (2001)
- [29] R.W. Driggers, C.M. Zahn., Obstetrics and Gynecology Clinics ; vol 35, issue 4, (pp. 583- 597). (2008)
- [30] Petit Larousse de la medecine – ISBN 2-03-560349-8
- [31] F. Natalya, La création de votre première ontologie. Université de Stanford, CA.
- [32] T. Eiter, G. Ianni, A. Polleres, R. Schindlauer and H. Tompits. Reasoning with rules and ontologies. In Reasoning Web 2006, volume 4126 of Lecture Notes in Computer Science, pages 93–127. Springer, September 2006.
- [33] G. Antoniou, C. V. Damasio, B. Grosf, I. Horrocks, M. Kifer, J. Maluszynski and P. F.Patel-Schneider. Combining Rules and Ontologies: A survey. Technical Report IST506779/Linkoping/I3-D3/D/PU/a1, Linkoping University, February 2005. IST2004- 506779 REVERSE Deliverable I3-D3. <http://reverse.net/publications/>.
- [34] I. Horrocks, PF. Patel-Schneider, H. Boley, S. Tabet and B. Grosf SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C, May 21, 2004.
- [35] H. Knublauch, M. O'Connor, S. Tu and M. Musen, Writing Rules for the SemanticWeb Using SWRL and Jess. 8th International Protégé Conference,Protege with Rules Workshop, Madrid, Spain, 2005.
- [36] Thygesen, K., Jaffe, A. S., Chaitman, B. R., White, H. D., Zealand, N., & Canada, P. J. D. (2018). Fourth Universal Definition of Myocardial Infarction (2018). Journal of the American College of Cardiology, 72(18), 2231–2264.
- [37] Lamine Benmimoune, Amir Hajjam, Parisa Ghodous, Emmanuel Andres, Samy Talha, et al.. Ontology-based Medical Decision Support System to Enhance Chronic Patients' Lifestyle within Ecare Telemonitoring Platform. International Conference on Informatics, Management and Technology in Healthcare ICIMTH, Jul 2015, Athènes, Greece. pp. 279-282. hal-01263682.
- [38] Rosier, Arnaud & Mabo, Philippe & Temal, Lynda & van Hille, Pascal & Dameron, Olivier & Deléger, Louise & Grouin, Cyril & Zweigenbaum, Pierre & Jacques... (2016). Remote Monitoring of Cardiac Implantable Devices: Ontology Driven Classification of the Alerts. Studies in health technology andinformatics. 221.