

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université Abbes Laghrour Khenchela  
Faculté des sciences et de la Technologie  
Département de Mathématiques et Informatique

N° de série : .....



Mémoire pour obtenir le diplôme de  
Master en Informatique

Spécialité : Informatique

Option : Sécurité et Technologies Web

## Enrichissement sémantique des données de capteurs Application au domaine météorologique

Présenter par :

Fatima Zahra AMARA et Khaoula FALEK

Dirigé par :

Dr. Mounir HEMAM

Devant le jury composé de :

Dr. Nawel TAKOUACHET	M.C. à l'Université Abbes Laghrour Khenchela	Présidente
Dr. Mounir HEMAM	M.C. à l'Université Abbes Laghrour Khenchela	Directeur
Mme. Asma BEZZA	M.A. à l'Université Abbes Laghrour Khenchela	Examinatrice

Soutenu le : 25 juin 2018

*À l'homme de ma vie  
mon exemple éternel, la source de mes efforts; mon père  
À la lumière de mes jours  
la flamme de mon cœur, ma vie et mon bonheur; ma mère  
avec tous mes sentiments  
de respect, d'amour, de gratitude et de reconnaissance  
pour tous les sacrifices déployés pour m'élever dignement  
et assurer mon éducation dans les meilleures conditions*

*À mon frère Amor  
À mes sœurs Yassamine et Taous  
À toute ma famille...*

*À mon binôme Khaoula  
Je dédie ce travail*

*Fatima Zahra*

*Je dédie ce travail en premier lieu  
À mes très chers parents à qui j'ai transmis  
mon stress et anxiétés, pour leur affection,  
leur patience, leur soutien et leurs encouragements  
qui m'ont permis d'arriver au bout de ce travail  
À mon père a qui je souhaite un vif rétablissement , À ma mère  
Et bien sûr a toutes les deux familles FALEK et CHERRABEN  
À mes chères tantes et cousines  
À mon Binôme Fatima Zahra  
À tout mes collègues  
À toutes mes amies*

*Khaoula*

# REMERCIEMENT

” La gratitude est non seulement la plus grande des vertus, mais c'est également la mère de tous les autres. “

Emil Cioran

*Avant tout, nous remercions Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail.*

*Nous tenons tout d'abord adresser toute notre gratitude et remerciement à **Dr. Mounir HEMAM** qui nous a permis de bénéficier de son encadrement. Les conseils qu'il nous a prodigué, la patience, la confiance qu'il nous a témoignés ont été déterminants dans la réalisation de notre travail.*

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

*Nos remerciements s'étendent également à tous les professeurs qui nous ont enseigné et qui par leurs compétences nous ont soutenu dans la poursuite de nos études.*

# RESUME

Les technologies de capteurs et les réseaux de capteurs sans fil (WSN) sont conçus pour collecter de grands volumes de données hétérogènes afin de surveiller les phénomènes environnementaux. Le partage, la réutilisabilité et la fusion de données de capteurs provenant de réseaux de capteurs hétérogènes ne sont pas possibles sans une compréhension de la sémantique des données. Dans notre travail, nous nous intéressons à l'interopérabilité des données de capteurs pour créer des applications Web prometteuses et interopérables pour des domaines spécifiques ou inter-domaines. Pour faire face à ce défi, nous proposons une approche basée sur les technologies du Web sémantique qui offre une interopérabilité sémantique pour les données de capteurs parmi les WSN hétérogènes. Pour ce faire, nous proposons d'abord un vocabulaire de données étendu à partir de l'ontologie SSN pour faciliter l'annotation manuelle sur les capteurs, les observations et les mesures, en fournissant des métadonnées sémantiques sur les propriétés spatiales, temporelles et thématiques. Nous interrogeons ensuite l'ontologie avec le langage de requêtes SPARQL. Enfin, nous appliquons un mécanisme de raisonnement sur des données de capteurs annotées sémantiquement pour en déduire des connaissances nouvelles ou implicites, découvrir des données significatives et répondre à des requêtes complexes. Pour illustrer notre approche, nous l'appliquons sur un exemple concernant le domaine météorologique.

**Mots-clés :** Web Sémantique, Web de Capteurs, Ontologie, Annotation Sémantique, Web de Capteurs Sémantique.

# Abstract

Sensor technologies and Wireless Sensor Networks (WSN) are designed to collect large volumes of heterogeneous data for monitoring environmental phenomenon. Sharing, reusability and fusion of sensor data from heterogeneous sensor networks is not possible without understanding the semantics of the data. In our work we focus on the interoperability of sensor data to build promising and interoperable domain-specific or cross-domain sensor Web applications. To deal with this challenge, we propose an approach based on semantic Web technologies which offers semantic interoperability for sensor data among heterogeneous WSN. For this purpose, we first propose a data vocabulary extended from SSN ontology to facilitate manual annotation on sensors, observations and measurements, by providing semantic metadata about spacial, temporal and thematic properties. We then query the ontology with the SPARQL query language. Finally, we apply a reasoning mechanism on semantically annotated sensor data to deduce new or implicit knowledge, discover significant data and answer complex queries. To illustrate our approach, we apply it on an example concerning the meteorological domain.

**Keywords:** Semantic Web, Sensor Web, Ontology, Semantic Annotation, Semantic Sensor Web.

## ملخص

تم تصميم تقنيات الاستشعار وشبكات الاستشعار اللاسلكية (WSN) لجمع كميات كبيرة من البيانات غير المتجانسة لرصد الظواهر البيئية. لا يمكن مشاركة وإعادة استخدام وإدماج بيانات المستشعر الملتقطة من شبكات الاستشعار غير المتجانسة بدون فهم دلالات البيانات. في عملنا، نحن مهتمون بالتشغيل البيئي لبيانات المستشعر لإنشاء تطبيقات ويب واعدة وقابلة للتشغيل المتبادل لمجالات محددة أو مشتركة بين النطاقات. للتعامل مع هذا التحدي، نقترح نهجاً يعتمد على تقنيات الويب الدلالي التي توفر إمكانية التشغيل البيئي الدلالي لبيانات المستشعر بين شبكات الاستشعار اللاسلكية الغير متجانسة. ولهذا الغرض، نقترح أولاً استخدام معجم للبيانات ممددة من نظام الأنطولوجيا SSN لتسهيل التعليق التوضيحي على أجهزة الاستشعار والملاحظات والقياسات، من خلال توفير بيانات وصفية دلالية عن الخصائص المكانية والزمنية والموضوعية. ثم نسأل الأنطولوجيا باستعمال لغة الاستفسار SPARQL وأخيراً، نطبق آلية تفكير حول بيانات المستشعرات ذات الدلالة الدلالية لاستخلاص معلومات جديدة أو ضمنية واكتشاف بيانات مهمة والإجابة على الاستفسارات المعقدة. لتوضيح نهجنا، نقوم بتطبيقه على مثال يتعلق بمجال الأرصاد الجوية.

**الكلمات المفتاحية:** الويب الدلالي، ويب أجهزة الاستشعار، الأنطولوجيا، الشرح الدلالي، ويب أجهزة الاستشعار الدلالي.

# TABLE DES MATIERES

<b>LISTE DES FIGURES</b> .....	<b>XI</b>
<b>LISTE DES TABLEAUX</b> .....	<b>XIII</b>
<b>ANNEXE</b> .....	<b>XIV</b>
<b>INTRODUCTION GENERALE</b> .....	<b>1</b>
1. CONTEXTE DE TRAVAIL ET PROBLEMATIQUE.....	1
2. NOTRE CONTRIBUTION .....	2
3. ORGANISATION DU MEMOIRE .....	2
<b>CHAPITRE 1</b> .....	<b>4</b>
<b>TECHNOLOGIES DU WEB SEMANTIQUE</b> .....	<b>4</b>
1. WEB SEMANTIQUE ET SES DEFERENTS COMPOSANTS .....	5
1.1. <i>Web sémantique</i> .....	5
1.2. <i>Composants du web sémantique</i> .....	5
2. STANDARDS ET ARCHITECTURE DU WEB SEMANTIQUE .....	6
3. META-DONNEES ET ANNOTATIONS SUR LE WEB SEMANTIQUE .....	7
4. LANGAGES DU WEB SEMANTIQUE .....	7
4.1. <i>XML (eXtensible Markup Language)</i> .....	7
4.2. <i>RDF (Ressource Discription Framework)</i> .....	7
4.3. <i>RDFs (Resource Description Framework Schema)</i> .....	9
4.4. <i>OWL (Web Ontology Language)</i> .....	9
4.5. <i>SPARQL (SPARQL Protocol and RDF Query Language)</i> .....	10
4.6. <i>SWRL (Semantic Web Rule Language)</i> .....	10
5. ONTOLOGIE ET WEB SEMANTIQUE.....	11
5.1. <i>Définition de la notion ontologie</i> .....	11
5.2. <i>Composants d'une ontologie</i> .....	11
5.3. <i>Classification des ontologies</i> .....	11
5.3.1. Dimension objet de conceptualisation.....	12
5.3.2. Dimension détail de l'ontologie.....	13
5.3.3. Dimension niveau de formalisme.....	13
5.4. <i>Formalismes de représentation</i> .....	14
5.4.1. Langage de frame (frame-based languages).....	14
5.4.2. Graphe conceptuel.....	15
5.4.3. Logique de description .....	15
5.5. <i>Outils de construction d'ontologies</i> .....	15
5.5.1. Outils dépendants du formalisme de représentation .....	15
5.5.2. Outils indépendants de formalisme de représentation.....	16
6. MOTEURS D'INFERENCE .....	17

7. CONCLUSION .....	17
<b>CHAPITRE 2.....</b>	<b>18</b>
<b>RESEAUX DE CAPTEURS SANS FILS ET WEB DE CAPTEURS .....</b>	<b>18</b>
1. CAPTEURS, SYSTEMES DE CAPTEURS ET RESSOURCES DE CAPTEURS .....	19
1.1. Définition d'un capteur .....	19
1.2. Anatomie d'un nœud de capteur .....	19
1.3. Limites des ressources de capteurs.....	20
2. RESEAUX DE CAPTEURS, UNE VALEUR AJOUTEE A LA COMMUNICATION .....	20
2.1. Définition des RCSF .....	20
2.2. Les limites des réseaux de capteurs.....	21
3. WEB DES CAPTEURS .....	21
3.1. Web de Capteurs, une ouverture des réseaux de capteurs.....	21
3.2. Couches inter-réseaux du Web de capteurs .....	22
3.3. Norme SWE (Sensor Web Enablement) du groupe OGC .....	23
4. WEB DES CAPTEURS SEMANTIQUE .....	24
4.1. Métadonnées des capteurs dans l'espace, le temps et le thème.....	24
4.2. Ontologies dans le domaine du Web de capteurs sémantique .....	25
4.3. Architecture Web sémantique pour les réseaux de capteurs (SWANS).....	26
4.3.1. Sources de données du réseau de capteurs .....	26
4.3.2. Couche d'ontologie .....	26
4.3.3. Couche de traitement du Web sémantique .....	27
4.3.4. Couche d'application.....	27
4.4. Traitement de données de capteur.....	27
5. QUELQUES DOMAINES D'APPLICATIONS DES RCSF .....	27
5.1. Détection des incendies de forêts .....	28
5.2. Agriculture .....	28
5.3. Contrôle de la pollution .....	28
5.4. Applications militaires .....	29
5.5. Surveillance médicale .....	29
6. CONCLUSION .....	29
<b>CHAPITRE 3.....</b>	<b>30</b>
<b>CONCEPTION DE L'ONTOLOGIE.....</b>	<b>30</b>
1. ONTOLOGIES DES RESEAUX DE CAPTEURS .....	30
2. CYCLE DE VIE D'UNE ONTOLOGIE .....	30
3. METHODOLOGIE DE CONSTRUCTION D'UNE ONTOLOGIE « METHONTOLOGIE ».....	31
4. PROCESSUS DE CONSTRUCTION DE L'ONTOLOGIE .....	31
4.1. Spécification .....	31
4.2. Conceptualisation .....	32
4.3. Formalisation .....	32
5. CONCEPTION DE NOTRE ONTOLOGIE .....	33
5.1. Spécification .....	33

5.2.	<i>Conceptualisation</i> .....	34
a.	Construction de glossaire de termes.....	35
b.	Construction de diagramme de classification de concepts .....	38
c.	Construction de diagramme de relations binaires .....	39
d.	Dictionnaire de concepts .....	40
e.	Tableaux des relations binaires .....	41
f.	Tableaux des attributs.....	41
g.	Tableaux des axiomes logiques.....	42
h.	Tableaux des instances .....	43
i.	Construction de la table des assertions.....	43
5.3.	<i>Formalisation</i> .....	44
5.3.1.	Construction de TBOX.....	44
a.	Définition des concepts .....	44
b.	Définition des rôles .....	45
5.3.2.	Construction de ABOX .....	45
a.	Partie assertionelle des concepts .....	45
b.	Partie assertionelle des rôles .....	46
6.	CONCLUSION .....	46
<b>CHAPITRE 4.....</b>		<b>47</b>
<b>IMPLEMENTATION ET ANNOTATION SEMANTIQUE .....</b>		<b>47</b>
1.	PRESENTATION DE L'OUTIL PROTEGE 5.2.....	47
2.	ETAPES DE CREATION DE NOTRE ONTOLOGIE.....	48
2.1.	<i>Définition des classes</i> .....	48
2.2.	<i>Définition des propriétés</i> .....	49
2.2.1.	Object property .....	49
2.2.2.	Data type property.....	50
2.3.	<i>Ajout des individus</i> .....	51
3.	ANNOTATION SEMANTIQUE DES CAPTEURS ET DES OBSERVATIONS AVEC RDF .....	51
4.	INTERROGATION DE L'ONTOLOGIE AVEC SPARQL .....	55
4.1.	<i>Ajout de SPARQL Query</i> .....	55
4.2.	<i>Exécution de requêtes</i> .....	55
5.	DEFINITION DES REGLES AVEC SWRL TAB.....	57
5.1.	<i>Le moteur de règles DROOLS</i> .....	57
5.2.	<i>Edition des règles</i> .....	58
5.3.	<i>Exécution des règles</i> .....	59
6.	CONCLUSION .....	60
<b>CONCLUSION ET PERSPECTIVES.....</b>		<b>61</b>
.....		<b>61</b>
<b>BIBLIOGRAPHIE .....</b>		<b>62</b>

# LISTE DES FIGURES

<b>FIGURE 1.1 : VISION DU WEB SEMANTIQUE.</b> .....	<b>5</b>
<b>FIGURE 1.2 : LES COUCHES D'ARCHITECTURE DU WEB SEMANTIQUE.</b> .....	<b>6</b>
<b>FIGURE 1.3: TRIPLET RDF.</b> .....	<b>8</b>
<b>FIGURE 1.4: EXEMPLE RDF (ANNOTATION XML ET GRAPHIQUE).</b> .....	<b>8</b>
<b>FIGURE 1.5: LES COUCHES DE L'OWL.</b> .....	<b>10</b>
<b>FIGURE 1.6: TYPOLOGIE ET CLASSIFICATION DES ONTOLOGIES.</b> .....	<b>14</b>
<b>FIGURE 2.1 : ANATOMIE D'UN NŒUD DE CAPTEUR.</b> .....	<b>19</b>
<b>FIGURE 2.2 : SYSTEME DE CAPTEURS ET CAPTEURS EN TANT QUE RESSOURCES DE CAPTEUR.</b> .....	<b>19</b>
<b>FIGURE 2.3 : ARCHITECTURE D'UN RCSF.</b> .....	<b>20</b>
<b>FIGURE 2.4 : L'AJOUT DE L'ASPECT COMMUNICATIF AUX RESSOURCES DE CAPTEURS CREE LES RESEAUX DE CAPTEURS.</b> .....	<b>21</b>
<b>FIGURE 2.5 : LE WEB DES CAPTEURS RASSEMBLE TOUS LES ELEMENTS DE MANIERE OUVERTE.</b> .....	<b>22</b>
<b>FIGURE 2.6 : LA PILE DE COUCHES INTER RESEAU DE CAPTEURS DU WEB DES CAPTEURS.</b> .....	<b>23</b>
<b>FIGURE 2.7 : SOUS-ENSEMBLE DE CONCEPTS ET RELATIONS REPRESENTES AVEC UNE SUITE D'ONTOLOGIES.</b> .....	<b>25</b>
<b>FIGURE 2.8 : UNE ARCHITECTURE DU WEB DES CAPTEURS SEMANTIQUE.</b> .....	<b>26</b>
<b>FIGURE 2.9 : APPLICATION DES RCSF.</b> .....	<b>28</b>
<b>FIGURE 3.1 : LE CYCLE DE VIE D'UNE ONTOLOGIE.</b> .....	<b>31</b>
<b>FIGURE 3.2: TERMES GENERALE DE NOTRE MODELE ONTOLOGIQUE.</b> .....	<b>33</b>
<b>FIGURE 3.3 : UN DOCUMENT RDF DE SPECIFICATION DE L'ONTOLOGIE.</b> .....	<b>34</b>
<b>FIGURE 3.4 : DIAGRAMME DE CLASSIFICATION DE CONCEPTS.</b> .....	<b>38</b>
<b>FIGURE 3.5 : DIAGRAMME DE RELATION BINAIRE.</b> .....	<b>39</b>
<b>FIGURE 4.1 : L'INTERFACE DE PROTEGE 5.2.</b> .....	<b>48</b>
<b>FIGURE 4.2 : LA HIERARCHIE DE CLASSES.</b> .....	<b>49</b>
<b>FIGURE 4.3 : L'ONGLET OBJECT PROPERTIES.</b> .....	<b>50</b>
<b>FIGURE 4.4 : L'ONGLET DATA PROPERTIES.</b> .....	<b>50</b>

<b>FIGURE 4.5: L'ONGLET INDIVIDUALS BY CLASS.....</b>	<b>51</b>
<b>FIGURE 4.6: GRAPHE RDF ET DESCRIPTION SEMANTIQUE DE « THERMOMETER_1 ».....</b>	<b>52</b>
<b>FIGURE 4.7 : GRAPHE RDF ET DESCRIPTION SEMANTIQUE DE « ANEMOMETER_1 ».....</b>	<b>52</b>
<b>FIGURE 4.8: GRAPHE RDF ET DESCRIPTION SEMANTIQUE DE « BAROMETER_1 ».....</b>	<b>53</b>
<b>FIGURE 4.9 : GRAPHE RDF ET DESCRIPTION SEMANTIQUE DE « OBS_1 ».....</b>	<b>53</b>
<b>FIGURE 4.10: GRAPHE RDF ET DESCRIPTION SEMANTIQUE DE « OBS_8 ».....</b>	<b>54</b>
<b>FIGURE 4.11: GRAPHE RDF ET DESCRIPTION SEMANTIQUE DE « OBS_10 ».....</b>	<b>54</b>
<b>FIGURE 4.12 : L'ONGLET SPARQL QUERY.....</b>	<b>55</b>
<b>FIGURE 4.13 : RESULTAT D'EXECUTION DE LA REQUETE 1.....</b>	<b>56</b>
<b>FIGURE 4.14 : RESULTAT D'EXECUTION DE LA REQUETE 2.....</b>	<b>56</b>
<b>FIGURE 4.15 : RESULTAT D'EXECUTION DE LA REQUETE DESCRIBE.....</b>	<b>57</b>
<b>FIGURE 4.16 : L'ONGLET SWRL TAB.....</b>	<b>58</b>
<b>FIGURE 4.17 : L'EDITEUR DE REGLES.....</b>	<b>58</b>
<b>FIGURE 4.18 : L'EXECUTION DES REGLES.....</b>	<b>59</b>
<b>FIGURE 4.19 : LES OBSERVATIONS AVANT L'EXECUTION DES REGLES.....</b>	<b>60</b>
<b>FIGURE 4.20 : LES OBSERVATIONS APRES L'EXECUTION DES REGLES.....</b>	<b>60</b>

# LISTE DES TABLEAUX

<b>TABLE 3.1 : GLOSSAIRE DE TERMES. ....</b>	<b>37</b>
<b>TABLE 3.2 : LE DICTIONNAIRE DE CONCEPTS. ....</b>	<b>40</b>
<b>TABLE 3.3 : TABLE DES RELATIONS BINAIRES. ....</b>	<b>41</b>
<b>TABLE 3.4 : LA TABLE DES ATTRIBUTS. ....</b>	<b>41</b>
<b>TABLE 3.5: LA TABLE DES AXIOMES LOGIQUE.....</b>	<b>42</b>
<b>TABLE 3.6 : LA TABLE DES INSTANCES. ....</b>	<b>43</b>
<b>TABLE 3.7: LA TABLE DES ASSERTIONS. ....</b>	<b>43</b>
<b>TABLE 3.8 : DEFINITION DES CONCEPT (DANS TBOX). ....</b>	<b>45</b>
<b>TABLE 3.9 : DEFINITION DES ROLES (DANS TBOX). ....</b>	<b>45</b>
<b>TABLE 3.10 : DESCRIPTION ASSERTIONELLE DES CONCEPTS. ....</b>	<b>45</b>
<b>TABLE 3.11 : DESCRIPTION ASSERTIONELLE DES ROLES. ....</b>	<b>46</b>

# Annexe

<b>OGC</b>	Open Geospatial Consortium
<b>OWL</b>	Web Ontology Language
<b>RCSF</b>	Réseau de Capteur Sans Fil
<b>RDF</b>	Ressource Discription Framework
<b>RDFs</b>	Resource Description Framework Schema
<b>SPARQL</b>	SPARQL Protocol and RDF Query Language
<b>SSN</b>	Semantic Sensor Network
<b>SSW</b>	Semantic Sensor Web
<b>SWANS</b>	Semantic Web Architecture Network Sensor
<b>SWE</b>	Sensor Web Enablement
<b>SWRL</b>	Semantic Web Rule Language
<b>URI</b>	Uniform Resource Identifier
<b>W3C</b>	World Wide Web Consortium
<b>WSN</b>	Wireless Sensor Network
<b>XML</b>	eXtensible Markup Language

# INTRODUCTION GENERALE

| ” *Le commencement est la moitié de tout* “  
Pythagore

## 1. Contexte de travail et problématique

Au cours des dernières années, les réseaux de capteurs sans fil ont attiré beaucoup d'attention des chercheurs en raison de leurs larges éventails d'applications potentielles. Les réseaux de capteurs ont introduit une nouvelle classe de systèmes informatiques et élargissent la capacité des individus à interagir à distance avec le monde physique. Les réseaux de capteurs sont utilisés pour de nombreuses applications incluant les sciences médicales pour le soin des patients à l'aide de capteurs biométriques, la détection des incendies, la météorologie pour la prévision météorologique, l'imagerie satellitaire pour l'observation terrestre et spatiale, les terres agricoles, etc.

La combinaison de réseaux de capteurs avec le Web, les services Web et les technologies de base de données a été baptisée il y a quelques années sous le nom de Web de Capteur [1]. Il est défini comme un réseau de capteurs hétérogènes interconnectés et un type particulier d'infrastructure d'information sur le Web pour la collecte, la modélisation, le stockage, la récupération, le partage, la manipulation, l'analyse et la visualisation d'informations sur les capteurs et leurs observations.

Les capteurs sont répartis à travers le globe, capturant et produisant en continu une quantité énorme de flux de données sur un certain nombre de phénomènes du monde réel [2]. Tous ces différents capteurs disponibles fournissent des données brutes hétérogènes, fournies à différents formats et sans sémantique pour décrire sa signification. L'absence de sémantique commune intensifie le problème existant de "trop de données et pas assez de connaissances" [3]. En vue de résoudre ce problème le Web des capteurs sémantique est apparue qui est le couplage du Web sémantique avec le Web de capteurs afin d'améliorer la sémantique des données de capteur et d'augmenter l'interopérabilité entre des réseaux de capteurs hétérogènes, ainsi que pour fournir des informations contextuelles essentielles pour la connaissance de la situation. Dans ce contexte, les techniques du Web sémantique peuvent construire un modèle conceptuel partagé, améliorer la sémantique des données de capteur, et réaliser l'interaction et l'accès des données de capteurs sur le web.

## **2. Notre contribution**

Notre contribution dans ce mémoire consiste l'enrichissement sémantiquement des capteurs et des flux de capteurs dans le domaine de la météorologie. Pour faire cela nous définissons une ontologie étendue à partir de l'ontologie W3C SSN. Le modèle ontologique développé est composé de différents concepts qui traitent les caractéristiques des capteurs et les données des capteurs liées aux mesures et observations météorologique en suite sur le modèle ontologique développé, nous proposons d'annoter et d'enrichir les descriptions de capteurs et les flux de capteurs avec des métadonnées sémantiques pour soutenir l'interopérabilité et fournir des informations contextuelles essentielles pour la connaissance de la situation. En particulier, cela implique d'annoter les données de capteurs avec des métadonnées sémantiques spatiales, temporelles et thématiques. Enfin, nous interrogeons notre ontologie avec le langage SPARQL et nous appliquons un raisonnement basé sur des règles sémantiques en utilisant le langage de règles du Web sémantique SWRL.

## **3. Organisation du mémoire**

Le manuscrit est structuré en quatre chapitres et une conclusion générale.

### **Chapitre 1 :**

Dans ce chapitre nous allons présenter quelques notions de base, que nous avons jugé utile dans le contexte de notre projet qui a pour but l'annotation sémantique des réseaux de capteurs utilisés pour la surveillance environnementale dans le domaine météorologique. Ces notions sont afférentes aux technologies du Web sémantiques, en mettant l'accent sur ses différents composants tels que les ontologies, les métadonnées et les annotations sémantiques.

### **Chapitre 2 :**

Ce chapitre est dédié à la présentation des réseaux de capteurs, leur évolution, d'un système classique basé sur un modèle stimulus-réponse à un système intelligent, capable de percevoir son environnement, raisonner et prendre des décisions dans une perspective d'émuler un comportement humain face à une situation quelconque et grâce à l'intégration des technologies des Web sémantiques dans les différents composants constituant ce type de réseau.

### **Chapitre 3 :**

Ce chapitre est consacré à la conception de notre ontologie de réseau de capteurs pour la météorologie basée sur l'ontologie SSN. Le modèle ontologique développé est composé de différents concepts qui traitent, d'une part, des caractéristiques des capteurs et

d'autre part, des données des capteurs liées aux mesures et observations du domaine de la météorologique. En utilisant un processus de construction dans le développement de l'ontologie partant de connaissances brutes et arrivant à une ontologie d'application opérationnelle représentée par le langage OWL.

#### **Chapitre 4 :**

Dans ce chapitre nous allons présenter en détail l'implémentation de notre ontologie ainsi que l'annotation sémantique des capteurs et leurs observations ensuite l'interrogation de l'ontologie avec SPARQL et enfin, nous appliquons un raisonnement basé sur des règles sémantiques pour déduire de nouvelle connaissance en se basant sur le langage de règles du Web sémantique SWRL.

Le mémoire s'achève avec une conclusion générale récapitulant le contexte de recherche de notre étude, la démarche suivie, nos contributions et énonce un ensemble de perspectives.

# CHAPITRE 1

## TECHNOLOGIES DU WEB SEMANTIQUE

*” [...] Le Web a été conçu comme un espace informationnel, avec pour but d’être utile non seulement à la communication entre humains, mais aussi pour que des machines puissent y participer et aider “*

Tim Berners-Lee

Si on est capable aujourd’hui d’utiliser le web c’est parce que l’être humain est extrêmement doué et flexible dans le traitement de données, le web actuel ou le web des documents est syntaxique est un ensemble de documents et pages liée par des liens hypertextes. On peut faire des recherches en effectuant des simples requêtes et c’est à l’utilisateur qu’il revient de les manipuler, les combiner et les interpréter pour obtenir l’information pertinente. Le Web des documents reste donc réservé à l’humain, la machine ne peut pas le remplacer efficacement et c’est la qui a apparu le web sémantique. Il s’agit donc d’arriver à un Web intelligent, où les informations ne seraient pas seulement stockées mais comprises par les machines pour apporter des réponses pertinentes à l’utilisateur.

Le Web sémantique est venu avec de nouvelles pratiques concernant l’organisation des contenus Web, et une nouvelle infrastructure permettant aux agents logiciels d’aider efficacement les internautes dans leurs accès aux sources d’information et aux services [4]. Grâce à ces différentes technologies et langages tels que XML, RDF, et OWL.

Dans ce présent chapitre nous allons voir des généralités sur le web sémantique ainsi ces défèrent langages qui ont conduit à la définition d’un standard permettant de représenter des ontologies sur le Web, tels que RDF qui offrent un format uniforme pour la description et l’échange du contenu du Web, OWL pour la création des ontologies servant de support aux traitements logiques (inférences, classification automatique), et SPARQL pour obtenir des informations à partir de graphes RDF.

## 1. Web sémantique et ses différents composants

### 1.1. Web sémantique

En Mai 2001, Tim Berners-Lee, James Hendler et Ora Lassila présentèrent le Web sémantique comme étant [5] « Le Web sémantique est une extension du web actuel dans lequel l'information a une signification bien définie, ce qui permet aux ordinateurs et aux gens de travailler en coopération ». Le but étant de disposer de nouveaux langages, tels que le XML, RDF(S) et OWL et de nouveaux outils pour permettre de comprendre et manipuler les contenus sémantiques des documents afin de réaliser un meilleur partage et mieux fixer leurs interprétations [6].

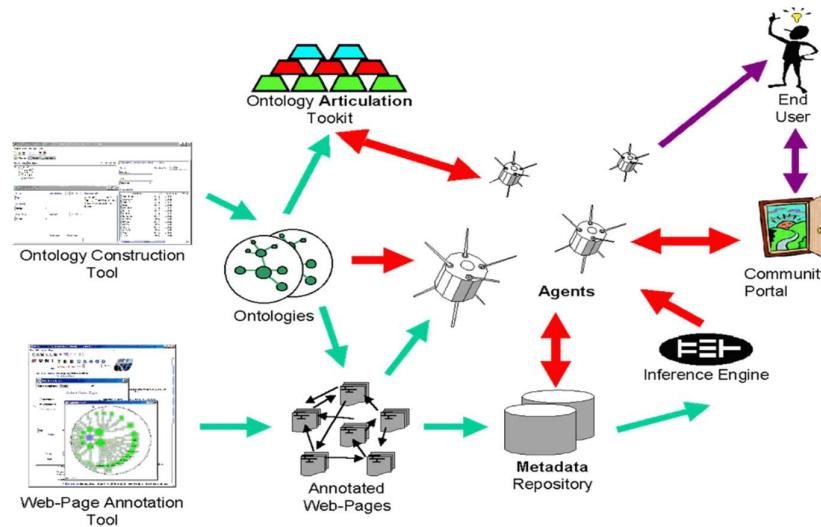


Figure 1.1 : Vision du web sémantique.

### 1.2. Composants du web sémantique

Pour réaliser cette nouvelle approche du Web sémantique, plusieurs composants ont été décelés à travers plusieurs travaux de recherche parmi lesquelles qui ont connu une large adoption par la communauté internationale et ont pris la route vers la concrétisation on peut citer [7] :

- Des langages et des formalismes de représentation et de structuration des connaissances qui permettent de modéliser et de représenter le contenu sémantique des ressources du web.
- Des ontologies pour la modélisation des connaissances nécessaires à la description et au traitement d'un ensemble de ressources, aussi pour structurer et définir la signification des termes de métadonnées actuellement collectés et normalisés.
- Des métadonnées qui doivent compléter l'information relative aux données à un niveau d'abstraction supérieur tout en étant capables de se décrire d'elles-mêmes.

- Des moteurs de raisonnement encapsulés dans des systèmes de requêtes et permettant d'inférer sur les annotations d'après les axiomes déclarés dans les ontologies, afin d'interroger le Web et agir sur les réponses obtenues.

## 2. Standards et architecture du Web sémantique

Les travaux visant la réalisation du Web sémantique se situent à des niveaux de complexité différents. La vision courante du Web sémantique proposée par son inventeur Berners-Lee peut être représentée en plusieurs couches (Figure 1.2).

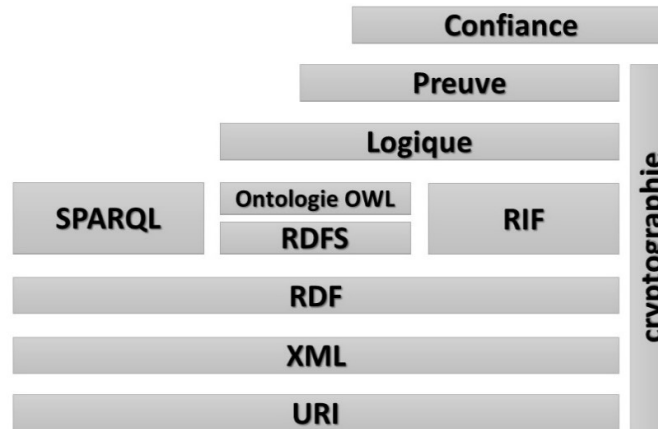


Figure 1.2 : Les couches d'architecture du Web sémantique.

Les couches les plus basses assurent l'interopérabilité syntaxique L'URI (Uniform Resource Identifier) fournit un adressage standard universel permettant d'identifier les ressources en suite XML (eXtensible markup Language) fournit une syntaxe pour décrire la structure du document, créer et manipuler des instances des documents.

Les couches RDF (Resource Description Framework) et RDF-Schema considérées comme les premières fondations de l'interopérabilité sémantique et dans la couche suivante on trouve OWL (Ontology Web Language) permet d'étendre le vocabulaire et les propriétés de RDF Il offre une grande souplesse dans la définition des relations. Dans le même niveau on trouve la couche du langage SPARQL ce langage est désigné par le W3C comme le standard pour l'interrogation des graphes RDF et OWL.

La couche de règles a pour objectif de normaliser la représentation des règles RDF elle comporte deux langages de règles SWRL (Semantic Web Rule Language) et RIF (Rule Interchange Format) la couche logique repose sur les langages ontologiques dans l'architecture recommandée par le W3C elle est généralement utilisée pour exprimer les règles d'inférences et en fin les couches preuve et confiance fournissent des éléments pour réaliser la vérification des déclarations effectuées dans le web sémantique.

### 3. Méta-données et annotations sur le Web sémantique

Un des grands principes du Web sémantique est qu'il est nécessaire d'associer aux ressources du Web des informations exploitables par des agents logiciels afin de favoriser l'exploitation de ces ressources [8] [9]. L'annotation est un processus qui vise à attacher des données complémentaires (méta-données) à d'autres données.

Une méta-donnée est une donnée sur une donnée qui présente des aspects spécifiques pour faciliter la recherche d'information, décrire le contenu et les relations entre les fichiers d'un site, mieux référencer un site ou une page sur Internet et faciliter l'interopérabilité.

### 4. Langages du web sémantique

Selon le consortium W3C, le web sémantique est une nouvelle vision fondée sur l'idée de devoir définir et organiser les informations sur le web de manière qu'elles soient utilisables par les machines pour l'intégration, le partage et leur réutilisation. Dans ce qui suit, nous allons présenter les langages XML, RDF, RDFs, OWL, SPARQL, SWRL.

#### 4.1. XML (eXtensible Markup Language)

Est une recommandation du W3C depuis 10 Février 1998, il utilise la notion de balisage, pour représenter un document de manière arborescente. C'est un méta langage qui permet de structurer un document en définissant ses propres balises en fonction des besoins et sans tenir compte de la signification de cette structure et des systèmes informatiques qui vont l'exploiter [3].

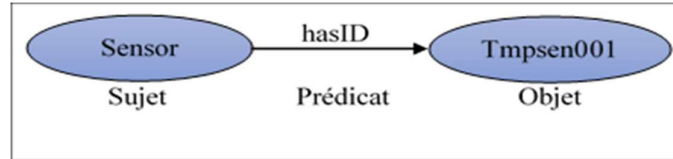
La définition de la syntaxe et de la grammaire d'un nouveau langage basé sur XML se fait soit par une DTD (Document Type Definition), soit par un XSD (XML Schema Definition). Mais XML est limité car il ne dispose pas d'une sémantique c'est pourquoi il est nécessaire de développer d'autres langages.

#### 4.2. RDF (Resource Description Framework)

RDF [10] est un langage d'assertion et d'annotations. Les assertions affirment l'existence de relations entre les objets. Elles sont donc adaptées à l'expression des annotations que l'on veut associer aux ressources du Web. RDF est un langage formel qui permet d'affirmer des relations entre des « ressources ».

RDF est un modèle de graphe destiné à décrire d'une façon formelle les ressources Web et leurs méta-données grâce à des triplets de la forme < sujet-prédicat-objet > où :

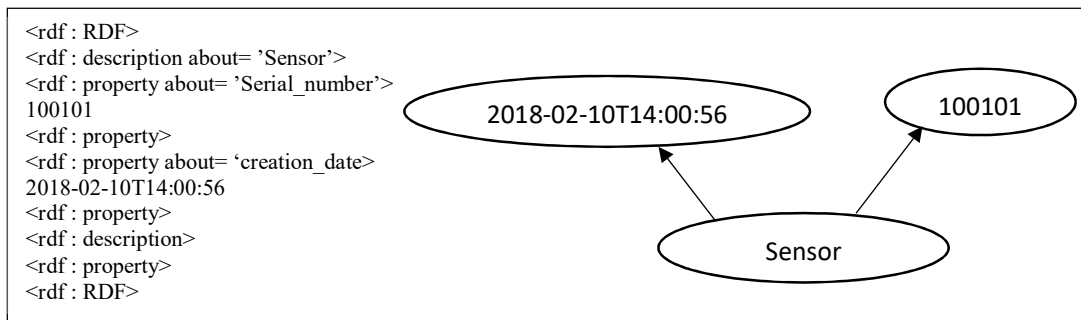
- Le sujet représente la ressource décrite. Il s'agit d'une entité référencée par un identificateur unique (URIs) et il peut être un nœud vide c'est-à-dire il représente une ressource anonyme.
- Le prédicat représente la propriété descriptive peut être un attribut ou une relation qui décrit la ressource.
- L'objet représente la valeur de cette propriété elle peut être une ressource spécifiée par une URI, ou une chaîne de caractères simple (un littéral).



**Figure 1.3:** Triplet RDF.

La syntaxe de RDF est basée sur celle de XML. Le modèle de base de RDF est conçu pour permettre d'associer des attributs aux ressources du Web en utilisant la description de métadonnées sémantiques, c'est ainsi que RDF structure le web comme étant un ensemble de ressources reliées par les liens sémantiques [11].

Un fichier XML/RDF a pour balise racine la balise `<rdf:RDF>` qui contient un ou plusieurs éléments `<rdf:Description>` pour chacune des descriptions de ressources comprises dans le document. Chaque description comprend un attribut `<rdf:about>` qui pointe vers l'URI de la ressource à décrire et un à plusieurs éléments représentant chacun un prédicat. Lorsqu'un prédicat a pour valeur une autre ressource, l'attribut `<rdf:resource>` pointerait vers son URI. [12]



**Figure 1.4:** Exemple RDF (annotation XML et graphique).

RDF propose aussi les conteneurs qui est une ressource qui contient d'autres ressources. On a trois classes de conteneurs :

- *rdf:Bag* définit une liste non ordonnée de ressources ou de littéraux
- *rdf:Seq* qui est une liste ordonnée de ressources ou de littéraux
- *rdf:Alt* désigne un conteneur présentant des alternatives parmi lesquelles une seule doit être sélectionnée.

Si RDF fournit une capacité d'échange de connaissances, il ne permet pas à l'utilisateur de définir le vocabulaire des termes à utiliser, ni d'établir la sémantique des objets utilisés, il est donc nécessaire de donner un sens aux informations stockées sous forme de triplets RDF. C'est le rôle de RDF Schéma, qui permet de créer des vocabulaires de méta-données [13].

### 4.3. RDFs (Resource Description Framework Schema)

Comme son nom l'indique, RDFS a pour but de définir des schémas de méta-données. Il définit le sens, les caractéristiques et les relations d'un ensemble de propriétés. La définition peut inclure des contraintes pour les valeurs potentielles et l'héritage des propriétés d'autres schémas. Il est, en effet, une extension sémantique de RDF afin de fournir un mécanisme pour décrire les groupes associés de ressources et les relations entre les ressources [14].

Les principales caractéristiques de RDFS :

- *rdfs:Class* permet de déclarer une ressource RDF comme une classe pour d'autres ressources.
- *rdfs:subClassOf* permet de définir des hiérarchies de classes.
- *rdfs:domain* définit la classe des sujets liée à une propriété.
- *rdfs:range* définit la classe ou le type de données des valeurs de la propriété.

L'intérêt de RDFS est qu'il facilite l'inférence sur les données et renforce la recherche sur ces données.

### 4.4. OWL (Web Ontology Language)

Le langage d'ontologie Web OWL [15] est conçu pour décrire et représenter un domaine de connaissances spécifiques, en définissant des classes de ressources ou objets et leurs relations mais aussi de définir des individus et affirmer des propriétés les concernant et de raisonner sur ces classes et individus dans la mesure où le permet la sémantique formelle du langage OWL. OWL est un standard basé sur la logique de description [16], il est construit sur RDF et RDFS et utilise la syntaxe RDF/XML.

Le langage OWL offre trois sous-langages qui proposent une expressivité croissante :

- **OWL Lite** : Le langage OWL Lite répond à des besoins de hiérarchie de classification et de fonctionnalités de contraintes simples, il ne permet que des valeurs de cardinalité de 0 ou 1.
- **OWL DL** : comprend toutes les structures de langage d'OWL avec des restrictions comme la séparation des types (une propriété doit être un individu ou une classe).
- **OWL Full** : est le langage complet. Il utilise tous les éléments disponibles en OWL. Il a l'avantage de la compatibilité complète avec RDF/RDFS, mais l'inconvénient d'avoir

un haut niveau de capacité de description, quitte à ne pas pouvoir garantir la complétude et la décidabilité des calculs liés à l'ontologie.

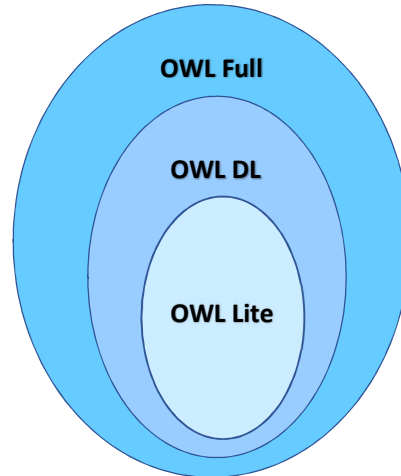


Figure 1.5: Les couches de l'OWL.

#### 4.5. SPARQL (SPARQL Protocol and RDF Query Language)

SPARQL [7] est un langage de requête et un protocole qui permet de rechercher, d'ajouter, de modifier ou de supprimer des données RDF disponible sur le Web. On pourrait oser dire que SPARQL est l'équivalent du langage SQL (Structured Query Language), le langage informatique normalisé qui sert à effectuer des opérations sur des bases de données. En utilisant SQL, on accède aux données d'une base de données via ce langage de requête, alors qu'avec SPARQL, on accède aux données structurées du Web des données. Cela signifie qu'en théorie, on pourrait accéder à toutes les données du Web avec ce standard, grâce à l'interopérabilité.

Les requêtes SPARQL [17] sont des requêtes sur les triplets qui constituent un graphe de données RDF.

SPARQL introduit quatre types de requêtes :

- **SELECT** qui renvoie la valeur de la variable.
- **ASK** qui renvoie vrai si la requête correspond, et faux dans le cas contraire
- **CONSTRUCT** qui retourne un graphe RDF en substituant les valeurs dans les modèles de données.
- **DESCRIBE** qui retourne un graphe RDF qui définit la ressource correspondante.

#### 4.6. SWRL (Semantic Web Rule Language)

SWRL est un langage standard basé sur OWL-DL et sur le langage Rule Markup Language (RuleML) qui fournit à la fois l'expressivité OWL-DL et les règles de RuleML [17].

Les règles de SWRL sont des règles d'implication. Par conséquent, la syntaxe de SWRL est de la forme suivante [18]

antécédent  $\rightarrow$  conséquent

Cette syntaxe implique que le conséquent doit être vrai lorsque l'antécédent est satisfait. Les expressions OWL peuvent se produire à la fois en antécédent et en conséquence [19].

## 5. Ontologie et Web sémantique

### 5.1. Définition de la notion ontologie

Le terme « ontologie » a suscité beaucoup d'intérêt dans plusieurs domaines et disciplines. La définition et l'utilisation de ce terme a subi des changements suivant son utilisation et selon les disciplines qui le traitent.

Dans [20] affirment qu'une ontologie définit les termes et les relations de base qui constituent le vocabulaire d'un domaine donné. En plus, l'ontologie doit établir les règles qui permettent la combinaison de ces termes et relations, afin de définir une extension de ce vocabulaire.

### 5.2. Composants d'une ontologie

Les connaissances traitées par l'ontologie sont modélisées à travers des éléments propres à cette dernière. Ces composants sont principalement concepts, relations, fonctions, axiomes et instances [21].

- **Les concepts** appelés aussi classes d'ontologie, modélisent une abstraction pertinente d'un segment du domaine traité. Selon [21], un concept peut être abstrait ou concret, atomique ou composé, réel ou fictif.
- **Les relations** constituent les associations qui peuvent exister entre les concepts. Elles assurent une certaine interaction entre les différents concepts. La relation sous-classe est un exemple d'une relation binaire entre deux classes.
- **Les fonctions** sont un cas particulier des relations dont un élément d'une relation est unique par rapport aux éléments qui le précèdent. Un exemple d'une fonction binaire est Mère-de qui donne la mère d'un individu. Ce dernier, doit avoir une seule mère.
- **Les axiomes** modélisent les connaissances considérées comme vrais dans le domaine traité.
- **Les instances** constituent des valeurs concrètes et des occurrences pour les concepts et les relations.

### 5.3. Classification des ontologies

La classification des ontologies permet de positionner chacune d'elles dans un contexte défini. Néanmoins, chaque ontologie peut être classée selon plusieurs dimensions. Ainsi, une

classification judicieuse doit prendre en compte une classification multidimensionnelle. Dans cette section, les principales classifications des ontologies en investissant trois dimensions, à savoir ; objet de conceptualisation, niveau de détail et niveau de formalisme de représentation.

### **5.3.1. Dimension objet de conceptualisation**

La classification selon l'objet de conceptualisation se divise principalement en :

#### **a. Ontologie de haut niveau**

L'ontologie de haut niveau est une ontologie générale. Elle s'intéresse à l'étude des catégories existantes dans notre monde (l'espace, le temps, les processus, les états...). Selon Guarino, une ontologie de haut niveau est basée sur trois théories fondamentales : la théorie de l'identité, la théorie de dépendance et la métrologie (étude des relations entre les parties et la totalité) [22]. Sowa étend cette vision en introduisant deux autres concepts ; Continuant et Occurrent [23]. En utilisant ces concepts et en combinant sept primitives, il obtient douze catégories dans l'ontologie de haut niveau qu'il propose.

#### **b. Ontologie générique**

L'ontologie générique appelée aussi méta-ontologie utilise des concepts moins abstraits que l'ontologie de haut niveau. Mais ces concepts sont assez généraux pour qu'ils puissent être utilisés dans des domaines précis [24]. Ce type d'ontologie s'adresse principalement à résoudre des problèmes génériques pouvant être utilisés à travers plusieurs domaines. Borst propose plusieurs ontologies de cette catégorie [25]. Il donne comme exemple l'ontologie métrologique et l'ontologie topologique.

#### **c. Ontologie de représentation de connaissances**

Le rôle de cette ontologie est de capturer les concepts permettant la représentation et la formalisation des connaissances. [26] propose les primitives utilisées dans les langages de frames (Classe, propriété, facette, instance...).

#### **d. Ontologie de domaine**

L'ontologie de domaine propose l'ensemble des termes et des concepts d'un domaine, comme scalpel ou scanner dans la médecine. Selon Mizoguchi, l'ontologie de domaine caractérise les connaissances du domaine où la tâche est réalisée [27].

#### **e. Ontologie de tâche**

Selon [28], le rôle de cette ontologie est de caractériser les connaissances concernant une tâche particulière. Cette dernière n'appartient pas nécessairement à un seul

domaine. Un exemple de cette ontologie pour le travail collaboratif est fourni dans le travail de [29].

#### **f. Ontologie d'application**

Cette ontologie est considérée comme la plus spécifique. Elle traite les connaissances liées à une application déterminée. Un nombre important des ontologies proposées appartient à cette catégorie. Maedche propose un ensemble d'ontologies pour la gestion des connaissances au sein de l'entreprise [30].

### **5.3.2. Dimension détail de l'ontologie**

Cette dimension de classification permet de donner un degré de détail approprié pour l'ontologie proposée. Elle permet aussi d'estimer son raffinement et sa complexité. On peut proposer deux niveaux de granularité pour exprimer ce degré de détail : granularité large et granularité fine.

#### **a. Granularité large**

Dans cette catégorie, même si l'ontologie traite un domaine ou une application précise, le vocabulaire est de nature générale. C'est le raffinement qui permet d'aboutir à des ontologies plus détaillées. On peut considérer une ontologie de granularité large comme étant une première version dans un processus de développement d'ontologies.

#### **b. Granularité fine**

Dans cette catégorie, l'ontologie possède un vocabulaire riche et détaillé. Elle permet de décrire de façon précise les concepts utilisés. Toute ontologie qui veut être opérationnelle, doit appartenir à cette catégorie.

### **5.3.3. Dimension niveau de formalisme**

Cette dimension permet de classer les ontologies selon le niveau de formalisme du langage de représentation utilisé. [31] traite cet aspect en indiquant que pour représenter une ontologie, on peut utiliser un langage formel ou informel. Une classification en quatre types de langage est donnée : informel, semi-informel, semi-formel et formel.

#### **a. Langage informel**

Représente le langage naturel. Ce langage est utilisé seulement dans la phase de discussion afin de présenter une vue globale de l'ontologie.

#### **b. Langage semi-informel**

Représente un langage naturel limité et structuré en éliminant d'éventuelles ambiguïtés.

**c. Langage semi-formel**

Représente un langage artificiel défini formellement. Le langage de modélisation UML (Unified Modeling Language) constitue un exemple de ce type. UML se base sur un langage graphique avec une définition rigoureuse pour les notations proposées.

**d. Langage formel**

Représente un langage artificiel contenant une sémantique formelle et des théorèmes et preuves sur ces propriétés.

Afin de résumer la classification, nous proposons la figure suivante :

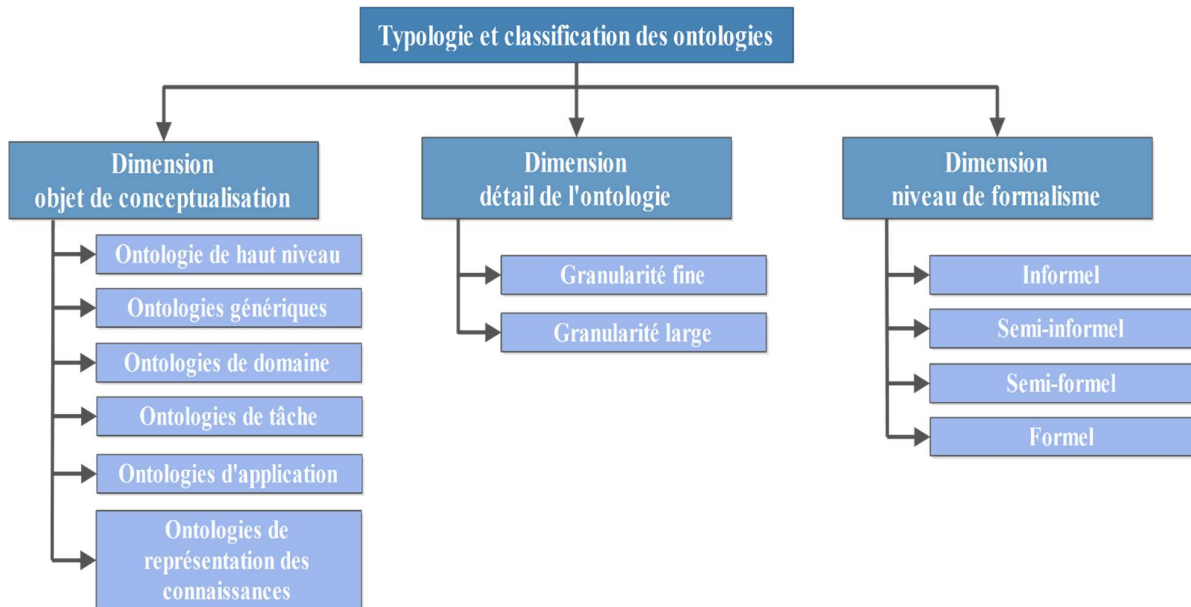


Figure 1.6: Typologie et classification des ontologies.

**5.4. Formalismes de représentation**

La représentation des connaissances propres à un domaine particulier consiste à décrire et à coder les entités de ce domaine d'une manière qu'elle soit manipulable par machine afin de raisonner. Comme alternative à la logique classique, l'intelligence artificielle a proposé divers formalismes de représentation, les plus utilisés pour représenter les ontologies sont :

**5.4.1. Langage de frame (frame-based languages)**

Le langage de Frame est un langage de programmation inventé par Marvin Minsky. Il repose sur la structure de donnée bénéficiant d'une structure globale standardisée appelée frame, également, un modèle conçu pour représenter les connaissances et permettant la réalisation de systèmes à base de règles et faits intimement liés, selon un principe similaire aux systèmes experts [32].

### 5.4.2. Graphe conceptuel

Introduit par J. SOWA au début des années 80, le modèle des Graphes Conceptuels (GC) appartient à la famille des réseaux sémantiques. Les réseaux sémantiques modélisent les connaissances sous forme de graphes, les nœuds étant associés à des concepts et les arêtes à des relations. Le modèle des GCs se décompose en deux parties [33] :

- Une partie terminologique dédiée au vocabulaire conceptuel des connaissances à représenter, c'est-à-dire les types de concepts, les types de relations et les instances des types de concepts. Cette partie correspond à la représentation du modèle conceptuel mais intègre également des connaissances sur la hiérarchisation des types de concepts et de relations.
- Une partie assertionnelle dédiée à la représentation des assertions du domaine de connaissance étudié.

### 5.4.3. Logique de description

Les logiques de description peuvent être considérées comme un fragment de la logique du premier ordre, dans lequel les formules ont une variable libre pour les descriptions de concept et deux variables libres pour les descriptions de relations [34].

Une LD est composée de deux parties : un langage terminologique TBOX et un langage assertionnel ABOX. Le langage assertionnel est dédié à la description de faits et le langage terminologique à la description de concepts et de rôles. La principale tâche de raisonnement au niveau terminologique est de calculer les relations de subsomption entre concepts [35].

Les entités de base qui sont définies et manipulées dans une logique de descriptions sont les concepts et les rôles. Un concept permet de représenter un ensemble d'individus, et un rôle représente une relation binaire entre concepts.

## 5.5. Outils de construction d'ontologies

On distingue deux types d'outils pour la construction des ontologies : les outils de construction d'ontologie dépendants du formalisme de représentation et les outils de construction d'ontologie indépendants du formalisme de représentation.

### 5.5.1. Outils dépendants du formalisme de représentation

#### a. WebOnto [36]

WebOnto est une application Web pour naviguer et développer de façon collaborative les ontologies. WebOnto supporte la navigation collaborative, la création et l'édition d'ontologies sur le Web. Les ontologies WebOnto sont implémentées dans le langage OCML, qui est une combinaison des frames et de la logique de premier ordre et qui permet de

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

**b. OilEd [37]**

OilEd (Oil Editor) est un éditeur d'ontologies utilisant le formalisme OIL. Il est essentiellement dédié à la construction de petites ontologies dont on peut ensuite tester la cohérence à l'aide de FACT (un moteur d'inférences bâti sur OIL). Le modèle de connaissance a été adapté depuis OIL à DAML+OIL et maintenant il va être adapté à OWL. [37]

**5.5.2. Outils indépendants de formalisme de représentation**

**a. ODE et WebOde**

L'outil ODE (Ontology Design Environment) permet de construire des ontologies au niveau connaissance, comme le préconise la méthodologie METHONTOLOGY. L'utilisateur construit son ontologie dans un modèle de type frame, en spécifiant les concepts du domaine, les termes associés, les attributs et leurs valeurs, les relations de subsomption. L'ontologie opérationnelle est alors générée en utilisant les formalismes ONTOLINGUA ou FLOGIC. [38] [39]

**b. OntoEdit**

OntoEdit [40] (Ontology Editor) est également un environnement de construction d'ontologies indépendant de tout formalisme. Il permet l'édition des hiérarchies de concepts et de relations et l'expression d'axiomes algébriques portant sur les relations, et de propriétés telles que la généralité d'un concept. Des outils graphiques dédiés à la visualisation d'ontologies sont inclus dans l'environnement. OntoEdit intègre un serveur destiné à l'édition d'une ontologie par plusieurs utilisateurs. Un contrôle de la cohérence de l'ontologie est assuré à travers la gestion des ordres d'édition.

**c. Protégé**

Protégé [41] est un éditeur d'ontologies distribué en open source par l'université en informatique médicale de Stanford. Il 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 aussi une plate-forme extensible, grâce au système de plug-ins, qui permet de gérer des contenus multimédias, interroger, évaluer et fusionner des ontologies, etc. L'outil Protégé

possède une interface utilisateur graphique (GUI) lui permettant de manipuler aisément tous les éléments d'une ontologie : classe, propriété, instance. Protégé peut être utilisé dans n'importe quel domaine où les concepts peuvent être modélisés en une hiérarchie des classes.

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

## 6. Moteurs d'inférence

Pour exploiter une ontologie et les règles, un ou des moteurs d'inférences sont requis. De nombreux moteurs d'inférences existent tels que Pellet [42], KAON [43], Hermit [44], FACT++ « Fast Classification of Terminologies » [45], Racer « Renamed ABox and Concept Expression Reasoner » [46], pour ne citer que ceux-là. Ces moteurs d'inférences sont tous déductifs. Ils permettent de [47] :

- Vérifier la cohérence pour s'assurer que l'ontologie ne contienne pas de fait contradictoire. En logique de description, cette opération vérifie si la ABox est conforme à la TBox.
- Vérifier la satisfiabilité des concepts pour détecter les classes qui peuvent ne pas avoir d'instances.
- Faire de la classification qui permet de dire qu'une classe donnée est sous-classe d'une autre. Cette opération crée une hiérarchie de classes et elle permet de produire des réponses à d'autres types de requêtes comme : lister toutes les sous-classes directes d'une classe donnée.
- Faire de la réalisation qui permet de trouver les classes les plus spécifiques à une instance.

## 7. Conclusion

Le Web sémantique désigne un ensemble de technologies visant à rendre le contenu des ressources du Web accessible et utilisable par les machines et agents logiciels, grâce à un système de métadonnées formelles, utilisant une famille de langages développés par le W3C. Ainsi que les ontologies qui jouent un rôle important dans la réutilisation et le partage de données, elles permettent de faciliter la communication entre les acteurs de différentes organisations et en particulier, la réalisation de l'interopérabilité entre différents systèmes, elles permettent aussi non seulement la création de systèmes mais capables de gérer et de raisonner sur ces connaissances.

# CHAPITRE 2

## RESEAUX DE CAPTEURS SANS FILS ET WEB DE CAPTEURS

Les progrès récents dans la technologie des systèmes micro électromécaniques, les communications sans fil et l'électronique numérique ont permis le développement de nœuds de capteurs multifonctionnels à faible coût et puissance, de petite taille et communiquant librement sur de courtes distances. Ces minuscules nœuds de capteurs, constitués de composants de détection, de traitement de données et de communication ont permis le développement des réseaux de capteurs basés sur l'effort collaboratif d'un grand nombre de nœuds [48].

Ce type de réseau constitue un grand intérêt dans la recherche en informatique et en sciences physiques, et leur utilisation pour la surveillance scientifique des environnements éloignés et hostiles est de plus en plus commune.

Les réseaux de capteurs sans fil (WSN) peuvent être définis comme des réseaux sans fil autoconfigurés et sans infrastructure pour surveiller les conditions physiques ou environnementales telles que la température, le son, les vibrations, la pression, les mouvements ou les polluants, en transmettant leurs données à un emplacement sur le réseau ou ce qui est appelé un puits, où les données peuvent être mitigées, observées et analysées [49]. La multiplication de ce type de réseau et le manque d'intégration isolent souvent le flux de données importants et intensifie le problème existant de trop de données et pas assez de connaissances. Ce problème doit être résolu dans des environnements ouverts qui brisent les cloisons entre les différents réseaux de capteurs en permettant l'exploitation de données à grande échelle, et avec une possibilité de faire des analyses sémantiques basées sur des interprétations correctes, homogènes et uniques de données en vue d'extraire des informations pertinentes.

A cet effet ; les technologies du Web sémantique ont suscité l'attention des professionnels des réseaux de capteurs et des informaticiens pour créer et développer le Web des capteurs sémantique (SSW) dans lequel les données de capteur sont annotées avec des méta-données sémantiques spatiales, temporelles et thématiques afin d'assurer l'interopérabilité et fournir des informations contextuelles essentielles pour mieux comprendre une situation quelconque ou un phénomène.

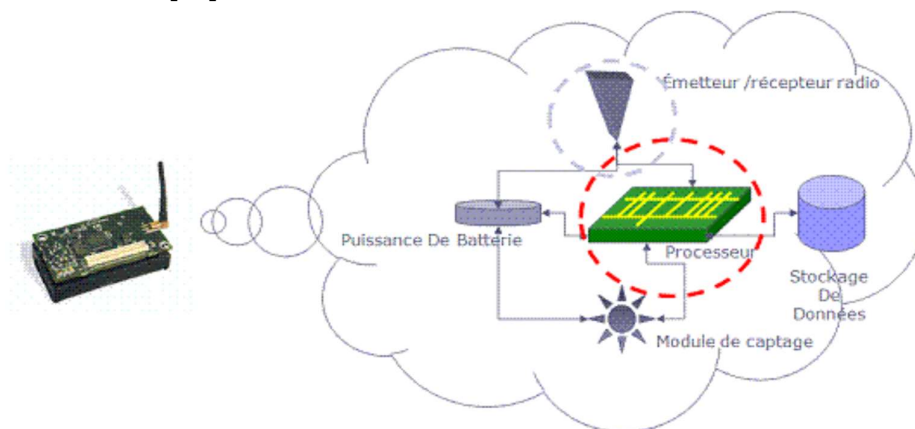
## 1. Capteurs, systèmes de capteurs et ressources de capteurs

### 1.1. Définition d'un capteur

Un capteur est une entité qui réagit à une sorte de stimulus physique ou chimique [50]. Il est de taille extrêmement réduite avec des ressources très limitées, autonomes, capable de traiter des informations et de les transmettre, via les ondes radio, à une autre entité sur une distance limitée à quelques mètres [49].

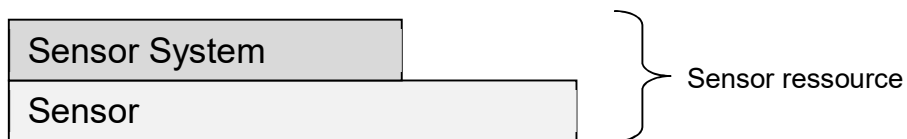
### 1.2. Anatomie d'un nœud de capteur

Un nœud capteur est composé principalement d'un processeur, une mémoire, un émetteur/récepteur radio, un ensemble de capteurs, et une pile (Figure 2.1). Il existe plusieurs modèles commercialisés dans le marché. Parmi les plus célèbres, les capteurs MICAx et TelosBe de Crossbow [51].



**Figure 2.1 :** Anatomie d'un nœud de capteur.

Un certain nombre de capteurs agrégés en une seule unité, fournissant une seule interface d'accès, est considéré comme un système de capteur [50]. Par exemple une station météorologique représente un système de capteurs composé d'un thermomètre, un pluviomètre et un capteur de vitesse du vent. Une « ressource de capteur » se réfère à un « capteur » et « système de capteur » comme une, illustré à la Figure 2.2



**Figure 2.2 :** Système de capteurs et capteurs en tant que ressources de capteur.

### 1.3. Limites des ressources de capteurs

Les ressources de capteurs typiques, que soit des plates-formes de capteurs sur lieu ou distantes, fournissent des observations spatio-temporelles discrètes liées à certaines propriétés physiques, souvent appelées un « phénomène ».

La ressource de capteur renvoie un résultat d'observation localisé et fournit de manière isolée sans contexte. Le manque de contexte résulte d'une collaboration limitée entre les ressources du capteur. Une ressource de capteur est capable de donner un retour sur l'état d'une partie du phénomène observé mais pas sur la dynamique du phénomène observé dans son ensemble [52].

Les ressources de capteur isolées présentent un problème de robustesse. Si une ressource de capteur échoue, les observations sur le phénomène seront réduites ou perdues [52].

Le besoin d'une approche plus robuste et dynamique des observations peut être remarquable, cette approche comprendrait des ressources de capteurs interconnectés qui seraient capables de contrôler, de charger et de répondre à d'autres capteurs et plates-formes afin de fournir une vue cohérente et continue sur des phénomènes observés.

## 2. Réseaux de capteurs, une valeur ajoutée à la communication

### 2.1. Définition des RCSF

Un RCSF (Réseau de Capteurs Sans Fil) est composé d'un ensemble de nœuds capteurs. Ces capteurs sont organisés en champs « Sensor Fields » (figure 2.3). Chacun de ces nœuds a la capacité de collecter des données et de les transférer à une passerelle qui représente un puits de données par l'intermédiaire d'une architecture multi-sauts. Le puits transmet ensuite ces données par Internet ou par satellite à l'ordinateur central « Gestionnaire de tâches » pour analyser ces données et prendre des décisions [51].

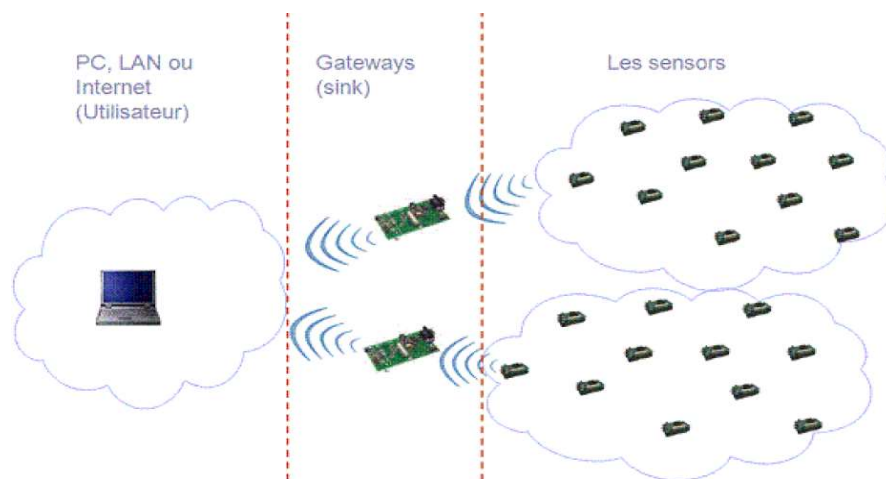
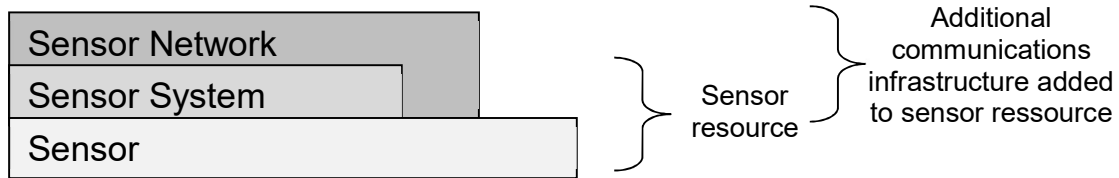


Figure 2.3 : Architecture d'un RCSF.

Les réseaux de capteurs offrent une approche d'observation robuste et dynamique. Au lieu de déployer une seule ressource sensorielle pour observer un phénomène, un réseau de capteurs est déployé. Un réseau de capteurs peut être considéré comme une « collection de ressources de capteurs en réseau » [53], comme le montre la figure 2.4.



**Figure 2.4 :** L'ajout de l'aspect communicatif aux ressources de capteurs crée les réseaux de capteurs.

Les réseaux de capteurs, bien que très utiles en tant que systèmes d'observation, ils sont sujet de quelques problèmes parmi lesquels on peut citer :

- Les réseaux de capteurs sont déployés de telle sorte que chacun existe dans son propre silo. Le réseau de capteurs n'a connaissance d'aucune autre ressource de capteur en dehors de ses propres limites de système. Les utilisateurs du réseau de capteurs utilisent le réseau de capteurs à l'intérieur de ce silo [54].

### 3. Web des capteurs

Le concept de Web de capteurs est né à la NASA/Jet Propulsion Laboratoire à la fin des années 1990 [55] [56]. Une nouvelle élaboration du concept du Web des capteurs, décrit un «Sensor Web» à un méta-niveau comme étant un système global distribué de capteurs, des observations de capteurs traitées, des bases de données, pouvant être configurées dynamiquement pour élaborer une synthèse d'informations sur une grande zone spatiale afin de suivre des phénomènes dynamiques [55] [56].

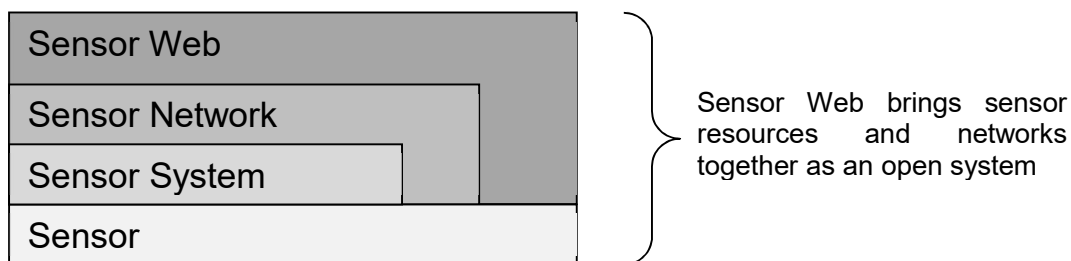
L'initiative SWE (Sensor Web Enablement) dirigée par le groupe l'OGC (Open Geospatial Consortium), qu'elle sera détaillée par la suite, a reconnu une large adoption dans le domaine du Web des capteurs, grâce à laquelle les ressources des capteurs sont interopérables et disponibles sur Internet via des interfaces standardisées, des codages de données et des métadonnées. Cela permettrait un traitement automatisé des données des capteurs et même l'attribution de tâches aux capteurs pour acquérir des observations intéressantes [57] [58].

#### 3.1. Web de Capteurs, une ouverture des réseaux de capteurs

L'effet «système dans les silos» des réseaux de capteurs évoqué auparavant nécessite certaines correctives, et afin de surmonter ces limitations il est question de trouver un mécanisme permettant de combiner les réseaux de capteurs pour former un seul et même macro-instrument,

cette idée a donné naissance au nouveau concept du Web des capteurs, qui est défini en tant que type particulier d'infrastructure d'information centrée sur le Web pour la collecte, la modélisation, le stockage, la récupération, le partage, la manipulation, l'analyse et la visualisation d'informations sur les capteurs et les observations de capteurs [59].

Il décrit aussi une couche complexe fournissant une fonction ombrelle au-dessus des capteurs, des systèmes de capteurs et des réseaux de capteurs [60] comme le montre la Figure 2.5.



**Figure 2.5 :** Le Web des capteurs rassemble tous les éléments de manière ouverte.

De nombreuses définitions examinent le Web des capteurs en termes de fonctionnalités à fournir ou en termes d'environnements ou d'infrastructures, une autre définition qui prend une vue purement systèmes définit le Web des capteurs en tant que système adaptatif complexe, ouvert et organisé comme un réseau de ressources de capteurs ouvertes, qui imprègne l'Internet et fournit un accès externe aux ressources de capteurs. Avec l'ouverture des ressources de capteurs, on peut inclure tout système ouvert, y compris un réseau de capteur, qui est une source de données de capteurs ou de métadonnées de capteurs [50].

### 3.2. Couches inter-réseaux du Web de capteurs

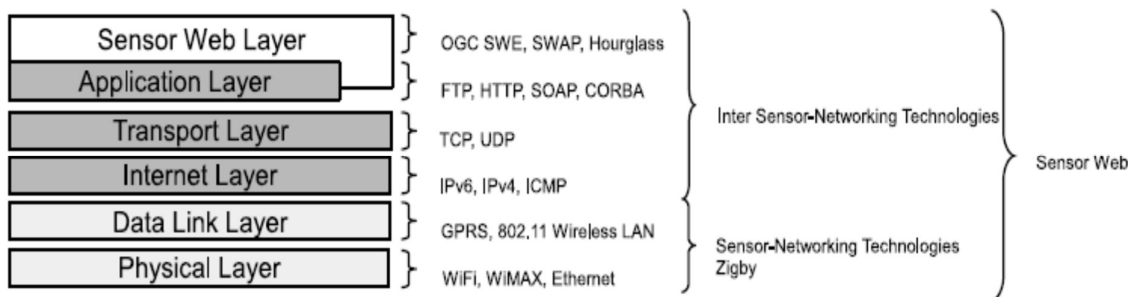
Le Web des capteurs, d'un point de vue purement réseau, peut être vu comme un certain nombre de couches. Chacune des couches présente un niveau d'abstraction supplémentaire et fournit des opportunités supplémentaires pour une communication ouverte entre les ressources du capteur. Les couches constituant la pile de technologies de mise en réseau inter-capteurs sont étroitement liées aux couches qui composent la pile inter-réseau d'Internet (TCP/IP, par exemple), du fait de l'infiltration du Web des capteurs dans Internet [61].

La pile de technologies de mise en réseau de capteurs est incluse pour afficher le niveau auquel les réseaux de capteurs peuvent commencer à exister.

La figure 2.6 montre la pile Web des capteurs qui est grossièrement divisée en deux. La moitié inférieure représente les technologies de mise en réseau de capteurs qui fournit l'infrastructure pour les réseaux de capteurs. La moitié supérieure des technologies de mise en réseau inter-

capteurs fournit l'infrastructure pour faciliter l'ouverture et l'interopérabilité des réseaux de capteurs et ainsi faire partie du Web des capteurs.

Actuellement, un certain nombre de technologies de mise en réseau inter-capteurs, à tous les niveaux de la pile, ont été proposées en tant que mécanismes pour fournir une infrastructure du Web des capteurs.



**Figure 2.6 :** La pile de couches inter-réseau de capteurs du Web des capteurs.

### 3.3. Norme SWE (Sensor Web Enablement) du groupe OGC

L'initiative SWE de l'OGC a permis de développer un ensemble de normes et spécifications ; parmi lesquelles on peut citer :

**a.** Le Sensor Modeling Language (SensorML) [57] est basé sur le langage de modélisation XML. SensorML fournit le mécanisme pour décrire toute la gamme allant des capteurs simples aux systèmes de capteurs complexes arbitraires et à ses plates-formes correspondantes. Chaque capteur est modélisé comme un processus faisant partie intégrante d'un système.

**b.** Observations and Measurements (O&M) [62] fournit un modèle conceptuel et un codage de données basé sur XML. O&M fournit les mécanismes de codage des observations et des mesures de manière à faciliter la fusion et à fournir suffisamment de métadonnées pour permettre le traitement automatisé des données.

**c.** Le service d'observation de capteur (SOS) [63] fournit une interface de service Web standard pour extraire des données de capteur à partir de sources arbitraires. Un SOS fournit aux utilisateurs la possibilité de récupérer des observations en temps réel.

**d.** Le SensorAlerting Service (SAS) [64], fournit une interface de service Web standard qui permet aux utilisateurs de s'abonner à des alertes. Le SAS envoyait l'alerte directement à ses abonnés ou utilisait le service de notification Web pour alerter un utilisateur de cet événement.

**e.** Le service de planification de capteurs (SPS) [65] fournit une interface de service Web standard qui permet aux utilisateurs de charger un capteur. Un utilisateur chargerait un capteur

de faire une observation, de (re) calibrer le capteur ou la plate-forme, ou de s'adapter à de nouvelles conditions, soit à l'intérieur du système, soit dans l'environnement.

#### **4. Web des capteurs sémantique**

Bien que le Web des capteurs a permis une ouverture à grande échelle des réseaux de capteurs, notamment avec l'initiative SWE de l'OGC qui constitue une grande étape vers l'intégration de données provenant de divers réseaux de capteurs pour déployer des applications Web de capteurs sur Internet, et du fait que cette initiative est basée essentiellement sur XML. Les données manipulées par les différentes technologies du Web des capteurs restent toujours incompréhensibles par les machines vus le manque d'une sémantique bien définie, ceci dû à l'incapacité du langage XML à décrire leurs significations. D'où la nécessité d'introduire d'autres technologies permettant d'ajouter des annotations sémantiques sur les données de capteur en fournissant des descriptions plus significatives et un accès amélioré que SWE seul. Le SSW (Semantic Sensor Web) est un Framework pour fournir un sens bien défini aux données issues de capteurs. En association avec l'annotation sémantique, les ontologies et les règles jouent un rôle important dans le SSW pour l'interopérabilité, l'analyse et le raisonnement sur les données de capteurs hétérogènes.

##### **4.1. Métadonnées des capteurs dans l'espace, le temps et le thème**

Les données issues des capteurs sont par nature opaques (souvent dans des formats binaires ou propriétaires) ; par conséquent, les métadonnées jouent un rôle essentiel dans la gestion de données des capteurs. Un réseau de capteurs sémantiquement riche fournirait des informations spatiales, temporelles et thématiques essentielles à la découverte et à l'analyse des données de capteurs.

- Les métadonnées spatiales fournissent des informations sur l'emplacement et les données du capteur, en termes de système de référence géographique, de référence locale ou d'emplacement désigné [59].
- Les métadonnées temporelles fournissent des informations sur l'instant ou l'intervalle de temps lorsque les données du capteur sont capturées [59].
- Les métadonnées thématiques décrivent un état du monde réel à partir des observations de capteurs, tels que des objets ou des événements [59]. Chaque discipline contient des informations spécifiques au domaine, telles que des concepts décrivant les phénomènes météorologiques ou les événements biomédicaux représentant l'état de santé d'un patient.

## 4.2. Ontologies dans le domaine du Web de capteurs sémantique

Dans le premier chapitre nous avons repris qu'une ontologie définit les termes utilisés pour décrire et représenter un champ d'expertise. Dans le domaine du SSW (Sensor Web Semantic), les ontologies ont été classées selon les trois types de sémantique associés aux données de capteurs à savoir [59] :

- **Spatiale** : Une ontologie spatiale définit les concepts généraux des objets spatiaux et leurs relations avec les domaines d'application spatiale. Parmi lesquelles on peut citer WGR-GEO développée par le groupe W3C pour de nombreuses applications telles que le routage, l'observation de la terre et l'interprétation des images.
- **Temporelle** : Une ontologie temporelle définit des concepts temporels, pour décrire les propriétés temporelles des ressources. Elle fournit des informations sur les durées, et la position temporelle, y compris la date et l'heure. Parmi lesquelles on peut citer OWL-Time utilisé pour décrire le contenu temporel des pages Web et les propriétés temporelles des services Web.
- **Thématique** : Une ontologie qui représente un domaine ou un champ d'application spécifique tel que la météorologie. La figure 2.7 montre un sous-ensemble de concepts d'une suite d'ontologies dans SSW et leurs relations, modélisant le domaine météorologique [59].

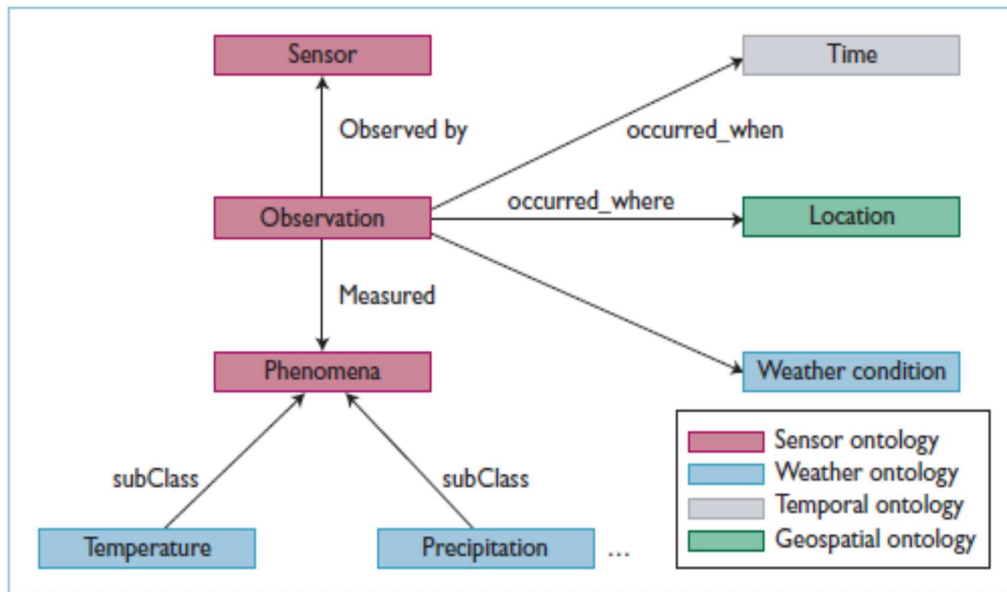


Figure 2.7 : Sous-ensemble de concepts et relations représentés avec une suite d'ontologies.

### 4.3. Architecture Web sémantique pour les réseaux de capteurs (SWANS)

Le domaine du SSW fait l'objet de plusieurs travaux de recherche visant entre autres à trouver une meilleure architecture pour représenter les réseaux de capteurs et les informations en résultant. Dans le cadre de notre projet, nous allons présenter une architecture proposée dans un article de recherche [65] pour le traitement des informations issues des capteurs appelée SWANS (Architecture Web Sémantique pour Réseaux de Capteurs). Cette architecture permet une meilleure compréhension de l'information véhiculée dans les données du capteur par la machine et un traitement de l'information avancé et adapté aux exigences des applications. L'architecture SWANS est divisée en quatre couches (Figure 2.8).

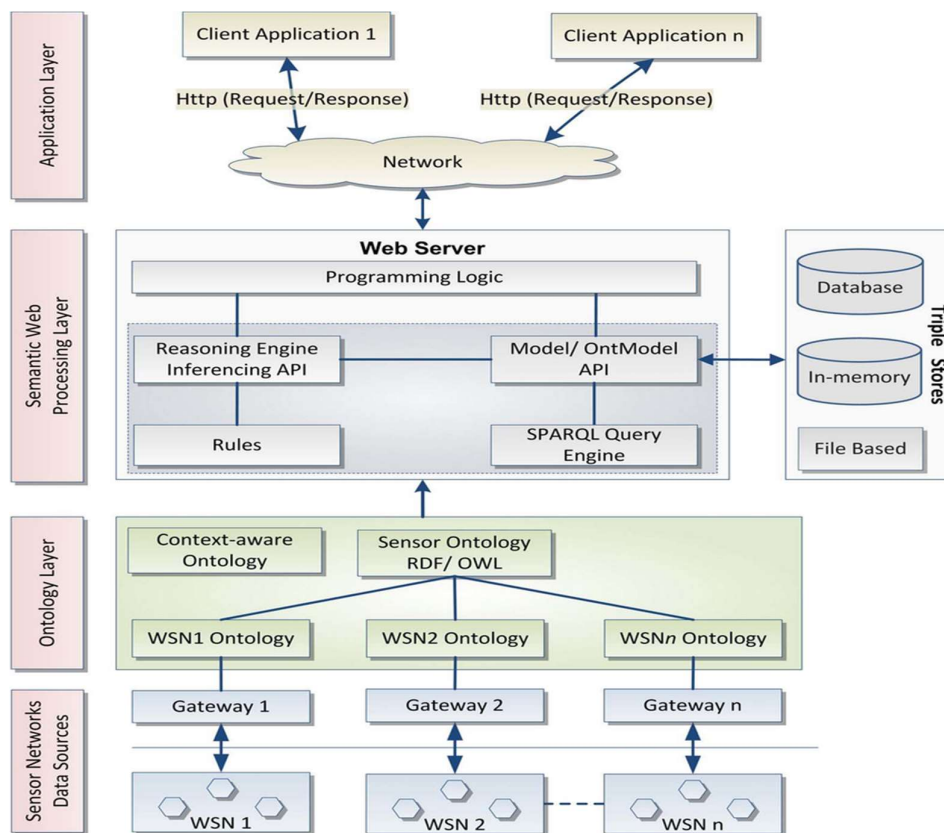


Figure 2.8 : Une architecture du Web des capteurs sémantique.

#### 4.3.1. Sources de données du réseau de capteurs

La première couche représente la source de données issues de réseaux WSN hétérogènes qui est constitués de différents nœuds de capteurs. Les données provenant des nœuds de capteurs sont collectées et peuvent être accédées via une passerelle de capteur standard.

#### 4.3.2. Couche d'ontologie

Une fois que les données du capteur sont disponibles via une passerelle, la sémantique de ces données doit être décrite. Cette deuxième couche de cette architecture est constituée

d'ontologies de capteurs. Chaque WSN peut avoir sa propre ontologie locale. Les concepts issus des ontologies locales seront décrits en utilisant les termes globaux définis dans l'ontologie des capteurs globale. Cette approche offre plus de flexibilité et d'extensibilité, car de nouveaux WSN peuvent facilement être ajoutés sans avoir besoin d'être modifiés dans d'autres ontologies locales ou dans un vocabulaire partagé. Les données de capteur qui proviennent des nœuds de capteurs à travers la passerelle, correspondent aux ontologies de cette couche.

#### **4.3.3. Couche de traitement du Web sémantique**

Les données du capteur peuvent être traitées après avoir défini leurs sémantiques en les mappant à l'ontologie. La couche d'ontologie interagit directement avec la couche de traitement Web sémantique. Elle peut aussi créer un modèle simple dans RDF ou un modèle d'ontologie dans OWL. Les données de capteur peuvent ensuite être stockées temporairement dans la mémoire principale ou sur une base de données persistante ou un système basé sur des fichiers.

#### **4.3.4. Couche d'application**

La couche d'application est constituée de différentes applications clientes nécessitant les données des capteurs. Ces dernières et après qu'elles soient traitées, elles peuvent être disponibles pour ces applications via n'importe quel serveur Web sur Internet ou intranet. L'interaction entre les applications clientes et le serveur Web s'effectue via HTTP.

### **4.4. Traitement de données de capteur**

Pour le traitement des données de capteurs, l'API Jena peut être utilisé. Les deux principaux aspects du traitement de données sont :

- L'interrogation ; une fois que les données de capteur brutes sont transformées au format RDF/OWL ; SPARQL peut être utilisé pour exécuter des requêtes.
- L'inférence, un raisonnement peut être effectué, sur les données de capteurs pour déduire de nouvelles informations ou comprendre une nouvelle situation édictée par un changement de l'environnement.

## **5. Quelques domaines d'applications des RCSF**

Les réseaux de capteurs sans fil ont gagné une popularité considérable en raison de leur flexibilité dans la résolution de problèmes dans différents domaines d'application, et leur potentiel de changer notre vie de différentes manières. Les WSN ont été appliqués avec succès dans diverses applications (figure 2.9).



Figure 2.9 : Application des RCSF.

### 5.1. Détection des incendies de forêts

Les capteurs peuvent être déployés d'une façon aléatoire, et dense dans une forêt, ils peuvent facilement détecter et rapporter l'origine de l'incendie à l'utilisateur avant qu'il se propage et devienne incontrôlable. Ces nœuds doivent cependant pouvoir fonctionner d'une manière autonome pendant une longue durée qui peut aller jusqu'à des années, pour cela ils doivent être équipés par des systèmes de rechargement d'énergie efficaces tels que les panneaux solaires. De plus, et pour faire une surveillance distribuée, ils doivent pouvoir collaborer et surmonter tous les obstacles qui existent tel que les rochers et les arbres [66].

### 5.2. Agriculture

Les réseaux de capteurs sont capables d'apporter des bénéfices considérables au domaine de l'agriculture, grâce à leur habilité de surveiller les taux de pesticides dans l'eau, le niveau de pollution de l'air en temps réel sur des surfaces agricoles, on peut aussi questionner le réseau de capteurs sur l'état du champ pour déterminer les surfaces les plus secs afin de les arroser en priorité.

### 5.3. Contrôle de la pollution

On pourrait disperser des capteurs au-dessus d'un emplacement industriel pour détecter et contrôler des fuites de gaz ou de produits chimiques. Ces applications permettraient d'alerter les utilisateurs en un temps record et de pouvoir suivre l'évolution de la catastrophe.

#### **5.4. Applications militaires**

Comme dans le cas de plusieurs technologies, le domaine militaire a été un moteur initial pour le développement des réseaux de capteurs. Dans le domaine militaire, un réseau de capteurs peut être rapidement déployé dans un environnement sans infrastructure, notamment pour surveiller toutes les activités des forces ennemies, ou d'analyser le terrain avant d'y envoyer des troupes [67].

#### **5.5. Surveillance médicale**

En implantant sous la peau de mini capteurs, on peut recevoir des informations en temps réel d'une partie du corps sans aucune chirurgie. On peut aussi surveiller la progression d'une maladie en collectant des informations physiologiques (rythme cardiaque, taux de glucose, tension artérielle, ... etc.) sur des patients pour faciliter le diagnostic de quelques maladies [67].

### **6. Conclusion**

Il est incontestable que les différentes technologies du Web sémantique ont influencé le développement des réseaux de capteurs, ce qui a donné naissance au Web des capteurs sémantiques, qui est à la fois une vision futuriste et une réalité. C'est une vision dans le sens où elle ne cesse de croître. De nouvelles normes, de nouveaux services, de nouvelles ontologies seront développées pour décrire la sémantique des données des capteurs, ce qui facilitera le partage et la réutilisabilité de ces données et par conséquent construire des applications sur mesure permettront une meilleure compréhension de notre environnement. C'est une réalité dans la mesure où les premières applications ont déjà vu le jour dans différents domaines.

# CHAPITRE 3

## CONCEPTION DE L'ONTOLOGIE

*”Il y a une science qui étudie l'être en tant qu'être et les attributs qui lui appartiennent essentiellement“*

Aristote

Ce chapitre sera consacré à la présentation de l'ontologie qu'on a conçue dans le cadre de ce projet, la conception a été faite en reposant sur une méthodologie, pour ce faire nous nous sommes basés sur la méthode METHONTOLOGIE qui est le support de base du processus de construction de l'ontologie qui commence par la spécification qui permet de décrire l'ontologie puis la conceptualisation de l'ontologie à créer, à travers un ensemble de représentations intermédiaires semi-formelles en suite la formalisation de l'ontologie avec un langage complètement formel et opérationnel.

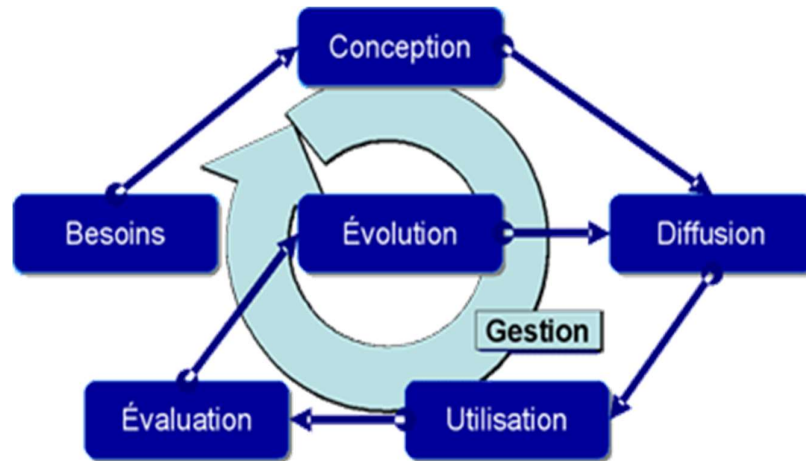
### **1. Ontologies des réseaux de capteurs**

Semantic Sensor Network sont basés sur des ontologies de référence pour annoter des données, des dispositifs et des services. Il existe de nombreuses propositions d'ontologies, dont l'objectif et la portée varient grandement. OntoSensor [68], CSIRO [69] et l'ontologie SSN [70] sont parmi les plus pertinents et largement utilisés. L'ontologie de réseau de capteurs sémantiques (SSN) est une ontologie de capteur générale qui fournit une norme unifiée pour décrire les données de réseau de capteurs. Elle a été créée par le groupe de travail W3C SSN Incubator Group. Elle a été utilisée dans plusieurs projets.

### **2. Cycle de vie d'une ontologie**

Un cycle de vie d'une ontologie débute du besoin qui se transforme en idée, la concrétisation de l'idée qui se traduit par la conception qui est diffusée pour son utilisation. Vient ensuite l'étape de l'évaluation qui donne naissance, le plus souvent à une étape d'évolution et de maintenance du modèle. Une réévaluation de l'ontologie et des besoins devra se faire après chaque utilisation significative. [71] Les ontologies sont des objets vivants, et

chaque étape de leur cycle de vie pose des problèmes de recherche. Ce cycle de vie rassemble sept activités : détection des besoins, conception, gestion et planification, évolution, diffusion, utilisation, évaluation. [72]



**Figure 3.1** : Le cycle de vie d'une ontologie.

### **3. Méthodologie de construction d'une ontologie « METHONTOLOGIE »**

C'est une méthode qui a été développée au laboratoire d'intelligence artificielle de l'université de Madrid. METHONTOLOGY [73] se veut être une approche intégrée s'inscrivant dans le cadre dans un processus de gestion de projet complet, allant de la spécification des besoins jusqu'à la phase réalisation et maintenance. L'approche METHONTOLOGY distingue les étapes suivantes : Spécification, Conceptualisation, Formalisation, Intégration, Implémentation, Maintenance.

#### **4. Processus de construction de l'ontologie**

Nous utilisons un processus de construction dans le développement de l'ontologie partant de connaissances brutes et arrivant à une ontologie d'application opérationnelle représentée par le langage OWL.

##### **4.1. Spécification**

Cette étape 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 quatre aspects suivants :

- a. Domaine de connaissance** : cet aspect consiste à déterminer précisément que possible le domaine que va couvrir l'ontologie.
- b. Objectifs** : cet aspect précise le but de l'ontologie à créer pour le domaine considéré.

- c. **Utilisateurs** : cet aspect consiste à identifier les utilisateurs prévus de l'ontologie à créer.
- d. **Portée de l'ontologie** : cet aspect consiste à déterminer a priori la liste des termes les plus importants pour le domaine à représenter.

#### 4.2. Conceptualisation

Cette étape est la plus importante dans le processus de construction de l'ontologie présentée par METHONTOLOGY, elle consiste à identifier et structurer les connaissances du domaine en utilisant un ensemble de représentations semi-formelles indépendantes des paradigmes de la représentation des connaissances, faciles à comprendre par les experts et dans lesquels l'ontologie sera formalisée. Pour cela on distingue les principales tâches suivantes :

- a. Construction du glossaire de termes.
- b. Construction du diagramme de classification des concepts.
- c. Construction du diagramme de relations binaires.
- d. Construction du dictionnaire de concepts.
- e. Décrire les relations dans une table de relations binaires.
- f. Spécifier des contraintes sur les attributs dans une table d'attributs.
- g. Spécifier des axiomes sur les concepts dans une table d'axiomes logiques.
- h. Décrire les instances des concepts dans une table d'instances.
- i. Construction de la table des assertions.

A la fin de cette étape, nous obtenons une ontologie conceptuelle.

#### 4.3. Formalisation

Dans cette étape on s'intéresse à la formalisation de l'ontologie conceptuelle obtenue précédemment 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 qui permet de représenter les connaissances relatives à un domaine de référence à l'aide de "descriptions" qui peuvent être des concepts, des rôles et des individus.

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.

## 5. Conception de notre ontologie

Dans cette section, nous construisons notre ontologie qui est une ontologie de réseau de capteurs pour la météorologie basée sur l'ontologie SSN. A cette fin, nous suivons les étapes du processus de construction d'ontologie proposé avant.

### 5.1. Spécification

Le développement de l'ontologie est débuté par la phase de spécification qui consiste à établir un document de spécification des besoins. Au sein de ce document, nous décrivons l'ontologie à construire à travers les quatre aspects suivants :

**a. Domaine de connaissance :** L'ontologie que nous venons de construire, s'inscrit dans le cadre des réseaux de capteurs sans fil, ainsi le domaine que va couvrir l'ontologie est la météorologie.

**b. Objectif :** l'objectif majeur de l'incorporation des ontologies au sein des réseaux de capteurs est l'enrichissement sémantique des descriptions des capteurs et des flux de capteurs et l'interopérabilité entre les données de capteurs provenant de différentes sources.

**c. Utilisateurs :** cet aspect présente l'ensemble des utilisateurs pouvant exploiter l'ontologie. Dans notre cas, les utilisateurs de l'ontologie représentent les applications Web qui doivent exploiter les ontologies afin d'atteindre l'objectif visé.

**d. Portée de l'ontologie :** Déterminer les termes globaux les plus importants référant les entités de la connaissance du domaine à représenter. Dans le contexte du réseau de capteurs pour la météorologie, nous pouvons déterminer les termes globaux suivants : {System, Sensor, SensorOutput, Phenomena ...}.

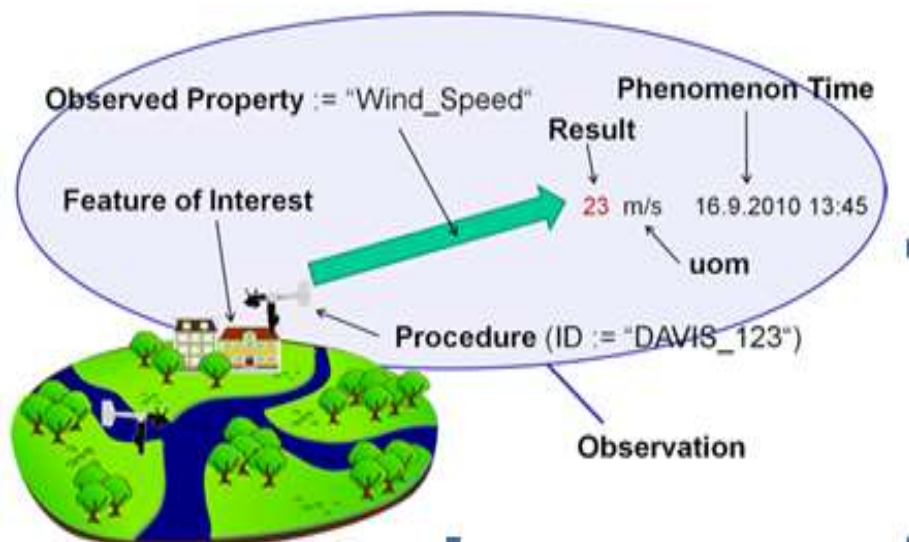


Figure 3.2: Termes générale de notre modèle ontologique.

Pour en résumer un document de spécification est illustré dans la figure suivante :

```

<rdf:RDF>
<rdf:Description about="URI of the ontologie">
<Domaine>Wireless Sensor Network</Domaine>
<Date>march 2018</Date>
<Développeur par>
<rdf:1_M. HEMAM, University of Abbes Laghrour Khenchela>
<rdf:2_F.Z. AMARA, University of Abbes Laghrour Khenchela>
<rdf:3_K. FALEK, University of Abbes Laghrour Khenchela>
</Développeur par>
<Objectif>
Une ontologie modélisant des différents concepts qui traitent les caractéristiques des capteurs, les
données des capteurs liées aux mesures et observations du domaine de la météorologie.
</Objectif>
<Niveau de formalité> formel </ Niveau de formalité >
<Termes>
<rdf:Sequence>
<rdf:1_Sensor><rdf:2_Observation><rdf:3_Feature of Interst>
<rdf:4_Phenomenon ><rdf:5_Time><rdf:6_System> .....
</rdf:Sequence>
</Termes>
</rdf:Description>
</rdf:RDF>

```

Figure 3.3 : Un document RDF de spécification de l'ontologie.

## 5.2. Conceptualisation

La conceptualisation mérite une attention particulière car elle détermine le reste de la construction de l'ontologie. L'objectif est d'organiser et de structurer la connaissance, en utilisant des représentations semi-formelles indépendantes des paradigmes de la représentation des connaissances dans lesquels l'ontologie sera formalisée. La conceptualisation a été faite avec une approche hybride (de haut en bas et de bas en haut) en définissant d'abord les concepts les plus importants et en les généralisant autant que possible.

Dans le domaine météorologique, nous avons besoin de connaissances ontologiques pour les observations de capteurs, les observations de données, les situations, et nous avons aussi besoin de connaissances ontologiques pour la modélisation des localisations temporelles et spatiales. Ainsi, en étendant l'ontologie du W3C SSN, nous avons défini des concepts de base et leurs relations pour la surveillance météorologique.

**a. Construction de glossaire de termes**

Ce glossaire contient la définition de tous les termes relatifs au domaine qui seront représentés dans l'ontologie finale. Le tableau ci-dessous fournit une liste détaillée des différents termes utilisés dans l'ontologie :

<b>Le terme</b>	<b>La description</b>
Physical Object	Une entité tangible et visible.
Platform	C'est une entité qui héberge d'autres entités, en particulier des capteurs, des actionneurs, des échantillonneurs et d'autres plates-formes.
System	Est une unité d'abstraction représentant les éléments de l'infrastructure de la détection. Il a des sous-systèmes (System) qui peuvent être Sensor, Wireless Sensor Network.
Wireless Sensor Network	Un RCSF est un réseau sans fil composé de dispositifs autonomes distribués dans l'espace qui utilisent des capteurs (Sensor).
Sensor	Un capteur est un dispositif sensible à un phénomène déterminé et transformant cette grandeur physique en signal (en général électrique) qui peut suivre une méthode de détection (Sensing) pour observer une propriété (Property) , détecte seulement l'entrée du capteur (Sensor Input).
GPS	Le Global Positioning System (GPS) est un Système de navigation et de géolocalisation par satellite.
Thermometer	Est un dispositif (Sensor) permet de transformer l'effet du réchauffement ou du refroidissement sur leurs composants en signal électrique.
Hygrometer	Est un dispositif (Sensor) permet, de mesurer l'humidité relative de l'atmosphère.
Anemometer	Est un dispositif permet d'évaluer la vitesse du vent.
Weathercock	Est un dispositif permet d'indiquer la direction du vent.
Barometer	Est un dispositif permet de mesurer la pression atmosphérique
Pluviometer	Un détecteur d'occurrence de précipitations est un appareil qui détecte la présence d'hydrométéores et en détermine le type (pluie, neige, bruine, etc.) et l'intensité.
Heliograph	Appareil enregistrant le nombre d'heures d'ensoleillement et utilisé pour mesurer la nébulosité
Type Sensor	Les différents types de capteurs qui peuvent être : Temperature Sensor, Humidity Sensor ,Percipitation sensor, Wind Speed Sensor ,Wind Direction Sensor, Pressure Sensor, Cloud Cover Sensor ou Insolation Sensor.
TemperatureSensor	Type caractérise les capteurs de température.
PrecipitationSensor	Type caractérise les capteurs de précipitation.
Pressure Sensor	Type caractérise les capteurs de pression.
HumiditySensor	Type caractérise les capteurs d'humidité.
Wind Speed Sensor	Type caractérise les capteurs de la vitesse du vent.

Wind Direction Sensor	Type caractérise les capteurs de la direction du vent.
Cloud Cover Sensor	Type caractérise les capteurs de la couverture nuageuse.
Event	Fait auquel aboutit une situation
Stimulus	Un événement dans le monde réel qui « déclenche» le capteur. Les propriétés associées au stimulus peuvent être différentes de la propriété ObservableProperty observée. C'est l'événement, pas l'objet, qui déclenche le capteur.
SensorInput	Entrer de capteur de l'environnement physique. Qui peut être la chaleur, l'humidité, la pression ou l'un des nombreux autres phénomènes environnementaux.
Process	C'est une méthode qui peut-être : Sensing, Communicating, Transforming.
Sensing	La méthode de détection est un processus qui aboutit à l'estimation ou le calcul de la valeur d'un phénomène. Décrit le principe qui sous-tend un capteur et la façon dont les observations ont été faites.
Transforming	C'est un processus qui peut décrire la façon dont les transformations des données ont été faites.
Communicating	Est un processus qui peut décrire la façon dont les communications ont été faites. Ses entrées peuvent être des sorties du Sensor Output.
SensorOutput	Est une valeur observée (mesure) produite par un capteur, représentée par une valeur.
Value	Sont les valeurs résultantes d'une observation.
Situation	Peut-être : Observation ou Communication.
Observation	Acte d'effectuer une procédure d'observation pour calculer une valeur d'une propriété d'une caractéristique observée (FeatureOfInterest).
Communication	C'est la situation qui permet de lier les différents nœuds du réseau, et l'acte d'observation.
Property	C'est une qualité (un aspect) observable d'un événement.
ID	C'est un identifiant de capteur (sensor).
Unit Of Measurment	Est un étalon nécessaire pour la mesure d'une grandeur physique.
Feature Of Interest	Les entités dans le monde réel qui sont la cible de la détection.
Medium	Caractérise le sol, l'air, l'eau de mer et de l'atmosphère.
Precipitation	Caractéristique liée aux chutes de neige et la pluie.
Wind	Concerne la vitesse et la direction du vent.
Surface Medium	Concerne l'humidité et la pression de l'air.
Location	Position géographique, déterminer l'emplacement où se situe une chose, un phénomène ou son origine.
Observation Location	La position géographique de l'observation.
Physical Quality	Caractérise les qualités physiques comme l'air et le vent.

Air Quality	Caractérise la qualité de l'air.
AirborneParticles	Caractérise des très fines particules constituées de matières solides ou liquides qui peuvent rester en suspension dans l'air et se propager au vent.
Air Temperature	Caractérise la température de l'air.
AtmosphericPressure	Caractérise la pression atmosphérique.
Rainfall	Désigne tous les météores qui tombent dans une atmosphère et il peut s'agir de solides ou de liquides.
RelativeHumidity	Caractérise le taux d'humidité présent dans l'air (quantité de vapeur d'eau).
Nebulosity	Caractérise la couverture nuageuse.
Wind Quality	Caractérise la qualité du vent la direction et la vitesse.
Wind Direction	Caractérise la direction du vent.
Wind Speed	Caractérise la vitesse du vent.
Phenomena	Représente le phénomène physique qui se produise dans un environnement.
Blizzard	Un phénomène qui désigne le brouillard.
Potentially-Icy	Phénomène potentiel de la formation d'une couche de glace naturelle très mince qui se forme sur un sol au-dessous de 0 °C.
Temperature	Degré de chaleur ou de froid de l'atmosphère.
Winds	Désigne le mouvement horizontal de l'air. Sa mesure comprend deux paramètres : sa direction et sa vitesse.
Rain	Eau qui tombe en gouttes des nuages sur la terre.
Pressure	Force qui agit sur une surface donnée.
Humidity	Désigne en météorologie la quantité de vapeur d'eau contenue dans l'air.
Cloud Cover	Désigne une masse visible constituée initialement d'une grande quantité de gouttelettes d'eau) en suspension dans l'atmosphère.
...	...

**Table 3.1** : Glossaire de termes.

**b. Construction de diagramme de classification de concepts**

Dans cette étape, nous construisons le diagramme de 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.

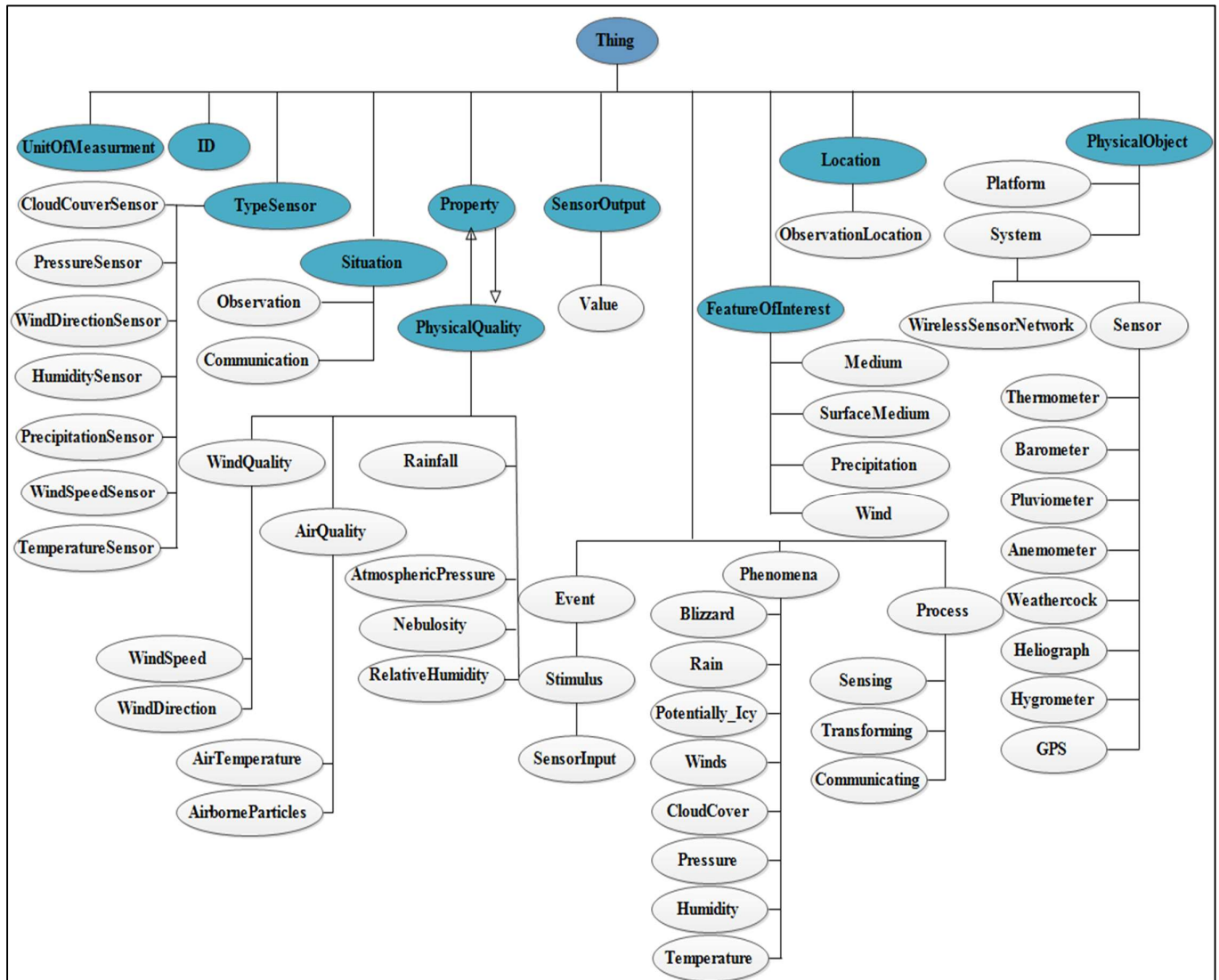


Figure 3.4 : Diagramme de classification de concepts.

c. Construction de diagramme de relations binaires

Nous représentons d'une manière graphique les différentes relations binaires et la hiérarchie entre les classes qui existent entre les divers concepts de notre ontologie.

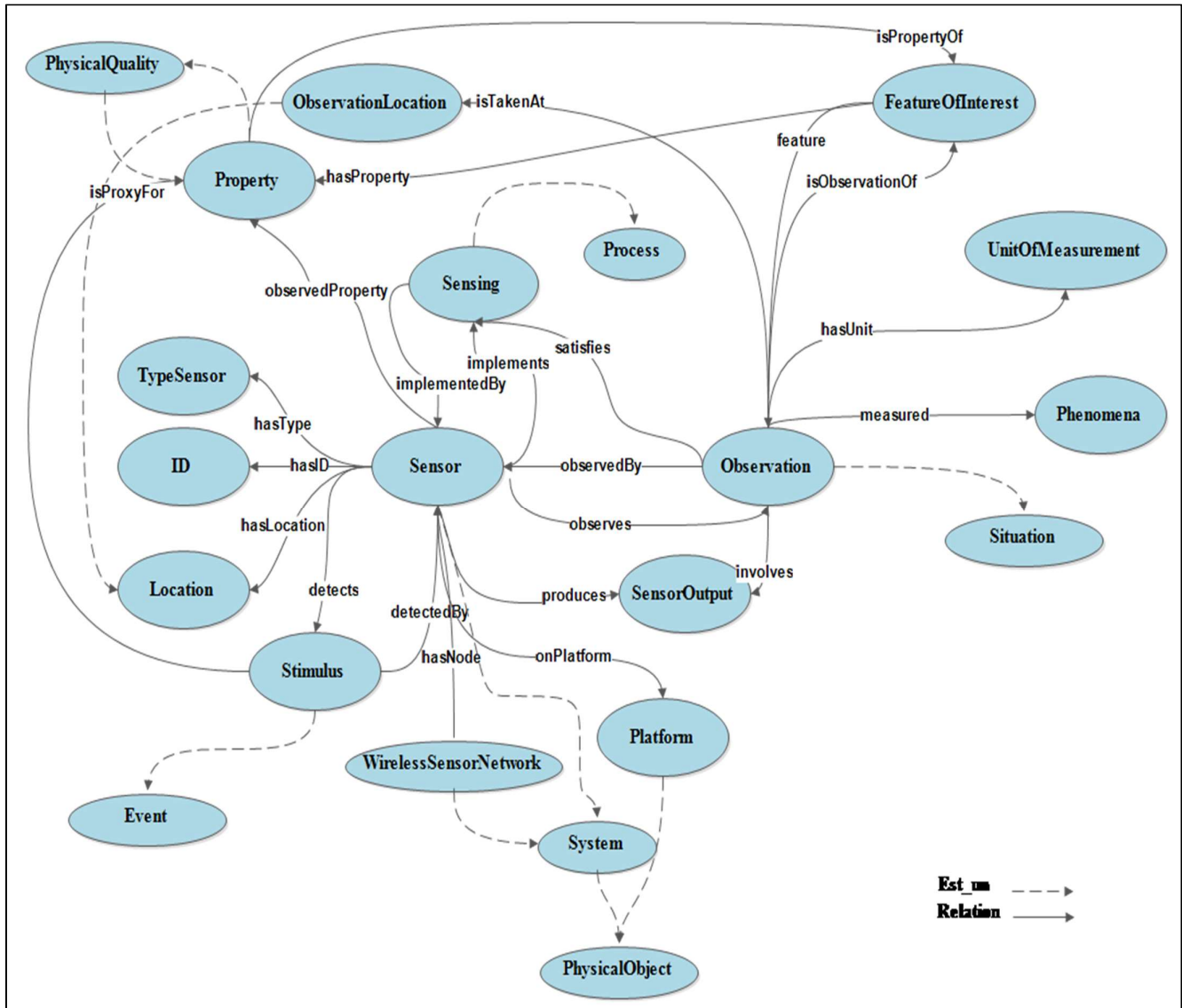


Figure 3.5 : Diagramme de relation binaire.

d. Dictionnaire de concepts

Nom du concept	Concepts Synonymes	Attributs	Instances
Sensor	Capteur Détecteur Sensingdevice	serial_number creation_date memory_capacity battery_capacity computing_capacity	Thermometer_1 Pluviometer_1 Hygrometer_1 Barometer_1 Weathercock_1 Anemometer_1 Heliograph_1
TypeSensor	Type de capteur	-	temperature_sensor percipitation_sensor humidity_sensor pressure_sensor winddirection_sensor windspeed_sensor cloudcover_sensor
Observation	-	hasValue hasAcquisitionTime Described	Obs_1 Obs_2 Obs_3 ...
ID	Identificateur	-	TmpSen001 PerSen001 HumSen001 PreSen001 ...
UnitOfMeasurment	Unité de mesure	-	Celsius kilometer_per_hour Milli_bar Okta ...
Location	Localisation	NameLocation	Loc_1 Loc_2
Platform	Plate-forme	NamePlatform	Station_1 Station_2
PhysicalQuality	Property Propriété	PropertyDescription	Wind_Direction Wind_Speed Air_Temperature Atmospheric_Pressure rainfall Relative_humidity ...
WirelessSensorNetwork	Réseau de capteurs sans fil	WSNdescription WSNidentification	WSN_1 WSN_2
...	...	...	...

Table 3.2 : Le dictionnaire de concepts.

**e. Tableaux des relations binaires**

Nom de la relation	Concept source	Cardinalité source	Concept cible	Cardinalité cible	Relation inverse
detectes	Sensor	(1,1)	Stimulus	(1,n)	detectedBy
Feature	FeatureOfInterest	(1,n)	Observation	(1,1)	isObservationOf
hasID	Sensor	(1,1)	ID	(1,1)	-
hasLocation	Sensor	(1,1)	Location	(1,n)	-
hasType	Sensor	(1,1)	TypeSensor	(1,n)	-
hasUnit	Observation	(1,1)	UnitOfMeasurment	(1,n)	-
hasNode	WirlessSensorNetwork	(1,1)	Sensor	(1,n)	-
implements	Sensor	(1,n)	Sensing	(1,1)	implementedBy
involves	Observation	(1,1)	SensorOutput	(1,1)	-
produces	Sensor	(1,1)	SensorOutput	(1,n)	isProducedBy
isTakenAt	Observation	(1,1)	ObservationLocation	(1,n)	-
measured	Observation	(1,n)	Phenomena	(1,n)	-
observedProperty	Sensor	(1,1)	Property	(1,n)	-
observes	Sensor	(1,n)	Observation	(1,n)	observedBy
onPlatform	Sensor	(1,1)	Platform	(1,n)	attachedSensor
isPropertyOf	Property	(1,1)	FeatureOfInterest	(1,1)	hasProperty
isProxyFor	Stimulus	(1,1)	Property	(1,n)	-
satisfies	Observation	(1,1)	Sensing	(1,1)	-
...	...	...	...	...	...

**Table 3.3** : Table des relations binaires.

**f. Tableaux des attributs**

Attribut	Type	Cardinalité (min/max)	Valeur par défaut	Domain des valeurs
serial_number	Integer	(1,1)	-	-
creation_date	DateTime	(1,1)	-	YYYY-MM-DD Thh:mm:ss
memory_capacity	String	(1,1)	-	-
battery_capacity	String	(1,1)	-	-
computing_capacity	String	(1,1)	-	-
hasValue	Real	(1,1)	-	-
hasAcquisitionTime	DateTime	(1,1)	-	YYYY-MM-DD Thh:mm:ss
Described	String	(1,1)	-	-
...	...	...	...	...

**Table 3.4** : La table des Attributs.

**g. Tableaux des axiomes logiques**

<b>Le concept</b>	<b>La description</b>	<b>L'expression logique</b>
Type Sensor	Un type de capteur (sensor) peut être : HumiditySensor, PercipitationSensor, Pressure Sensor, TemperatureSensor, Wind Direction Sensor, Wind Speed Sensor ou Cloud Couver Sensor.	$\forall (X), \text{TypeSensor}(X) \Rightarrow \text{TemperatureSensor}(X) \vee \text{HumiditySensor}(X) \vee \text{PercipitationSensor}(X) \vee \text{pressureSensor}(X) \vee \text{WindDirectionSensor}(X) \vee \text{WindSpeedSensor}(X) \vee \text{CloudCoverSensor}(X).$
Process	Un processus est une methode qui peut être : Communicating, Transforming ou Sensing.	$\forall (X), \text{Process}(X) \Rightarrow \text{Communicating}(X) \vee \text{Transforming}(X) \vee \text{Sensing}(X).$
PhysicalObject	Un objet physique peut être : Platform ou System.	$\forall (X), \text{PhysicalObject}(X) \Rightarrow \text{Platform}(X) \vee \text{System}(X).$
System	Un système peut être : Sensor (capteur), device ou Wireless Sensor Network (réseau de capteur sans fil).	$\forall (X), \text{System}(X) \Rightarrow \text{Sensor}(X) \vee \text{Device}(X) \vee \text{WirelessSensorNetwork}(X).$
Phenomena	Un phénomène peut être : Blizzard, Potentialy-icy, Winds, Temperature, CloudCouver, Rain, Pressure ou Humidity.	$\forall (X), \text{Phenomena}(X) \Rightarrow \text{Blizzard}(X) \vee \text{Potential-icy}(X) \vee \text{Winds}(X) \vee \text{Temperature}(X) \vee \text{CloudCover}(X) \vee \text{Rain}(X) \vee \text{Pressure}(X) \vee \text{Humidity}(X).$
PhysicalQuality	Une qualité physique peut être : Air Quality, Wind Quality, Relative Humidity, Atmospheric Pressure, Rainfall ou Nebulosity.	$\forall (X), \text{PhysicalQuality}(X) \Rightarrow \text{AirQuality}(X) \vee \text{WindQuality}(X) \vee \text{Rainfall}(X) \vee \text{AtmosphericPressure}(X) \vee \text{RelativeHumidity}(X) \vee \text{Nebulosity}(X).$
Sensor	Un capteur (Sensor) peut être : Thermometer, Barometer, Anemometer, Pluviometer, Weathercock, Hygrometer, Heliograph ou GPS.	$\forall (X), \text{Sensor}(X) \Rightarrow \text{Thermometer}(X) \vee \text{Hygrometer}(X) \vee \text{Pluviometer}(X) \vee \text{Barometer}(X) \vee \text{Weathercock}(X) \vee \text{Heliograph}(X) \vee \text{Anemometer}(X) \vee \text{GPS}(X).$

**Table 3.5:** La table des axiomes logique.

**h. Tableaux des instances**

L'instance	L'attribut	La valeur
Thermomater_1	serial_number creation_date computing_capacity memory_capacity battery_capacity	140428 2014-04-28T15:30:40 High High High
Obs_1	hasValue hasAcquisitionTime Described	24.0 2018-05-07T13:20:32 Température modérée
Station_1	NamePlatform	Station météorologique
...	...	...

**Table 3.6 :** La table des instances.

**i. Construction de la table des assertions**

Relation	Instance source	Instance cible
hasID	Thermometer_1	TmpSen001
hasType	Thermometer_1	temperature_sensor
hasLocation	Thermometer_1	Loc_1
onPlatform	Thermometer_1	Station_1
ObservedProperty	Thermometer_1	Air_Temperature
Observes	Thermometer_1	Obs_1
hasID	Hygrometer_1	PreSen001
hasType	Hygrometer_1	pressure_sensor
hasLocation	Hygrometer_1	Loc_1
onPlatform	Hygrometer_1	Station_1
ObservedProperty	Hygrometer_1	Atmospheric_Pressure
Observes	Hygrometer_1	Obs_2
hasUnit	Obs_1	Celsius
isTakenAt	Obs_1	Loc_1
hasUnit	Obs_2	Milli_bar
isTakenAt	Obs_2	Loc_1
hasNode	Wsn_1	Thermometer_1 Barometer_1 Hygrometer_1 Heliograph_1 Pluviometer_1 Anemometer_1 Weathercock_1
...	...	...

**Table 3.7:** La table des assertions.

### 5.3. Formalisation

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

#### 5.3.1. Construction de TBOX

Nous définissons ici les concepts et les rôles relatifs à notre domaine, en utilisant les constructeurs fournis par les logiques de descriptions pour donner des descriptions structurées aux concepts et rôles.

##### a. Définition des concepts

Concept	Définition	Relation de subsumption
Sensor	$\text{GPS} \cup \text{Thermometer} \cup \text{Pluviometer} \cup \text{Barometer} \cup \text{Hygrometer} \cup \text{Anemometer} \cup \text{Weathercock} \cup \text{Heliograph} \cap (\exists \text{serial\_number}, \text{Integer}) \cap (\exists \text{battery\_capacity}, \text{string}) \cap (\exists \text{memory\_capacity}, \text{String}) \cap (\exists \text{computing\_capacity}, \text{String}) \cap (\exists \text{creation\_date}, \text{DateTime}) \cap (\forall \text{onPlatform}, \text{Platform}) \cap (\forall \text{observes}, \text{Observation}) \cap (\forall \text{observedProperty}, \text{Property}) \cap (\forall \text{hasID}, \text{ID}) \cap (\forall \text{hasLocation}, \text{Location}) \cap (\forall \text{hasType}, \text{TypeSensor}) \cap (\forall \text{detects}, \text{Stimulus}) \cap (\forall \text{implements}, \text{Sensing}) \cap (\forall \text{produces}, \text{SensorOutput})$ .	$\text{Sensor} \subseteq \text{System}$
TypeSensor	$\text{PressureSensor} \cup \text{PrecipitationSensor} \cup \text{temperatureSensor} \cup \text{HumiditySensor} \cup \text{WindDirectionSensor} \cup \text{WindSpeedSensor} \cup \text{CloudCouverSensor}$ .	$\text{TypeSensor} \subseteq \text{Thing}$
Observation	$\text{Situation} \cap (\forall \text{hasUnit}, \text{UnitOfMeasurment}) \cap (\forall \text{ObservedBy}, \text{Sensor}) \cap (\forall \text{isTakenAt}, \text{ObservationLocation}) \cap (\forall \text{satisfies}, \text{Sensing}) \cap (\forall \text{involves}, \text{SensorOutput}) \cap (\forall \text{isObservationOf}, \text{FeatureOfInterest}) \cap (\exists \text{hasValue}, \text{Real}) \cap (\exists \text{hasAcquisitionTime}, \text{DateTime}) \cap (\exists \text{Described}, \text{String})$ .	$\text{Observation} \subseteq \text{Setuation}$
Process	$\text{Communication} \cup \text{Transforming} \cup \text{Sensing}$	$\text{Process} \subseteq \text{Thing}$
Sensing	$\text{Process} \cap (\forall \text{implementedBy}, \text{Sensor})$ .	$\text{Sensing} \subseteq \text{Process}$
FeatureOfInterest	$\text{Meduim} \cup \text{SurfaceMeduim} \cup \text{Precipitation} \cup \text{Wind} \cap (\forall \text{feature}, \text{Observation}) \cap (\forall \text{hasProperty}, \text{Property})$ .	$\text{FeatureOfInterest} \subseteq \text{Thing}$

WirlessSensorNetwork	System $\cap$ ( $\forall$ hasNode, Sensor) $\cap$ ( $\exists$ WSNdescription, String) $\cap$ ( $\exists$ WSNidentification, String).	WirlessSensorNetwork $\subseteq$ system
Phenomena	Blizzard $\cup$ Potentialy-icy $\cup$ Temperature $\cup$ Winds $\cup$ Pressure $\cup$ Humidity $\cup$ Rain $\cup$ CloudCover.	Phenomena $\subseteq$ Thing
Platform	PhysicalObject $\cap$ ( $\forall$ attachedSensor, Sensor) $\cap$ ( $\exists$ Nameplatform, string).	Platform $\subseteq$ PhysicalObject
...	...	...

**Table 3.8 :** Définition des concept (dans TBOX).

**b. Définition des rôles**

Rôle	Couple (cible, source)	Rôle inverse
detects	(Sensor, Stimulus)	detectedBy
feature	(FeatureOfInterest, Observation)	isObservationOf
hasID	(Sensor, ID)	-
hasLocation	(Sensor, Location)	-
hasType	(Sensor, TypeSensor)	-
hasUnit	(Observation, UnitOfMeasurment)	-
implements	(Sensor, Sensing)	implementedBy
involves	(Observation, SensorOutput)	-
produces	(Sensor, SensorOutput)	isProducedBy
isTakenAt	(Observation, ObservationLocation)	-
...	...	...

**Table 3.9 :** Définition des rôles (dans TBOX).

**5.3.2. Construction de ABOX**

Le langage assertionnel est dédié à la description des faits, en spécifiant les individus (avec leurs classes) et les relations entre eux de la manière suivante : « A : C » Pour dire que A est une instance de la classe C ; Par exemple « Obs\_1 : Observation ».

« (A1, A2) : R » Pour dire que les deux individus A1 et A2 sont reliés par la relation R ;

Par exemple : « (Thermometer\_1, Obs\_1) : observes ».

**a. Partie assertionelle des concepts**

Concepts	Description
Sensor	Thermometer_1 :Sensor Barometer_1 :Sensor ; ...
Observation	Obs_1 :Observation Obs_2 :Observation ; ...
UnitOfMeasurment	Celsius : UnitOfMeasurment Okta : UnitOfMeasurment ; ...
...	...

**Table 3.10 :** Description assertionelle des concepts.

**b. Partie assertive des rôles**

<b>Relation</b>	<b>Description</b>
hasType	(Thermometer_1, temperature_sensor) :hasType (Barometer_1, pressure_sensor) :hasType ; ...
observes	(Thermometer_1, Obs_1): observes (Barometer_1, Obs_2): observes ; ...
hasUnit	(Obs_1, Celsius) :hasUnit (Obs_8, Okta) :hasUnit ;...
...	...

**Table 3.11** : Description assertive des rôles.

**6. Conclusion**

Dans ce chapitre on a présenté les différentes phases par lesquelles on a passé pour réaliser la conception de l'ontologie de notre projet, et on a présenté le résultat de la conception sous forme d'une liste de concepts, une liste de relations et une autre des attributs en plus d'une représentation hiérarchique.

La prochaine étape consiste à rendre cette ontologie opérationnelle c'est-à-dire exploitable par un ordinateur, c'est ce qu'on va présenter dans le prochain chapitre, en plus du détail de réalisation de l'application de ce projet.

# CHAPITRE 4

## IMPLEMENTATION ET ANNOTATION SEMANTIQUE

*“Le savoir n’est pas difficile, seule sa mise en pratique l’est.”*

Proverbe chinois

Après avoir conçu le modèle d’ontologie dans le chapitre conception, nous passons maintenant à l’implémentation de l’ontologie qui est une tâche complexe même pour une petite ontologie, celle-ci va prendre plusieurs lignes de code, un grand effort et du temps notamment si cette ontologie est codée directement par le développeur en langage d’ontologie sans l’utilisation d’un outil. A cette égard les éditeurs d’ontologie sont conçus pour générer automatiquement la structure de l’ontologie créée en langage de spécification. Notre choix porte sur l’éditeur Protégé 5.2 en utilisant OWL qu’est un langage de codification qui a des fonctionnalités sémantiques riches.

Ce chapitre contient deux parties la première où nous allons présenter le travail d’implémentation que nous avons fait, ainsi l’interrogation des triplets RDF avec SPARQL et les règles de raisonnement avec SWRL. La deuxième partie est consacrée à l’annotation sémantique pour donner une signification aux données brutes.

### 1. Présentation de l’outil Protégé 5.2

Pour l’implémentation de notre ontologie nous avons utilisé l’éditeur Protégé 5.2 qui est une version améliorée de protégé qui contient diverses améliorations et correctifs. Il a été développé par le Stanford Medical Informatics au sein de l’Université de Stanford, qui fournit un environnement graphique permettant l’édition, la visualisation et le contrôle (vérification des contraintes) d’ontologies.

L’interface de Protégé complet ainsi que l’architecture logicielle ouverte et extensible permettant l’insertion de plug-ins, ont grandement participé au succès de Protégé.

Ces plug-ins pouvant apporter de nouvelles fonctionnalités entre autres OntoGraf permet de gérer des représentations sous forme graphique, SPARQL Query qui est dédié pour l'exécution des requêtes SPARQL, et un plugin dédié à OWL.

## 2. Etapes de création de notre ontologie

Après le lancement de Protégé nous obtiendrons une interface divisée en (Figure 4.1) qui est divisée en vues qui dépendent des onglets (Active Ontology, Entities, Classes, ...).

Dans l'onglet Active Ontologie nous définissons l'URI de notre ontologie dans Ontology IRI, dans le menu système File/Save nous choisissons le format de sauvegarde de l'ontologie, dans notre cas il s'agit de OWL/XML ainsi l'endroit où sera enregistré notre ontologie.

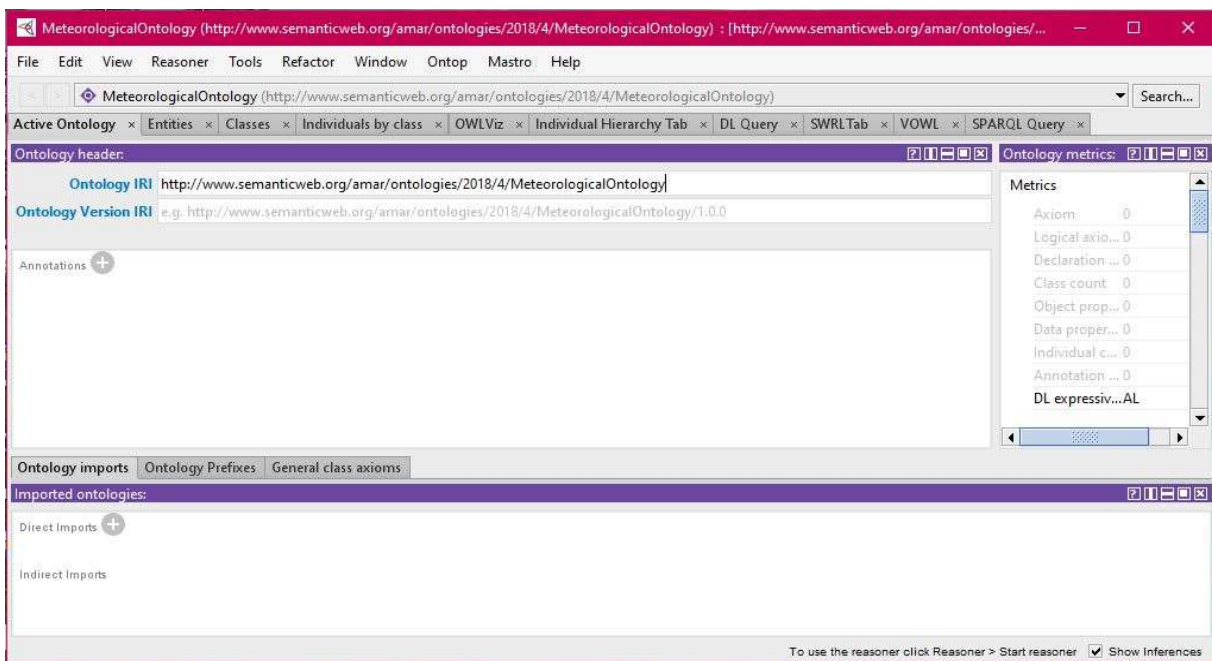


Figure4.1 : l'interface de protégé 5.2.

### 2.1. Définition des classes

Dans le sous-onglet **Classes** de l'onglet **Entities** (Figure 4.2). En cliquant sur owl:Thing dans la vue **Class hierarchy** qui est la racine de toutes les classes que nous allons créer. En appuyant sur le premier bouton en haut à gauche de cette vue, un dialogue apparaît qui permet de créer une classe qui sera une sous-classe de la classe sélectionnée, le deuxième bouton permet de créer une classe sœur de la classe sélectionnée et le troisième bouton détruit la classe sélectionnée ainsi que ses sous-classes. La partie droite de l'écran permet de voir des informations sur les classes et d'en changer les caractéristiques. Pour indiquer que les classes

sont disjointes nous choisissons une classe et nous indiquons l'autre dans **Disjoint With** dans la vue **Description**.

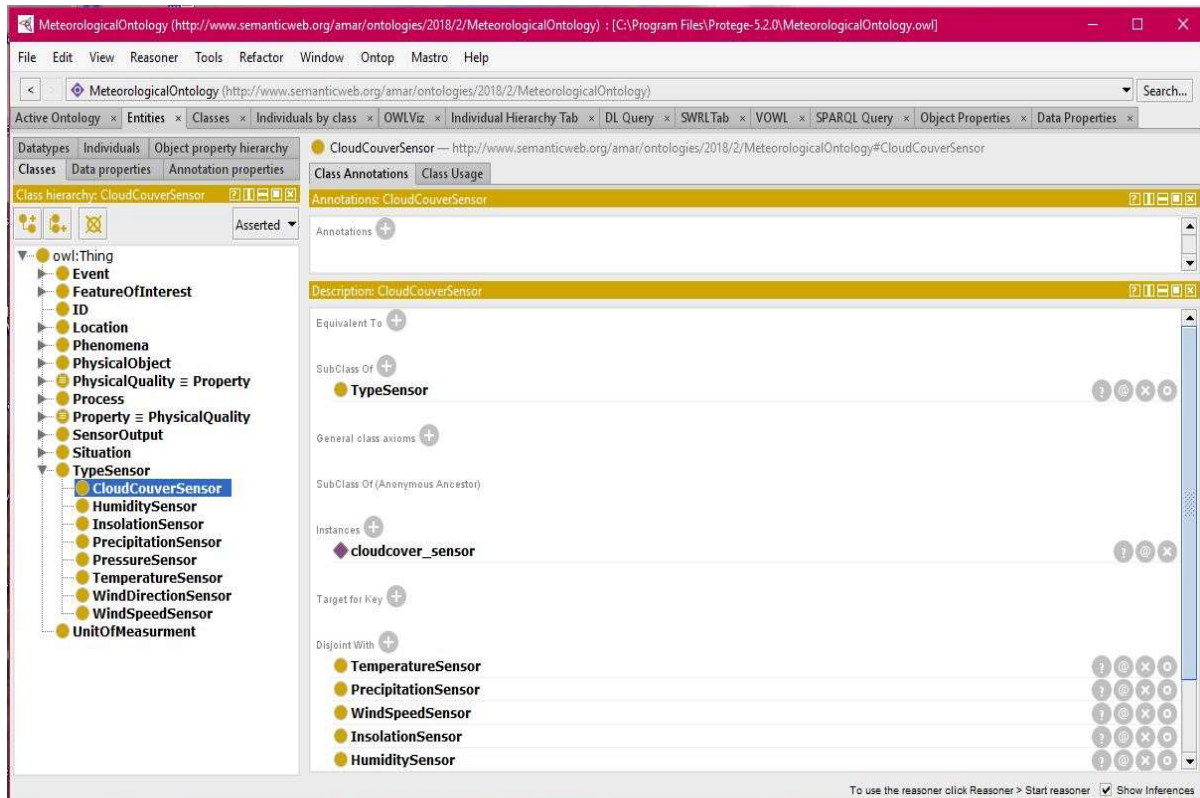


Figure 4.2 : La hiérarchie de classes.

## 2.2. Définition des propriétés

Les propriétés des classes peuvent être de deux types différents attribut ou rôle. Il s'agit des attributs « datatypeProperty » des concepts et des rôles « objectProperty » qui relient les concepts.

### 2.2.1. Object property

Dans l'onglet **Object Properties** (Figure 4.3) nous allons créer la hiérarchie des propriétés et nous spécifions les caractéristiques (fonctionnalité, fonctionnalité inverse, symétrie, transitivité) et les autres propriétés équivalentes et différentes dans la vue **Characterist**. leur domaine (rdfs : domain), co-domaine (rdfs : range) et la relation inverse qui lui correspond (owl : inverseOf) dans la vue **Description**.

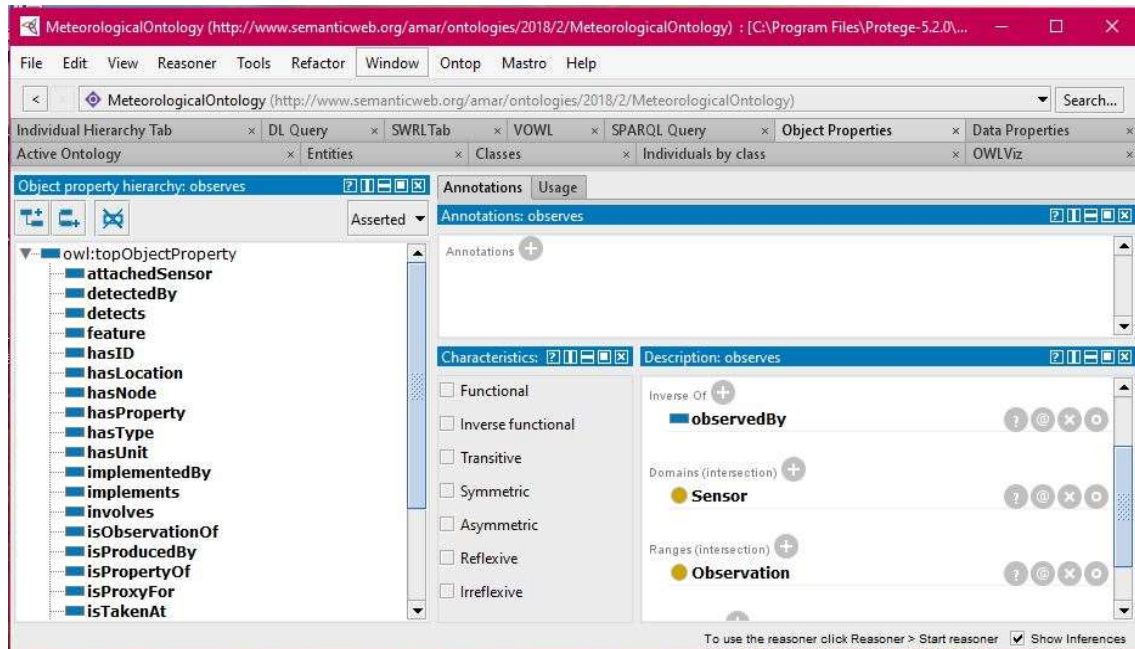


Figure 4.3 : L'onglet Object Properties.

### 2.2.2. Data type property

Dans l'onglet Data Properties (Figure 4.4) en cliquant sur le bouton de l'ajout d'une propriété dans la vue Data property hierarchy pour ajouter les propriétés et dans la vue **Description** nous spécifions leur domaine (rdfs : domain), co-domaine (rdfs : range) qu'il s'agit d'un type de valeur.

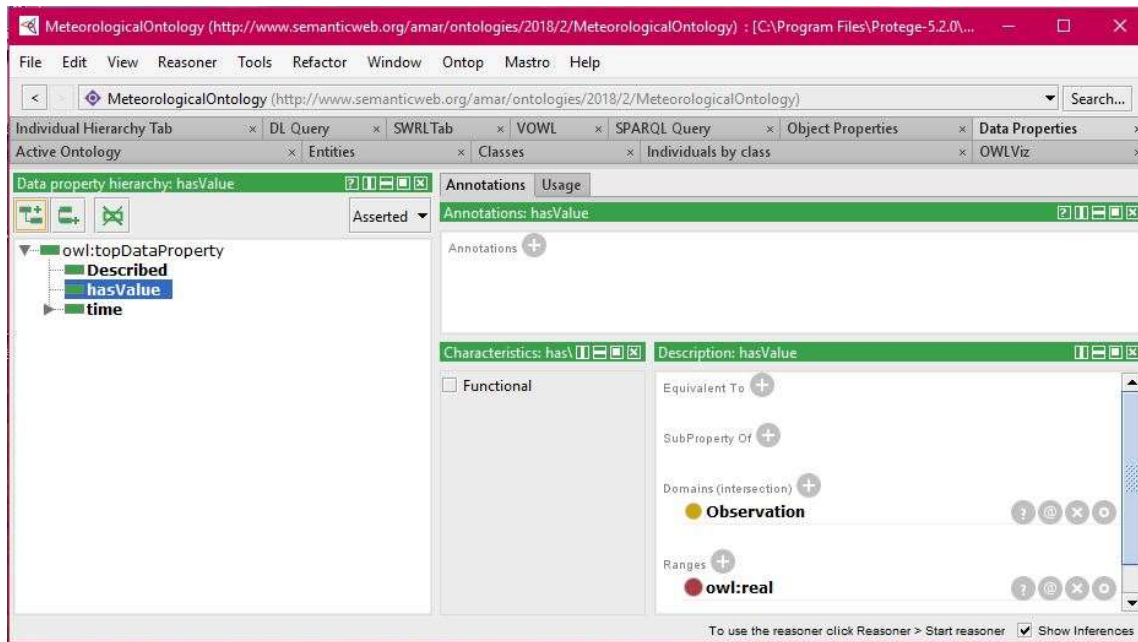


Figure 4.4 : L'onglet Data Properties.

### 2.3. Ajout des individus

Pour ajouter des instances des classes, dans l'onglet **Individuals by class** (Figure 4.5) nous sélectionnons la classe à instancier dans la vue en haut à gauche et nous ajoutons un individu comme membre dans la vue **Instances**. Dans la vue **Property assertions** nous complétons les champs soit par les valeurs des attributs « **Data property assertions** », soit par les noms des instances « **Object property assertions** » avec lesquelles cette instance est reliée par un rôle.

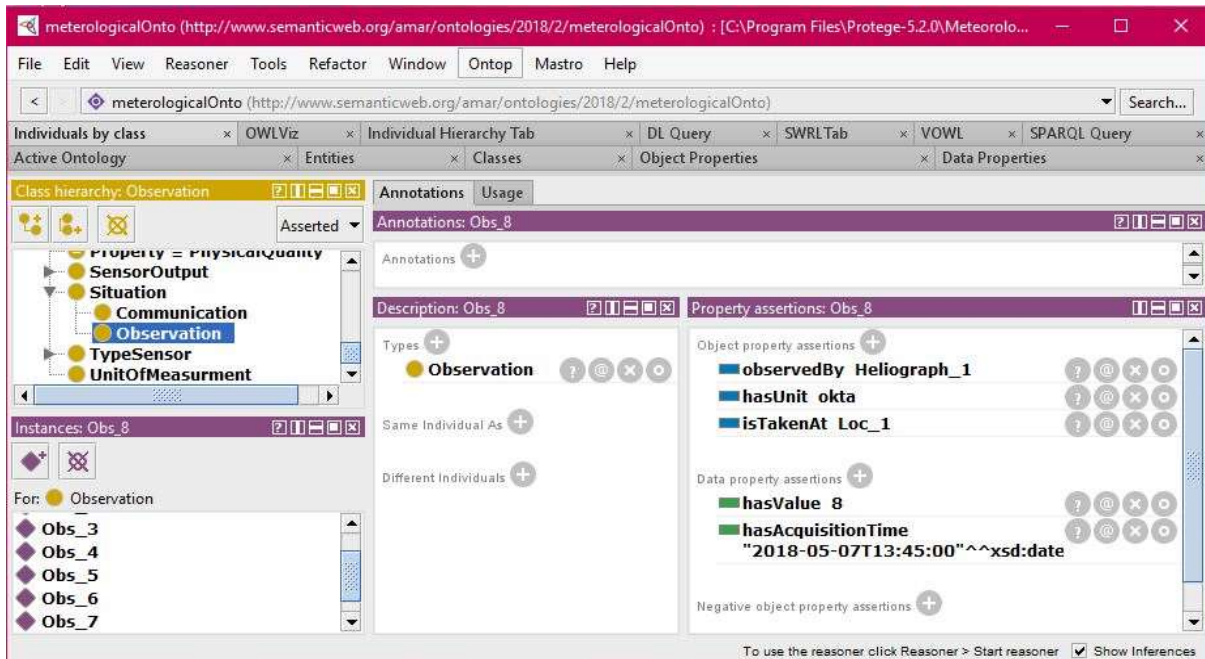


Figure 4.5: L'onglet Individuals by class.

### 3. Annotation sémantique des capteurs et des observations avec RDF

Pour enrichir sémantiquement des données nous devons les annoter. La deuxième partie de notre approche consiste à proposer un ensemble d'annotations sémantiques qui permet d'associer des méta-données aux capteurs (propriétés des capteurs et localisation... etc.) et les flux des capteurs (mesures et observations obtenue à partir des capteurs), en prenant par considération les deux critères spatial et temporel. Pour l'annotation des capteurs et des observations nous utilisons le langage RDF, selon notre modèle ontologique proposé les annotations sont stockées dans un dépôt de triplets. Ce dernier est utilisé pour le stockage et la récupération de triplets via des requêtes sémantiques ainsi de les partager et de les réutiliser entre les systèmes et les utilisateurs.

Les capteurs et leurs observations sont annotés au format RDF comme suit :

### Thermometer\_1:

#### Annotation RDF:

```
<ex: Thermometer_1 rdf:type ex:Sensor>
<ex: Thermometer_1 ex:observedProperty ex:Air_Temperature>
<ex: Thermometer_1 ex:onPlatform ex:Station_1>
<ex: Thermometer_1 ex:hasLocation ex:Loc_1>
<ex: Thermometer_1 ex:hasID ex:TmpSen001>
<ex: Thermometer_1 ex:hasType ex:temperature_sensor>
<ex: Thermometer_1 ex:observes ex:Obs_1>
```

#### Graphe RDF:

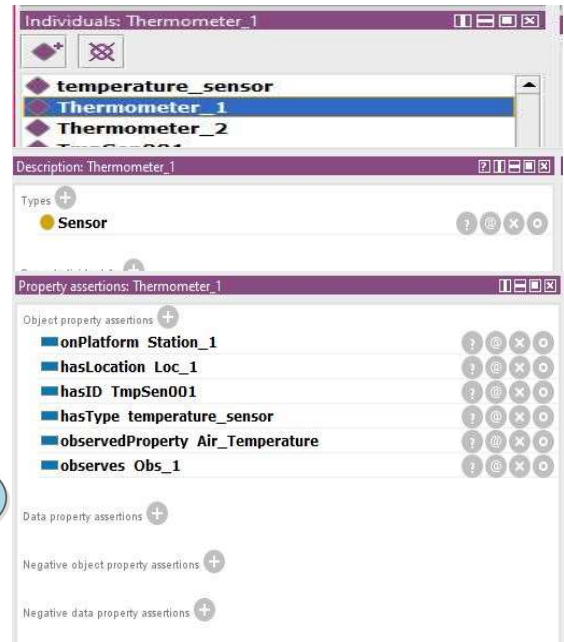
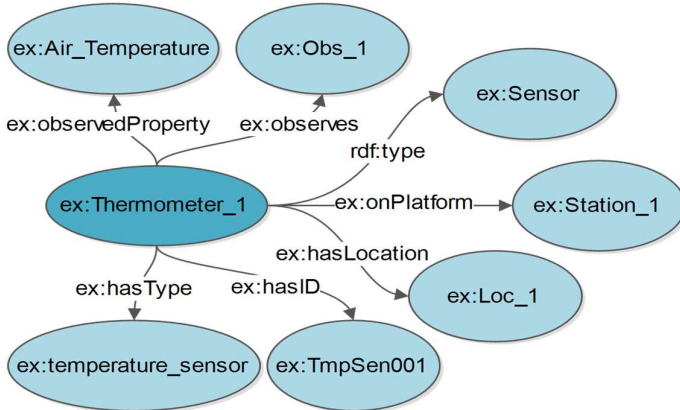


Figure 4.6: Graphe RDF et description sémantique de « Thermometer\_1 ».

### Anemometer\_1:

#### Annotation RDF:

```
<ex: Anemometer_1 rdf:type ex:Sensor>
<ex: Anemometer_1 ex:observedProperty ex:wind_speed>
<ex: Anemometer_1 ex:onPlatform ex:Station_1>
<ex: Anemometer_1 ex:hasLocation ex:Loc_1>
<ex: Anemometer_1 ex:hasID ex:WindSSen001>
<ex: Anemometer_1 ex:hasType ex:windspeed_sensor>
<ex: Anemometer_1 ex:observes ex:Obs_2>
```

#### Graphe RDF:

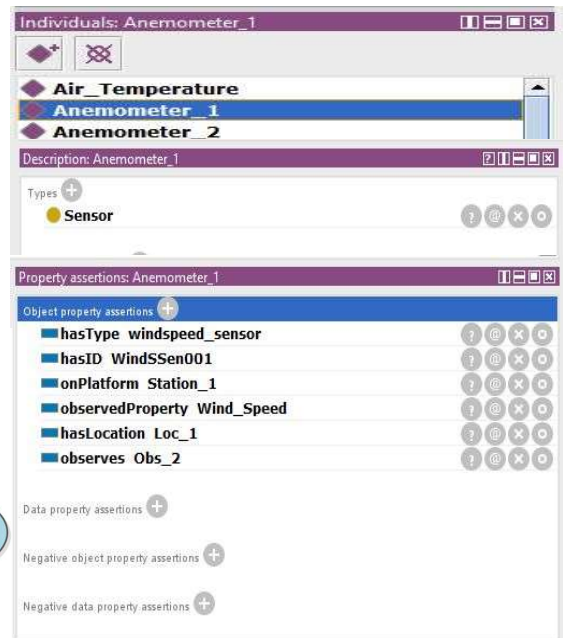
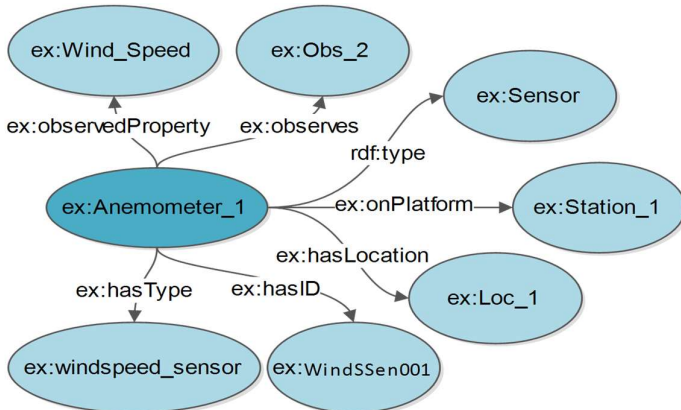


Figure 4.7 : Graphe RDF et description sémantique de « Anemometer\_1 ».

### Barometer\_1:

#### Annotation RDF:

```
<ex:Barometer_1 rdf:type ex:Sensor>
<ex:Barometer_1 ex:observedProperty ex:Atmospheric_Pressure>
<ex:Barometer_1 ex:onPlatform ex:Station_1>
<ex:Barometer_1 ex:hasLocation ex:Loc_1>
<ex:Barometer_1 ex:hasID ex:PreSen001>
<ex:Barometer_1 ex:hasType ex:pressure_sensor>
<ex:Barometer_1 ex:observes ex:Obs_3>
```

#### Graphe RDF:

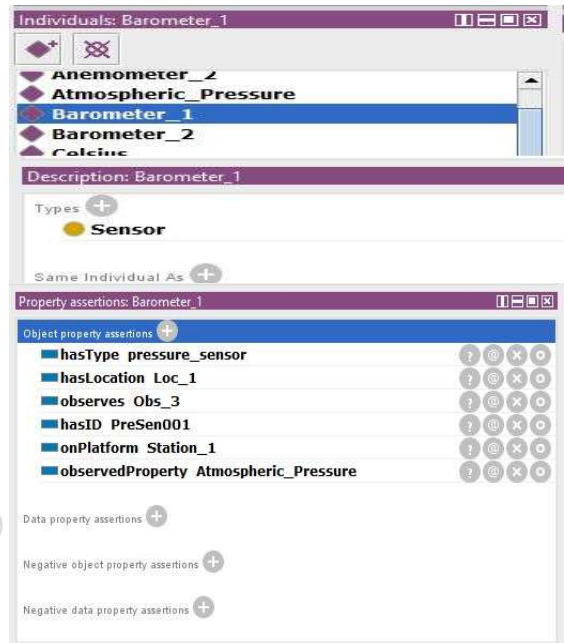
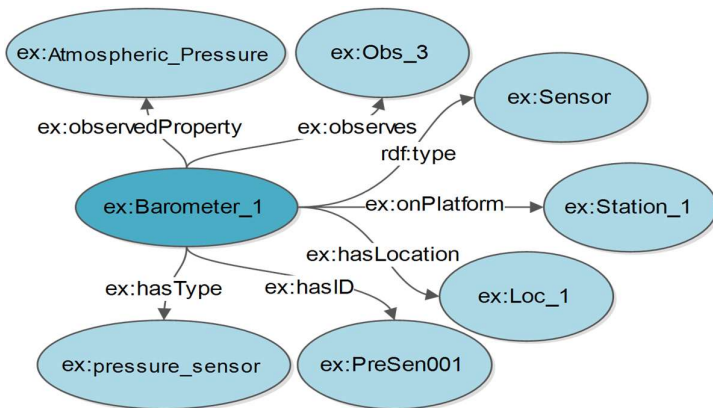


Figure 4.8: Graphe RDF et description sémantique de « Barometer\_1 ».

### Obs\_1:

#### Annotation RDF:

```
<ex:Obs_1 rdf:type ex:Observation>
<ex:Obs_1 ex:isTakenAt ex:Loc_1>
<ex:Obs_1 ex:hasValue "24">
<ex:Obs_1 ex:hasAcquisitionTime "2018-05-07T13:04:00">
<ex:Obs_1 ex:hasUnit ex:Celsius">
<ex:Obs_1 ex:observedBy ex:Thermometer_1>
<ex:Obs_1 ex:measured ex:temperature>
```

#### Graphe RDF:

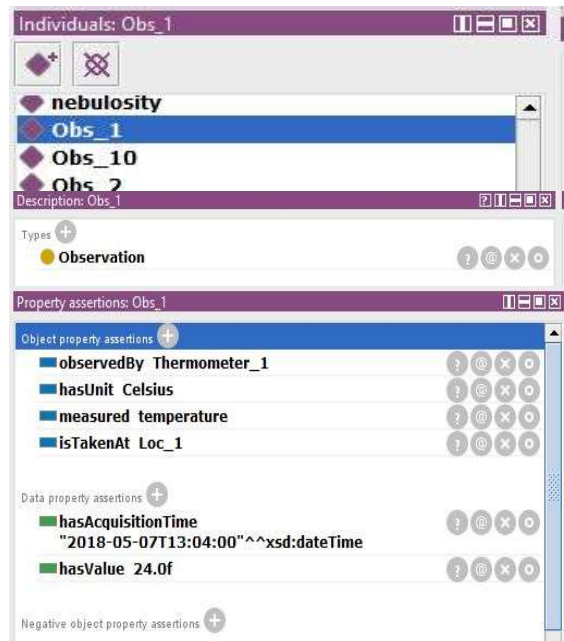
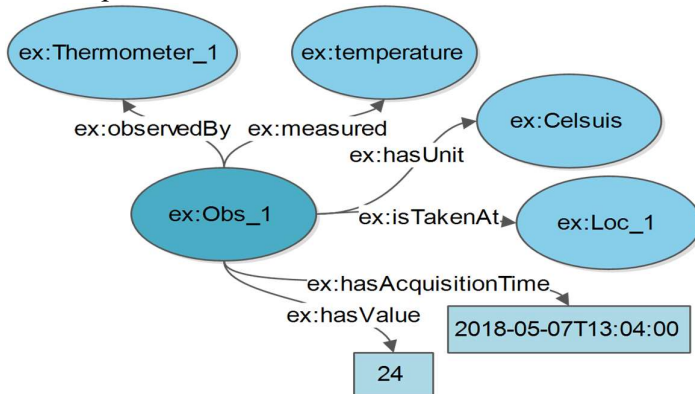


Figure 4.9 : Graphe RDF et description sémantique de « Obs\_1 ».

**Obs\_8:**

Annotation RDF:

```
<ex:Obs_8 rdf:type ex:Observation>
<ex:Obs_8 ex:isTakenAt ex:Loc_2>
<ex:Obs_8 ex:hasValue "0">
<ex:Obs_8 ex:hasAcquisitionTime "2018-05-09T22:04:00">
<ex:Obs_8 ex:hasUnit ex:okta>
<ex:Obs_8 ex:observedBy ex:Heliograph_1>
<ex:Obs_8 ex:measured ex:cloudcover>
```

Graphe RDF :

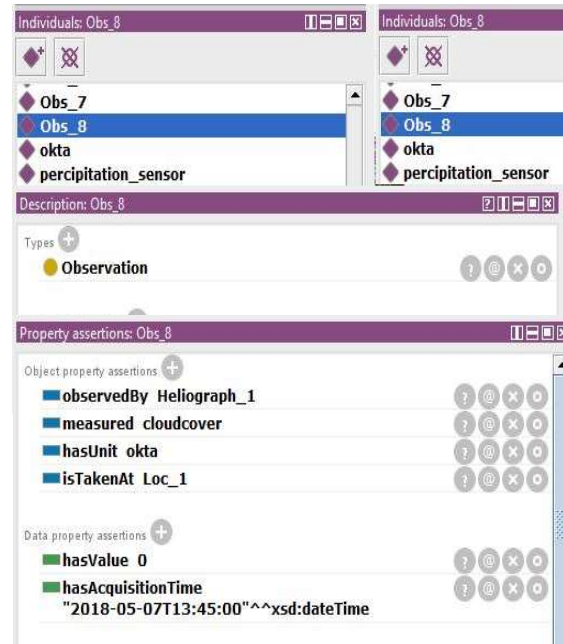
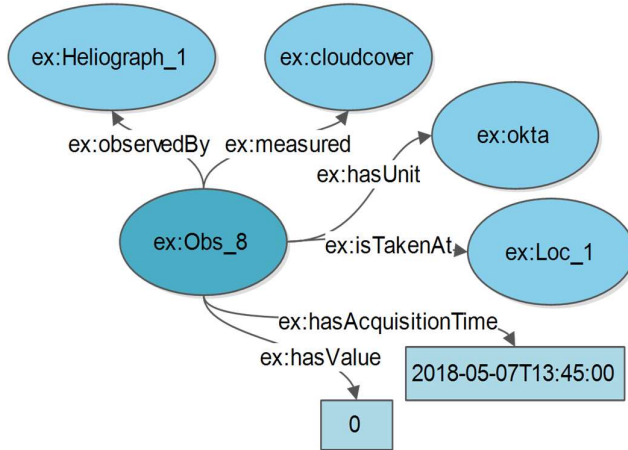


Figure 4.10: Graphe RDF et description sémantique de « Obs\_8 ».

**Obs\_10:**

Annotation RDF:

```
<ex:Obs_10 rdf:type ex:Observation>
<ex:Obs_10 ex:isTakenAt ex:Loc_3>
<ex:Obs_10 ex:hasValue "-5">
<ex:Obs_10 ex:hasAcquisitionTime " 2018-03-09T22:04:00">
<ex:Obs_10 ex:observedBy ex:Pluviometer_2>
<ex:Obs_10 ex:observedBy ex:Thermometer_2>
<ex:Obs_10 ex:measured ex:Temperature>
<ex:Obs_10 ex:measured ex:rain>
```

Graphe RDF:

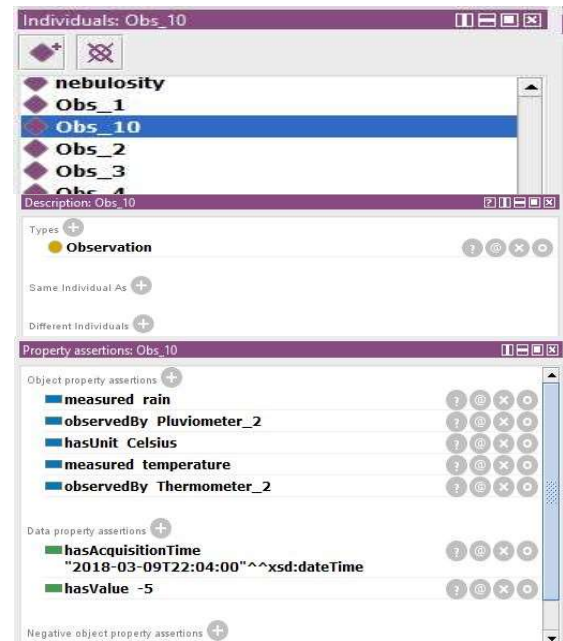
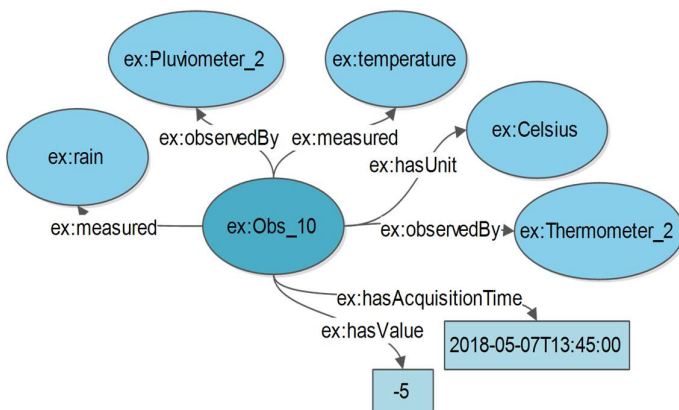


Figure 4.11: Graphe RDF et description sémantique de « Obs\_10 ».

## 4. Interrogation de l'ontologie avec SPARQL

SPARQL est l'équivalent de SQL car comme en SQL, on accède aux données d'une base de données alors qu'avec SPARQL, on accède aux données du Web des données. Cela signifie qu'en théorie, nous pouvons accéder à toutes les données du Web avec ce standard.

Grâce au SPARQL Query plug-in de Protégé qui fournit un support pour l'écriture et l'exécution de requêtes SPARQL, nous interrogeons notre ontologie.

### 4.1. Ajout de SPARQL Query

Dans le menu système **Window/Tabs** nous choisissons SPARQL Query pour ajouter son onglet (Figure 4.12).

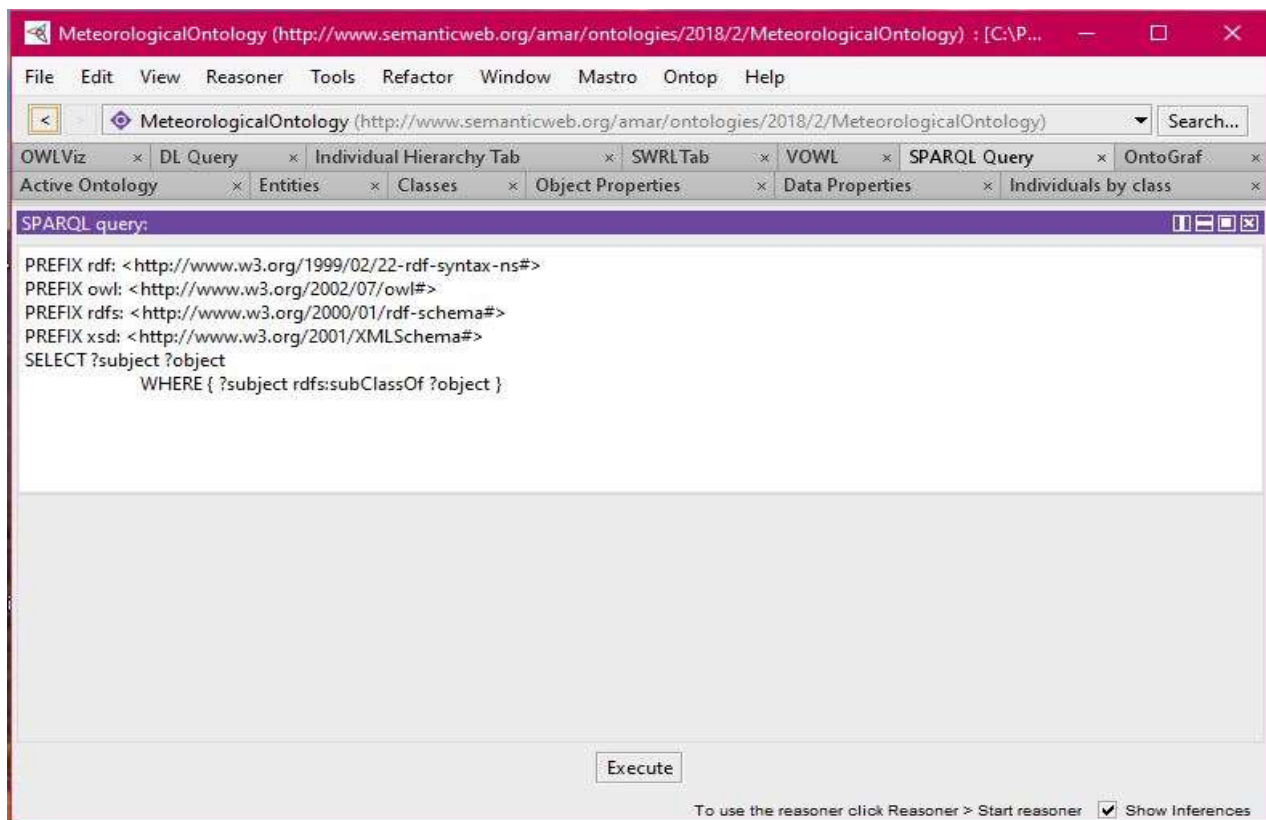


Figure 4.12 : L'onglet SPARQL Query.

### 4.2. Exécution de requêtes

Dans la partie d'édition de requêtes nous écrivons la requête à exécuter, en cliquant sur le boutons **Execute** le résultat va apparaître juste en bas. Dans la Figure 4.13 nous montrons une requête SPARQL simple qui demande des ID et des emplacements de capteurs qui mesurent la température de l'air.

SPARQL query:

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ex: <http://www.semanticweb.org/amar/ontologies/2018/2/meteorologicalOnto#>
SELECT ?SENSOR ?ID ?LOCATION
  WHERE {
    ?SENSOR rdf:type ex:Sensor.
    ?SENSOR ex:hasType ex:temperature_sensor.
    ?ID rdf:type ex:ID.
    ?SENSOR ex:hasID ?ID.
    ?LOCATION rdf:type ex:Location.
    ?SENSOR ex:hasLocation ?LOCATION.
  }
    
```

SENSOR	ID	LOCATION
Thermometer_2	TmpSen002	Loc_2
Thermometer_1	TmpSen001	Loc_1

Figure 4.13 : Résultat d'exécution de la requête 1.

Dans la Figure 4.14 suivante une requête SPARQL qui demande les valeurs, le temp, le capteur qui la observer et le phénomène mesuré des observations qui se localise à Loc\_1.

SPARQL query:

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ex: <http://www.semanticweb.org/amar/ontologies/2018/2/meteorologicalOnto#>
SELECT ?OBS ?VALUE ?TIME ?SENSOR ?PHENOMENA
  WHERE {
    ?OBS rdf:type ex:Observation.
    ?OBS ex:isTakenAt ex:Loc_1.
    ?OBS ex:hasValue ?VALUE.
    ?OBS ex:hasAcquisitionTime ?TIME.
    ?SENSOR rdf:type ex:Sensor.
    ?OBS ex:observedBy ?SENSOR.
    ?PHENOMENA rdf:type ex:Phenomena.
    ?OBS ex:measured ?PHENOMENA.
  }
    
```

OBS	VALUE	TIME	SENSOR	PHENOMENA
Obs_5	"0.5"^^<http://www	"2018-05-07T13:32:08"^^<htt	Pluviometer_1	rain
Obs_1	"24.0"^^<http://ww	"2018-05-07T13:04:00"^^<htt	Thermometer_1	temperature
Obs_2	"6.0"^^<http://www	"2018-05-07T13:27:00"^^<htt	Anemometer_1	winds
Obs_3	"869.0"^^<http://w	"2018-05-07T13:30:00"^^<htt	Barometer_1	pressure
Obs_4	"65.0"^^<http://ww	"2018-05-07T13:31:05"^^<htt	Hygrometer_1	humidity
Obs_8	"8"^^<http://www.	"2018-05-07T13:45:00"^^<htt	Heliograph_1	cloudcover
Obs_6	"108.0"^^<http://w	"2018-05-07T13:38:00"^^<htt	Weathercock_1	winds

Figure 4.14 : Résultat d'exécution de la requête 2.

Dans la figure 4.15 nous exécutons la requête DESCRIBE qui demande de décrire « Thermometer\_1 ». Le résultat est sous forme de triplets RDF en substituant les valeurs dans les modèles de données.

SPARQL query:

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ex: <http://www.semanticweb.org/amar/ontologies/2018/2/meterologicalOnto#>
DESCRIBE ex:Thermometer_1
    
```

Subject	Predicate	Object
Thermometer_1	rdf:type	owl:NamedIndividual
Thermometer_1	hasLocation	Loc_1
Thermometer_1	rdf:type	owl:NamedIndividual
Thermometer_1	rdf:type	owl:NamedIndividual
Thermometer_1	hasID	TmpSen001
Thermometer_1	rdf:type	owl:NamedIndividual
Thermometer_1	rdf:type	Sensor
Thermometer_1	rdf:type	owl:NamedIndividual
Thermometer_1	onPlatform	Station_1
Thermometer_1	rdf:type	owl:NamedIndividual
Thermometer_1	rdf:type	owl:NamedIndividual
Thermometer_1	observedProperty	Air_Temperature

Figure 4.15 : Résultat d'exécution de la requête DESCRIBE.

## 5. Définition des règles avec SWRL Tab

Le SWRLTab est un plugin Protégé qui fournit un environnement de développement pour travailler avec les règles SWRL. Nous utilisons SWRLTab Protégé Plug-in 2.0.2, et le raisonneur Pellet Plug-In 2.2.0.

Avant de commencer il faut vérifier que le raisonneur Pellet est sélectionné. Dans le menu **système/Reasoner** nous sélectionnons Pellet. Maintenant nous démarrons le raisonneur **Reasoner/Start reasoner**.

### 5.1. Le moteur de règles DROOLS

Un moteur de règles est un système capable de définir des règles et de les appliquer à des faits. En vulgarisant, un fait est une donnée.

Le SWRLAPI Drools est un plug-in pour le SWRLAPI qui supporte l'exécution des règles SWRL en utilisant le moteur de règles Drools.

## 5.2. Edition des règles

Dans le menu système **Window/Tabs** nous sélectionnons **SWRLTab** pour ajouter son onglet qui contient deux vue horizontale (Figure 4.16) la vue en haut doit contenir les règles que nous allons ajouter et celle en bas est pour les différents messages de l'exécution des règles.

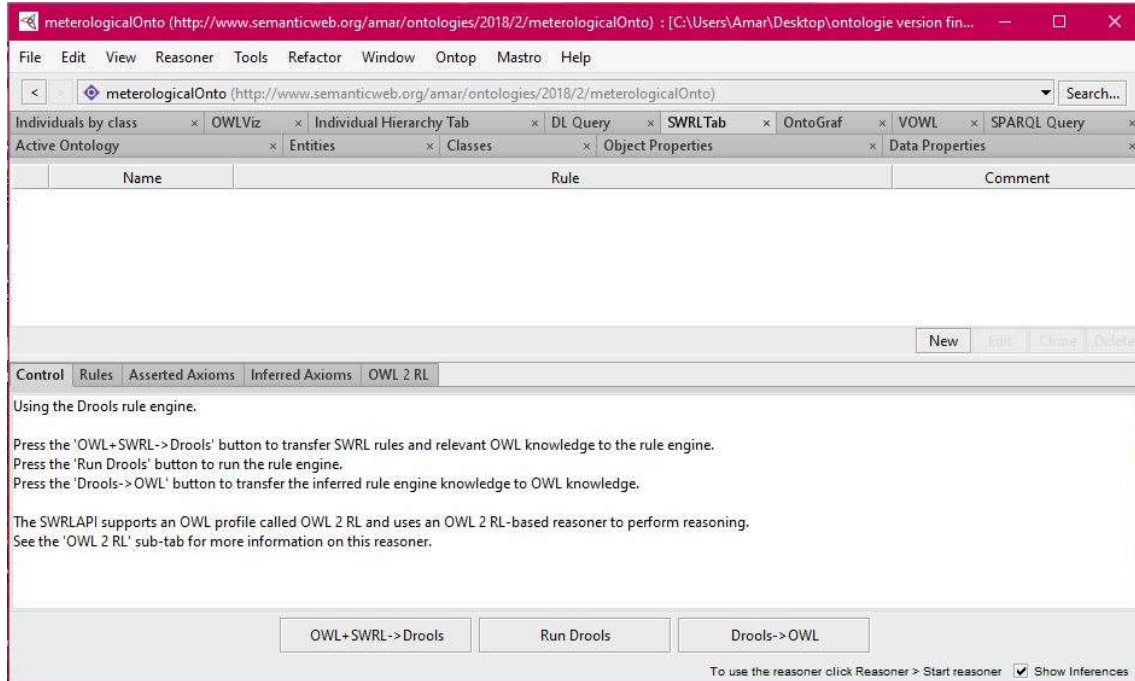


Figure 4.16 : L'onglet SWRL Tab.

En cliquant sur **New**, une fenêtre apparaît qui permet de saisir une règle (Figure 4.17) et en appuyant sur **Ok** pour l'ajouter.

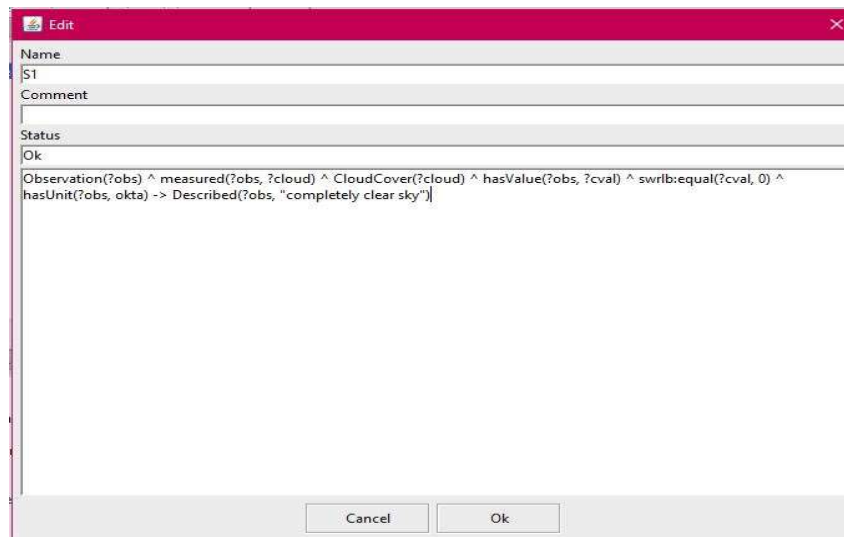


Figure 4.17 : L'éditeur de règles.

### 5.3. Exécution des règles

Après l'ajout des règles dans la vue en bas en cliquant sur **OWL+SWRL->Drools** pour transférer les règles SWRL et les connaissances de l'ontologie au moteur de règles, puis sur **Run Drools** pour exécuter le moteur de règles en suite sur **Drools->OWL** pour transférer les connaissances du moteur de règles déduites aux connaissance OWL.

Nous allons exécuter ces deux règles :

**Règle 1** : Si la valeur de la couverture nuageuse, collectée à un emplacement donné, est égale à 0 okta, alors un moteur de raisonnement basé sur des règles peut déduire de nouvelles connaissances : le ciel est complètement dégagé. Cette règle peut être décrite en SWRL par la phrase suivante :

```
Observation(?obs) ^ measured(?obs, ?cloud) ^ CloudCover(?cloud) ^ hasValue(?obs, ?cval) ^
    swrlb:equal(?cval, 0) ^ hasUnit(?obs, okta) -> Described(?obs, "Completely clear sky")
```

**Règle 2** : La règle suivante stipule que si la température est inférieure à 0 degré Celsius et qu'il pleut, les routes sont potentiellement glacées. Cette règle peut être décrite par la phrase SWRL suivante :

```
Observation(?obs1) ^ measured(?obs1, ?prec) ^ Rain(?prec) ^ measured(?obs1, ?temp) ^
    Temperature(?temp) ^ hasValue(?obs1, ?tval) ^ swrlb:lessThanOrEqual(?tval, 0) ^
    hasUnit(?obs1, Celsius) -> Described(?obs1, "Icy Roads")
```

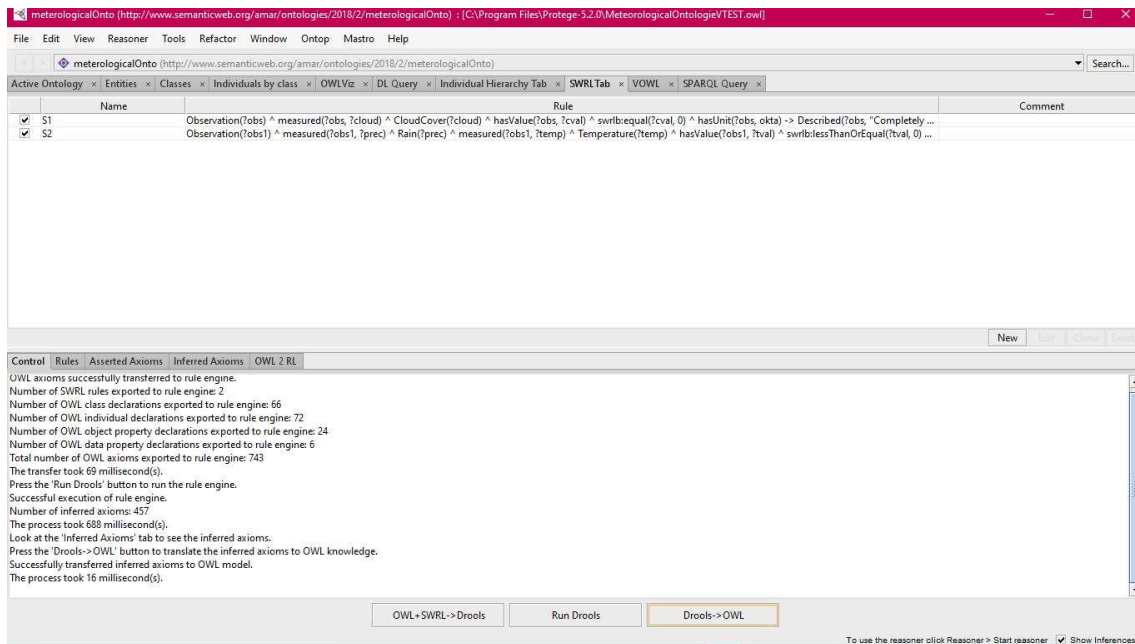


Figure 4.18 : L'exécution des règles.

Les figures suivantes montrent les observations avant l'exécution des règles :

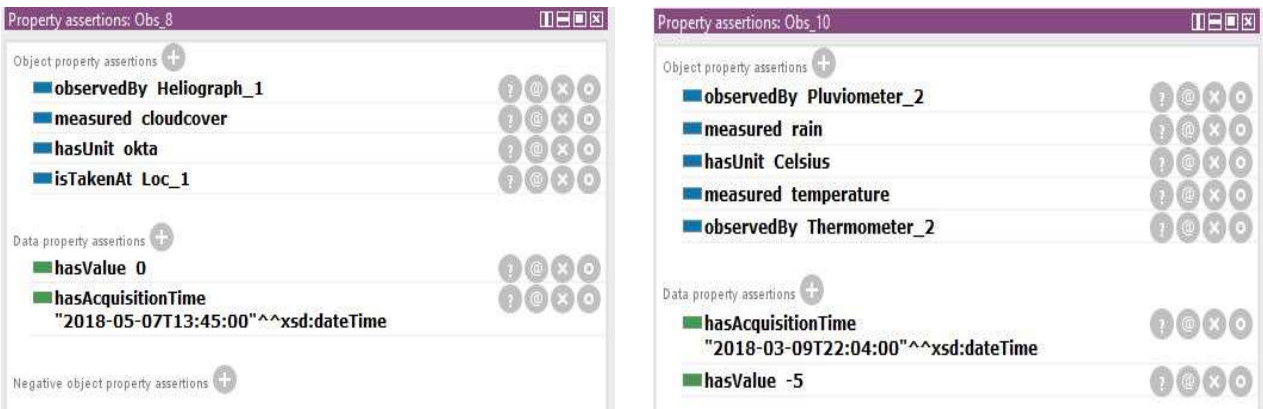


Figure 4.19 : Les observations avant l'exécution des règles.

Les figures suivantes après l'exécution des règles nous voyons qu'il a ajouté les descriptions aux observations :

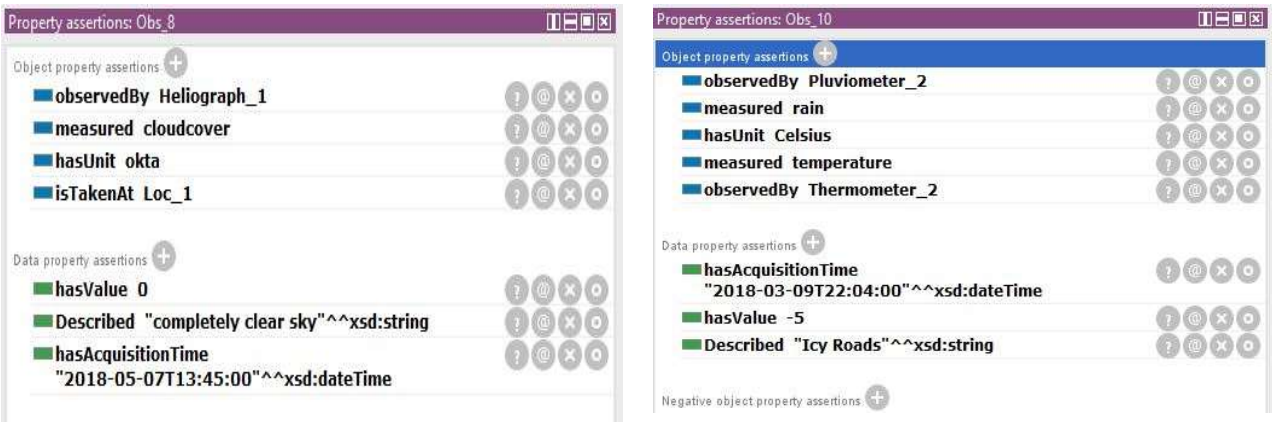


Figure 4.20 : Les observations après l'exécution des règles.

## 6. Conclusion

Dans ce chapitre nous avons présenté les détails de l'implémentation de notre ontologie sous l'éditeur protégé, d'une part. D'autre part, l'annotation sémantique des capteurs et leurs observations avec le langage RDF. Ensuite nous avons interrogé notre ontologie avec le langage SPARQL et enfin nous avons défini un ensemble de règles, en utilisant le langage SWRL, qui permet de déduire de nouvelles connaissances à partir des données de capteurs qui sont sémantiquement annotées.

# CONCLUSION ET PERSPECTIVES

*” Ce n'est pas la fin.  
Ce n'est même pas le commencement de la fin.  
Mais, c'est peut-être la fin du commencement “*

Winston Churchill

Ce travail a été motivée par la nécessité de rendre le traitement des données des capteurs interopérable. Les technologies du Web sémantiques peuvent améliorer l'interopérabilité et l'accessibilité des descriptions et des observations des capteurs en les enrichissant sémantiquement.

Dans notre mémoire nous avons proposé une ontologie web de capteur météorologique. Cette ontologie a été développée en étendant l'ontologie SSN W3C avec des connaissances ontologiques pour les capteurs, les observations, des situations, mais aussi avec des connaissances ontologiques pour la modélisation temporelles et spatiales des localisations.

Il y a plusieurs directions de recherche futures pour ce travail. Premièrement, nous complétons des instances et des règles pour décrire davantage de ressources Web de capteurs météorologiques et acquérir des connaissances utiles à partir de données existantes, publier les ensembles de données de capteurs en tant que données liées pour permettre l'interrogation et l'analyse des descriptions de capteurs et des flux de capteurs collectés et fournir une interface utilisateur conviviale permettant aux utilisateurs finaux d'annoter, d'interroger et de raisonner sémantiquement avec l'ontologie proposée.

# BIBLIOGRAPHIE

- [1] D. P. Deng, G. S. Mai, T. R. Chuang, R. Lemmens and K. T. Shao, (2011) “Social Web Meets Sensor Web: From User-Generated Content to Linked Crowdsourced Observation Data”, the Semantic Web – ISWC Lecture Notes in Computer Science, Springer Berlin Heidelberg, vol. 7031, pp 257–272.
- [2] K. R. Llanes, M. A. Casanova, and N. M. Lemus, (2017) “From sensor data streams to linked streaming data: a survey of main approaches”, Journal of Information and Data Management, vol. 7, no. 2, pp. 130–140.
- [3] A. Sheth, C. Henson, and S. Sahoo, (2008) “Semantic sensor web”. IEEE Internet Computing, vol.12 (4), pp. 78–8.
- [4] J. Charlet, P. Laublet and C. Reynaud. (2003) Web sémantique. Rapport final de l’action spécifique 32, cnrs. STIC (version 3 de décembre 2003), publié chez Cépadués (Hors-série de la collection Information interaction intelligence).
- [5] T. Berners-Lee, J. Hendler, et O. Lassila, O. (2001). The Semantic Web. Scientific American.
- [6] T. Raimbault, (2008) « Transition de modèles de connaissances. Un système de connaissance fondé sur OWL, Graphes conceptuels et UML », Thèse de doctorat. Université de Nantes.
- [7] M. K. KHELIF. (2006) Web sémantique et mémoire d’expériences pour l’analyse du transcriptome. UNIVERSITE de Nice-Sophia Antipolis.
- [8] S. Handschuh et S. Staab (2003). Annotation for the semantic web. Vol. 96of Frontiers in Artificial Intelligence and Applications, IOS Press.
- [9] M. Nilsson, M. Palmer, et al. (2002) Semantic Web Metadata for e-Learning – Some Architectural Guidelines. 11th World Wide Web Conference (WWW2002). Hawaii, USA.
- [10] Resource Description Framework (RDF). <http://www.w3.org/RDF/>.
- [11] S. Henry, C. Thompson, M. Sperberg-McQueen, Shudi (Sandy) Gao, Noah Mendelsohn, David Beech, and Murray Maloney, (2006) « XML Schema 1.1 Part 1: Structures ». Technical report. W3C. <http://www.w3.org/TR/xmlschema11-1/>
- [12] M. Hemam, (2012) « Développement des ontologies multi-points de vue : une approche basée sur la logique de descriptions », Université Mentouri Constantine, Thèse de doctorat.
- [13] D. Brickley, R.V. Guha, (Eds.) (2004) “RDF Vocabulary Description Language 1.0: RDF Schema.” W3C Recommendation. <http://www.w3.org/TR/rdf-schema/>
- [14] RDF-Schema. <http://www.w3.org/TR/rdf-schema/>

- [15] M.K. Smith, C. Welty, D.L. McGuinness. (2004) "OWL Web Ontology Language", <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>.
- [16] I. Horrocks, P. Patel-Schneider, H. Boley, S. Tabet, B. Grosz, M. Dean. (2010) SWRL: A Semantic Web Rule Language combining OWL and RuleML.
- [17] A. Chebotko, S. Lu, H. M. Jamil, et F. Fotouhi, (2006). Semantics Preserving SPARQL-to-SQL Query Translation for Optional Graph Patterns. Technical Report TR-DB-052006-CLJF. <http://www.cs.wayne.edu/~artem/main/research/TR-DB-052006-CLJF.pdf>
- [18] D. Mun, K. Ramani. (2011) Knowledge-based part similarity measurement utilizing ontology and multi-criteria decision-making technique. *Advanced Engineering Informatics* 25.
- [19] V. Hirankitti, T. Xuan. (2011) A meta-reasoning approach for reasoning with SWRL ontologies, *International Multiconference of Engineers*.
- [20] R. Neches, R. Fikes, T. Finin, T. Gruber, R. Patil, T. Senator, and W. R. Swartout. (1991) "Enabling Technology for Knowledge Sharing". *AI Magazine*, Volume 12, No. 3, Pages 37-56. Fall.
- [21] A. Gómez-Pérez. (1999) "Ontological Engineering: A state of the art". *Expert Update*, Volume 2 (3), Pages 33-43.
- [22] N. Guarino "Some organizing principles for a unified top-level ontology". *AAAI Spring Symposium on Ontological Engineering*, 57-63.
- [23] "Top-level ontological categories," *International J. of Human-Computer Studies*, Volume 43:5/6, Pages 669-686. (1996).
- [24] G. Van Heijst, A. Schreiber et B. J. Wielinga. (1997) "Using Explicit Ontologies in KBS Development". *International Journal of Human and Computer Studies / Knowledge Acquisition*, Volume 46, No (2/3), Pages 183-292.
- [25] W. N. Borst (1997). "Construction of Engineering Ontologies for knowledge sharing and reuse". CTIT Ph. D-thesis series No. 97-14. Center for Telematica and Information.
- [26] T. Gruber. (1993) "A Translation Approach to Portable Ontology Specifications". *Knowledge Acquisition*, Volume 5, No 2. Pages 199-220.
- [27] R. Mizoguchi, J. Vanwelkenhuysen, M. Ikeda. (1995) "Task Ontology for Reuse of Problem Solving Knowledge". *Knowledge Building & Knowledge Sharing (2nd International Conference on Very Large-Scale Knowledge Bases)*, Enschede, The Netherlands, Pages.46-59.
- [28] R. Mizoguchi, K. Kozaki, T. Sano and Y. Kitamura. (2000) "Construction and Deployment of a Plant Ontology". *Knowledge Engineering and Knowledge Management - Methods, Models and Tools*, The 12th International Conference, EKAW2000, Lecture Notes in Artificial Intelligence 1937, Springer-Verlag, Pages.113-128, Juan-les-Pins, France.

- [29] A. Inaba, Thepchai Supnithi, Mitsuru Ikeda, Riichiro Mizoguchi, Jun'ichi Toyoda. (2000) "An overview of Learning Goal Ontology". Proc. of ECAI2000 Workshop on Analysis and Modelling of Collaborative Learning Interactions, Pages.23-30, Berlin, Germany.
- [30] A. Maedche, B. Motik, L. Stojanovic, R. Studer et R. Volz. (2003) "Ontologies for Enterprise Knowledge Management". IEEE Intelligent Systems, Volume 18 (2), Pages 26-33.
- [31] M. Uschold, et M. Gruninger. (1996) "Ontologies: principles, methods and applications", Knowledge Engineering Review, vol. 11, no. 2.
- [32] [https://fr.wikipedia.org/wiki/Frame\\_representation\\_language](https://fr.wikipedia.org/wiki/Frame_representation_language). (2018)
- [33] J.F. Sowa, (1984) "Conceptual structures: information processing in mind and machine", Addison-Wesley.
- [34] M. Buchheit, f. Donini, and A. Shaerf. (1993) "Decidable reasoning in terminological knowledge representation systems". Journal of artificial intelligence research, 1 : 109-138.
- [35] A. Napoli, (1997) "une introduction aux logiques de descriptions" N° 3314.
- [36] J. Domingue, E. Motta, and O. Corcho Garcia, (1999) Knowledge Modelling in WebOnto and OCML: A User Guide. Available in PDF or gzipped PDF
- [37] OILED, OIL Editor Home Page, <http://oiled.man.ac.uk/>, (2004).
- [38] N. NOY et M. A. MUSEN, (2002) Evaluating ontology-mapping tools: requirements and experience, in Proceedings of the Workshop on Evaluation of Ontology Tools (EON'2002) at EKAW'2002.
- [39] ODE, Ontology Design Environment Home Page, <http://www.swi.psy.uva.nl/wondertools/html/ODE.html>, (2004).
- [40] ONTOEDIT, Ontology Editor Home Page, <http://www.ontoprise.de/com/>, (2004).
- [41] <http://protege.stanford.edu/support.php>. (2010)
- [42] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, et Y. Katz, (2007). Pellet: A practical OWL DL reasoner. Web Semantics: Science, Services and Agents on the World Wide Web, 5(2), 51–53.
- [43] B. Motik, et U. Sattler (2006). A Comparison of Reasoning Techniques for Querying Large Description Logic A Boxes. In the 13th international conference on Logic for Programming, Artificial Intelligence, and Reasoning (pp. 227–241).
- [44] R. Shearer, B. Motik et I. Horrocks, (2008). HermiT: A Highly-Efficient OWL Reasoner. In OWLED (pp. 1–10).

- [45] D. Tsarkov, et I. Horrocks, (2006). FaCT ++ Description Logic Reasoner: SystemDescription. In Automated Reasoning (IJCAR 2006 (pp. 292–297).
- [46] V. Haarslev, et M. Ralf, (2000). Consistency Testing: The RACE Experience. In Proceedings, Automated Reasoning with Analytic tableau and related methods (pp. 57–61).
- [47] Pierre Sidoine V. Donfack Guefack. (2013) Modélisation des signes dans les ontologies biomédicales pour l'aide au diagnostic. Médecine humaine et pathologie. Université Rennes 1.
- [48] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, (2001) Research paper “Wireless sensor networks: a survey”.
- [49] F. Abdelfatah. (2008) Rapport Développement d’une bibliothèque de capteurs. Université Montpellier 2 Sciences Et Techniques.
- [50] T. L. van Zyl, I. Simonis and G. McFerren, (2008) Research paper The Sensor Web: systems of sensor systems.
- [51] Y, Challal, (2008) Support de cours « Réseaux de Capteurs Sans Fils ».
- [52] K,Delin. (2002) The Sensor Web: A macro-instrument for coordinated sensing. Sensors, 2 (1), 270\_285.
- [53] NASA Geospatial Interoperability Office 2005. (2005) Geospatial interoperability return on investment. Technical report.
- [54] A. Gray, et al. Technical report Adaptation and re-configuration in the global sensor web, NASA., 2007.
- [55] K. Delin, et al. (2005) Environmental studies with the Sensor Web: Principles and practice. Sensors, 5 (1-2), 103\_117.
- [56] S. Liang, et C. Tao, (2005) A distributed geospatial infrastructure for the sensor web. Computers and Geosciences, 31 (2), 221\_231.
- [57] M. Botts, et al. (2006) Open GIS sensor web enablement architecture document. Technical report, Open Geospatial Consortium Inc.
- [58] G. Percival, et C. Reed, (2006) OGC sensor web enablement standards. Sensors and Transducers Journal, 71 (9), 698\_706.
- [59] A. Sheth, C. Henson, and S. S. Sahoo. (2008) Semantic Sensor Web.
- [60] I. Simonis, (2007) Technical report. South African international workshop on Sensor Web Enablement. Meraka Institute, CSIR.
- [61] D. Comer, (1999) Computer networks and internets. Upper Saddle River, NJ: Prentice Hall.

- [62] S. Cox, (2006) Observations and measurements. Technical report, Open Geospatial Consortium Inc.
- [63] A. Na, and M. Priest, (2006) Sensor observation service. Technical report, Open Geospatial Consortium Inc.
- [64] I. Simonis, (2005) Sensor alert service implementation specification. Technical report, Open Geospatial Consortium Inc.
- [65] V. Huang, M. Kashif Javed. (2014) Semantic Sensor Information Description and Processing. The Second International Conference on Sensor Technologies and Applications.
- [66] M.A. Matin and M.M. Islam. (2012) Research paper “Overview of Wireless Sensor Network”.
- [67] D.E Boubiche, (2008) Protocole de routage pour les réseaux de capteurs sans fils. Mémoire de magistère. Université de l’Hadj Lakhdar-Batna.
- [68] D.J. Russomanno, C. Kothari, and O. Thomas. (2005) “Sensor ontologies: from shallow to deep models”. Proceedings of the Thirty-Seventh Southeastern Symposium on System Theory, pp. 107– 112.
- [69] M. Compton, H. Neuhaus, K. Taylor, and Ki-N. Tran. (2009) “Reasoning about sensors and compositions”. Proceedings of the Semantic Sensor Networks., pp. 33–48.
- [70] M. Compton, P. Barnagh, L. Bermudez, R. Garcia-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, K. Janowicz, W. David Kelsey, D. Le Phuoc, L. Lefort, M. Leggieri, H. Neuhaus, A. Nikolov, A. Page, A. Passant, A. Sheth, and K. Taylor. (2012) “The ssn ontology of the w3c semantic sensor network incubator group”. Web Semantics: Science, Services and Agents on the World Wide Web, 17(C), pp. 25–32.
- [71] R. DIENG, O. CORBY, F. GANDON, A. GIBOIN, J. GOLEBIEWSKA, N. MATTA et M. RIBIERE, (2001) « Méthodes et outils pour la gestion des connaissances : une approche pluridisciplinaire du knowledge Management ». Dunod, 2 edition.
- [72] F. Gandon, « Ontologies informatiques ». (2006).
- [73] M. Fernandez, A. Gomez-Perez et N. Juristo, (1996) “METHONTOLOGY: from ontological art toward ontological engineering”. Spring symposium series on ontological engineering. AAAI97, USA.

# An approach for Semantic Enrichment of Sensor Data

Meriem Djezzar  
LIRE Laboratory,  
Department of Computer Science  
University of Constantine 2  
Constantine, Algeria  
djezzar.meriem@gmail.com

Fatima Zahra Amara  
Department of Computer Science  
University of Abbes Laghrour  
Khenchela, Algeria  
amafatima.zahra@gmail.com

Mounir Hemam  
Department of Computer Science  
University of Abbes Laghrour  
Khenchela, Algeria  
mounir.hemam@gmail.com

Khaoula Falek  
Department of Computer Science  
University of Abbes Laghrour  
Khenchela, Algeria  
falek.khaoula@gmail.com

Moufida Maimour  
CRAN Laboratory,  
University of Lorraine  
Nancy, France  
moufida.maimour@univ-lorraine.fr

Zianou Ahmed Seghir  
Department of Computer Science  
University of Abbes Laghrour  
Khenchela, Algeria  
zianou\_ahmed\_seghir@yahoo.fr

**Abstract**—Sensor technologies and Wireless Sensor Networks (WSN) are designed to collect large volumes of heterogeneous data for monitoring environmental phenomenon. Sharing, reusability and fusion of sensor data from heterogeneous sensor networks is not possible without understanding the semantics of the data. In this paper, we focus on the interoperability of sensor data to build promising and interoperable domain-specific or cross-domain sensor Web applications. To deal with this challenge, we propose an approach based on semantic Web technologies which offers semantic interoperability for sensor data among heterogeneous WSN. For this purpose, we first propose a data vocabulary extended from SSN ontology to facilitate manual annotation on sensors, observations and measurements, by providing semantic metadata about spacial, temporal and thematic properties. We then propose to publish the sensor datasets as linked data to allow querying and analysis over collected sensor descriptions and sensor streams. Finally, we apply a reasoning mechanism on semantically annotated sensor data to deduce new or implicit knowledge, discover significant data and answer complex queries. To illustrate our approach, we apply it on an example concerning the meteorological domain.

**Keywords**— *Semantic Web, Linked Data, Sensor Web, Ontology, Semantic Annotation*

## I. INTRODUCTION

Over the last few years, wireless sensor networks have attracted a great deal of research attention due to their wide range of potential applications. Sensor networks introduced a new class of computer systems and expand the ability of individuals to remotely interact with the physical world. Sensor networks are used for numerous applications including medical sciences for patient care using biometric sensors, wildfire detection, meteorology for weather forecasting, satellite imagery for earth and space observation, agricultural land, etc.

The combination of sensor networks with the Web, web services and database technologies, was named some years ago as the Sensor Web or the Sensor Internet [7, 8, 13, 14]. The sensor Web is a special type of Web-centric information infrastructure for collecting, modeling, storing, retrieving,

sharing, manipulating, analyzing, and visualizing information about sensors and sensor observations of phenomena.

The sensors are distributed across the globe, capturing and continuously producing an enormous amount of data streams about a number of real world phenomena [1]. All these different available sensors provide heterogeneous raw data, which is provided at different formats and with no semantics to describe its meaning. The lack of common semantics intensifies the existing problem of “*too much data and not enough knowledge*” [2]. With a view to addressing this problem, sensor data can be annotated with semantic metadata to increase interoperability between heterogeneous sensor networks, as well as to provide contextual information essential for situation awareness. In this context, Semantic Web techniques can construct a shared conceptual model, enhance the sensor data semantics, and realize interaction and access of sensor data on the web.

In this paper, we present an approach for semantic enrichment of sensor descriptions (i.e. sensor characteristics) and sensor streams (i.e. data sources that represent observations and measurements collected by sensory devices), with the purposes of facilitating sensor data retrieval, enabling inference and increasing interoperability among sensor data from different sources. To deal with this challenge, we exploit semantic Web technologies [3] for several reasons:

- semantics enables an explicit description of the meaning of sensor data in a structured way, so that machines could understand and manage it;
- semantics facilitates the exchange of data and interoperability, since heterogeneous sensor data is converted according to the same vocabulary;
- sensor data streams can be easily published and linked to existing resources on the Web, so that they can be accessed and consumed by external applications;
- semantic reasoning engines can be employed to infer high-level abstractions by exploiting the domain knowledge expertise.

So, the main contributions of this paper are the following: