



JOURNAL OF SPATIAL INFORMATION SCIENCE  
Number 6 (2013), pp. 155–185

doi:10.5311/JOSIS.2013.6.104

RESEARCH ARTICLE

# A dynamic and context-aware semantic mediation service for discovering and fusion of heterogeneous sensor data

Mohamed Bakillah<sup>1,2</sup>, Steve H.L. Liang<sup>1</sup>, Alexander Zipf<sup>2</sup>, and Mir Abolfazl Mostafavi<sup>3</sup>

<sup>1</sup>Department of Geomatics Engineering, University of Calgary, Canada

<sup>2</sup>Rupprecht-Karls-Universität, Institute for GI-Science, Germany

<sup>3</sup>Centre de recherche en géomatique, Université Laval, Canada

*Received: July 25, 2012; returned: September 27, 2012; revised: December 3, 2012; accepted: January 16, 2013.*

---

**Abstract:** Sensors play an increasingly critical role in capturing and distributing observations of phenomena in our environment. The vision of the semantic sensor web is to enable the interoperability of various applications that use sensor data provided by semantically heterogeneous sensor services. However, several challenges still need to be addressed to achieve this vision. More particularly, mechanisms that can support context-aware semantic mapping and that can adapt to the dynamic metadata of sensors are required. Semantic mapping for the sensor web is required to support sensor data fusion, sensor data discovery and retrieval, and automatic semantic annotation, to name only a few tasks. This paper presents a context-aware ontology-based semantic mediation service for heterogeneous sensor services. The semantic mediation service is context-aware and dynamic because it takes into account the real-time variability of thematic, spatial, and temporal elements that describe sensor data in different contexts. The semantic mediation service integrates rule-based reasoning to support the resolution of semantic heterogeneities. An application scenario is presented showing how the semantic mediation service can improve sensor data interpretation, reuse, and sharing in static and dynamic settings.

**Keywords:** semantic mediation, semantic heterogeneity, sensor data, sensor web enablement, SQWRL, context modeling, dynamic metadata, rule-based reasoning, semantic web

---

## 1 Introduction

The development of sensor networks is opening a wide range of opportunities in a variety of domains, ranging from environmental monitoring to health-care, urban traffic management, and satellite imaging. Technical developments of sensing devices were followed by the development of the sensor web through the Open Geospatial Consortium (OGC) Sensor Web Enablement (SWE) initiative, which has as its main objective to make sensor data discoverable and accessible on the Web through standardized interfaces and specifications. According to Zafeiropoulos et al. [54], it is not unrealistic to expect that in the near future, the pervasiveness of sensor networks will significantly increase, with these sensors producing data that will be accessible over the Web. Within this context, the improvement of the capacity to discover, retrieve, interpret, and manage sensor data is a fundamental requirement. The concept of a semantic sensor web, which means semantic enablement of the sensor web, will be instrumental in offering this capacity. More specifically, the semantic sensor web combines SWE information models and service standards with semantic web technologies to improve the description of sensor data and make it more meaningful [50]. The semantic sensor web therefore aims at enabling meaningful sensor data exchange, reuse, and interpretation. For example, RDF (resource description framework) is a semantic web language that can be used to formally describe the meaning of sensor data (e.g., the observed property, the phenomenon observed, the units for measurement, etc.). Such a description will support the automated discovery of relevant sensor data with respect to a given query. The work presented in this paper also aims at contributing to the ongoing development of the semantic sensor web.

The semantic sensor web is still at an early development stage, and significant work needs to be achieved to attain the vision it conveys. "Sensor semantics" refers to the formal description of different aspects of sensor data, ranging from the description of the sensor device itself to the application domain point of view. Research on sensor semantics has resulted in the development of several sensor network ontologies [13]. Nevertheless, a common metadata model for representing the context of sensors is still lacking. In this respect, SWE standards already include the SensorML model for describing sensor metadata. However, it has been argued that the structure of this model is rather loose, making it difficult to process and compare different SensorML descriptions automatically [32]. Furthermore, recent research on remaining challenges related to the development of the semantic sensor web [13] highlights the issue of the integration and fusion of data coming from independent sensor networks. One avenue to resolve this issue is to develop semantic mediation systems that are suitable for the sensor web. While no agreed-upon definition for semantic mediation system exists in the literature, we refer to such a system as a middle layer that manages the semantics of data and resolves heterogeneity of data to enable its integration and fusion. For example, a semantic mediation system may include a domain ontology to describe data semantics, an ontology repository, a semantic annotation tool to associate data with corresponding elements of the domain ontology, a semantic service broker, and a semantic querying interface [3].

Some argue that the sensor data management field commonly considers sensor networks as distributed database systems [54]. However, approaches for semantic mediation of distributed databases are not directly applicable to the sensor web. In particular, semantics of sensor observations cannot be dealt with in the exact same way as that of geospatial data, notably because sensor semantics are more likely to be dynamic, as the context within

which data is produced is often evolving, especially in the case of mobile sensors [17, 31]. Taking these aspects into consideration, in this paper we present a context-aware and dynamic semantic mediation system for the semantic sensor web. Context-awareness is the ability of an application or a service to be aware of its physical environment or situation and to react proactively and intelligently based on such awareness [37]. Although much work has been done on context-aware systems (e.g., [48, 53]), context-awareness has not been directly exploited in the development of semantic mediation systems. However, according to Keßler et al. [34], the influence of context on semantic mediation is a well-known phenomenon. In the future, it is likely that semantic web technologies will have to consider dynamic contexts associated to sensors. The conceptual basis of the proposed context-aware semantic mediation system is a sensor metadata model for sensor observations that has been proposed in previous research published in the W2GIS 2012 conference [4]. It is shown in this paper how this model is adaptable to dynamic changes with context rules. The proposed semantic mediation system also contributes to the semantic mediation field by integrating rule-based reasoning, which goes beyond existing semantic mediation systems based on subsumption reasoning. Rule-based reasoning enables reasoning with more complex semantic structures, therefore supporting a more accurate semantic mediation process.

This paper is organized as follows: first, related work on sensor semantics and semantic mapping is presented to set the grounds for this research. Then, we present the sensor metadata model and dynamic context rules in Section 3. This is followed by the presentation of the context-aware semantic mediation system in Section 4. A case study of the system is presented in Section 5, where it is demonstrated that the proposed system can support meaningful fusion of heterogeneous sensor data in static and dynamic settings. Conclusions and open research avenues are provided in Section 6.

## 2 Related work

Sensor networks need to be upgraded with semantics in order to obtain so-called “semantic sensor networks,” which support reuse and integration of sensor data, as well as querying and extracting complex knowledge from heterogeneous sensors. Semantic sensor networks are one of the main steps towards the development of the sensor web.

Semantic sensor networks require declarative descriptions of sensors, including their physical characteristics, such as their location and power supply; their observations and characteristics of these observations, such as accuracy and frequency of measurements; and information on the domain, such as the property being observed [12]. The OGC’s SWE standards propose syntactic models for sensors, including SensorML and the O&M (observation and measurements) models [7]. These models enable interoperability at the syntactic level, but are not designed to address the issue of semantic interoperability. Moreover, one of the drawbacks of SensorML is that it is a generic standard that allows the specification of the same information through different structures [32]. This makes it difficult to process and compare different SensorML descriptions automatically. To support the discovery of relevant sensor data, we need a consistent set of relevant metadata elements to describe all available sensors services.

Ontologies, which are defined as formal specifications of a conceptualization [26], provide definitions of concepts and relations to describe a domain of interest [1]. Ontologies are proposed as a solution to the semantic interoperability problem [6]. By formally represent-

ing semantics of data, ontologies support the interpretation and sharing of data. Therefore, a certain number of ontologies for sensors have been developed to address the problem of heterogeneous sensor data. For a technology review in this area, see [12]. Notably, Russomanno et al. [47] have developed OntoSensor, a generic sensor ontology which is drawn from SensorML as well as the IEEE SUMO (Suggested Upper Merged Ontology). OntoSensor was designed to support sensor selection, reasoning with, and querying sensor data. OntoSensor was described as one of the most comprehensive ontologies for sensors [12]. Other ontologies for sensors include the ontology developed by Kim et al. [35], which is an extension of OntoSensor for service-oriented sensor networks that describe the functionality of sensor services, as well as the context and physical characteristics of sensors. Eid et al. [19] have developed a sensor ontology that mainly focuses on describing sensor observations, including accuracy and frequency of observations, and the sensors' responses to stimuli. The A3M3 [27] and ISTAR [25] ontologies are more focused on describing the sensor device itself (the identification and manufacturing details of the sensor, information on deployment and configuration, etc.). Therefore, the latter ontologies are more useful for representing the capabilities of sensors and assessing whether a sensor can fulfill a given task.

Existing ontologies are useful to provide a uniform terminology to describe sensor data. However, their goal is not to provide a consistent set of relevant metadata elements to describe all available sensors services, required to compare sensors on a common basis. Therefore, in this paper, we adopt a sensor metadata model that was developed in previous research [4]. While the proposed metadata model can be employed to provide a uniform description of available sensors, terms from existing sensor ontologies can be employed to instantiate the model using a consistent terminology. Therefore, the research presented in this paper is complementary to the above-mentioned sensor ontologies.

Nevertheless, it is still unlikely that a single sensor ontology would be used across a network of heterogeneous sensors. Some sensors can be described using generic sensor ontologies, while others may rely on more specialized and domain-specific ontologies. Therefore, despite having a common metadata model, semantic mapping approaches are required to reconcile heterogeneous instances of the model.

## 2.1 Semantic mapping

Kalfoglou and Schorlemmer [33] define semantic mapping as a morphism consisting of a set of functions assigning the symbols used in one ontology to the symbols of another ontology. More concretely, the semantic mapping process takes as input two or more ontologies, (composed of classes, properties, rules, etc.) and returns the semantic relations (also called alignments) between ontology components. Semantic mapping relations can consist of a simple match (i.e., an element of ontology O1 matches a second element of ontology O2), or it can include several types of semantic relations (equivalence, overlap, containment, etc.) Meaningful data exchanges are supported by the semantic mapping process of ontologies. The emergence and spreading of ontologies have given rise to numerous semantic mapping approaches, These approaches are thoroughly reviewed in Euzenat and Shvaiko [21], who have identified the following categories of semantic mapping techniques:

1. Linguistic techniques compare terms to identify linguistic relations among them (e.g., synonymy, hypernymy, hyponymy, etc.). Linguistic techniques are often employed as a preliminary phase of the complete semantic mapping process (e.g., the S-Match

- algorithm of Giunchiglia et al. [24], the OWL-Lite Aligner (OLA) of Euzenat and Valtchev [22]; the FALCON matching system of Hu and Qu [29]);
2. Constraint-based techniques use structural similarity to find matches between ontology/schema elements. For example, if two classes have a similar set of properties, it can be an indication that they represent similar entities. Structures being used include taxonomies, graph of ontology, and properties of concepts (Giunchiglia et al. [24]; Hu and Qu [29]);
  3. Techniques based on auxiliary information use external resources to find semantic relations, for example global or domain ontologies, dictionaries, or thesauri (e.g., the similarity flooding algorithm of Melnik et al. [42] and the COMA++ system of Massmann et al. [41]).
  4. Formal matching techniques based on logic reasoning engines use reasoning tools to infer semantic relations, such as S-Match (Giunchiglia et al. [24]) and Ctx-Match (Serafini et al. [49]), which identify hierarchical (subsumption) relations between classes of different ontologies based on SAT (satisfiability) solvers.

Efficient semantic mapping systems combine several of the above techniques to retrieve a maximal number of relevant mappings (e.g., [5, 24, 29, 49]). In this paper, the semantic mapping system that we propose also combines techniques from these four categories.

There exist some approaches that pertain to the field of semantic sensor networks and that use some form of semantic mediation, such as semantic-based services for query processing on sensor data [39]; semantic mediation for publish-subscribe sensor data services [44,52]; and in support of sensor plug-and-play frameworks [10]. However, in this paper, the focus is on proposing a semantic mapping approach that is specifically dedicated to sensors, while above-mentioned approaches are mostly dedicated to geospatial data. Although concepts used for semantic interoperability of geospatial data can be reused for semantic interoperability of sensor data, some adaptation is required [55]. More specifically, in contrast with geospatial databases, sensor metadata is dynamic and can be modified in real time to reflect changes in the environment, mobile devices, etc. Therefore, metadata for sensors is also dynamic. In the proposed approach, we integrate dynamic nature of metadata into the semantic mapping process. In addition, we exploit the most recent semantic reasoning techniques to find semantic relations. Semantic reasoning is the inference of logical consequences that result from the combination of asserted facts and logic rules [11]. In the context of semantic mapping process, we propose in this paper that:

- basic asserted facts are the definitions of classes and properties describing sensor data;
- logic rules are rules that express the conditions for a semantic relation to hold between two classes or properties; and
- a logical consequence is the semantic relation between two classes or properties.

In contrast with approaches that use subsumption reasoning [24, 49], which only enables to find generalization/specialization relations, rule-based reasoning engines can support the identification of a broader range of semantic relations, and compare more complex ontological structures. For example, with a rule-based approach, it is possible to specify the conditions for several types of semantic relations to hold, such as overlap, weak-overlap, strong-overlap, compatibility, as we will present in this paper. A broader range of semantic relations allows detecting various states of similarity among data sets. Meanwhile, subsumption-based reasoning cannot identify semantically overlapping data sets,

and if used in a retrieval system, can possibly discard relevant data sets. In this paper, we put forward the use of semantic reasoning engines to improve the ability of semantic mapping systems to identify more relevant semantic relations and to support the user in a better understanding of shared sensor data.

Of note is that the rule-based approach is also similar to the concept of active databases. Active databases combine database technology with rule-based programming to add reactive capabilities to the database [14]. They contrast with traditional databases where only users or applications can actively perform operations on the available data. However, the approach proposed in this paper provides more reasoning capabilities because it is based on a richer representation of semantics with an ontological approach.

Before presenting the semantic mapping system, the following section presents the sensor metadata model used as knowledge representation by the dynamic semantic mapping service.

### 3 Sensor metadata model for modeling context

The notion of context has been considered from various perspectives, including from those of context-aware systems [2, 16, 51, 55], semantic interoperability [8], semantic similarity [34, 46], and geospatial semantic web [18]. Consequently, context has been given a variety of definitions, which are more-or-less dependent on the application domain. Among the most frequently cited definitions, we find for example the definition of [16], which states that context is “anything that can be used to characterize the situation of an entity.”

In the field of semantic interoperability, context can be considered as the set of properties that define a geospatial concept [8], or “any information that helps to specify the similarity of two entities more precisely concerning the current situation” [34]. In the field of sensor web, Zander and Schandl [55] define context as the information that gives meaning to the terms used to describe sensor data. While the various context definitions provided could be valid in our research, we also consider in this work that definitions provided by [16] and [34] correspond to the focus of this paper. Specifically, we consider context to be the current information that characterizes the situation of an entity (in this case, a sensor data set or stream), and that helps to specify its relation with other entities or its relevance with respect to a query. Our goal in this paper is to study and formalize the influence of the context on semantic mapping for the sensor web. We cannot assume that the terms used to describe a set of sensor observations are sufficient to enable the comparison of these observations nor the assessment of the relevance of these observations with respect to users’ requirements. For example, a set of observations on “water pollution” could mean that the concentration of some water pollutant is being measured, for example, at a natural, drinkable water spring; or in watersheds near industrial facilities. Hence, a formalized representation of context is necessary to support semantic mapping for the semantic sensor web. In this paper, we formalize the representation of context with the following sensor metadata model.

#### 3.1 Sensor metadata model

The sensor metadata model defines the context of the sensor. By committing to a common metadata model, applications can benefit from shared semantics and therefore enhance



semantic interoperability [45]. The proposed model, which is formalized in Figure 1 with UML, is compliant with the OGC’s SWE standards for modeling sensor data and observations, notably, the O&M standard data model for sensor data.

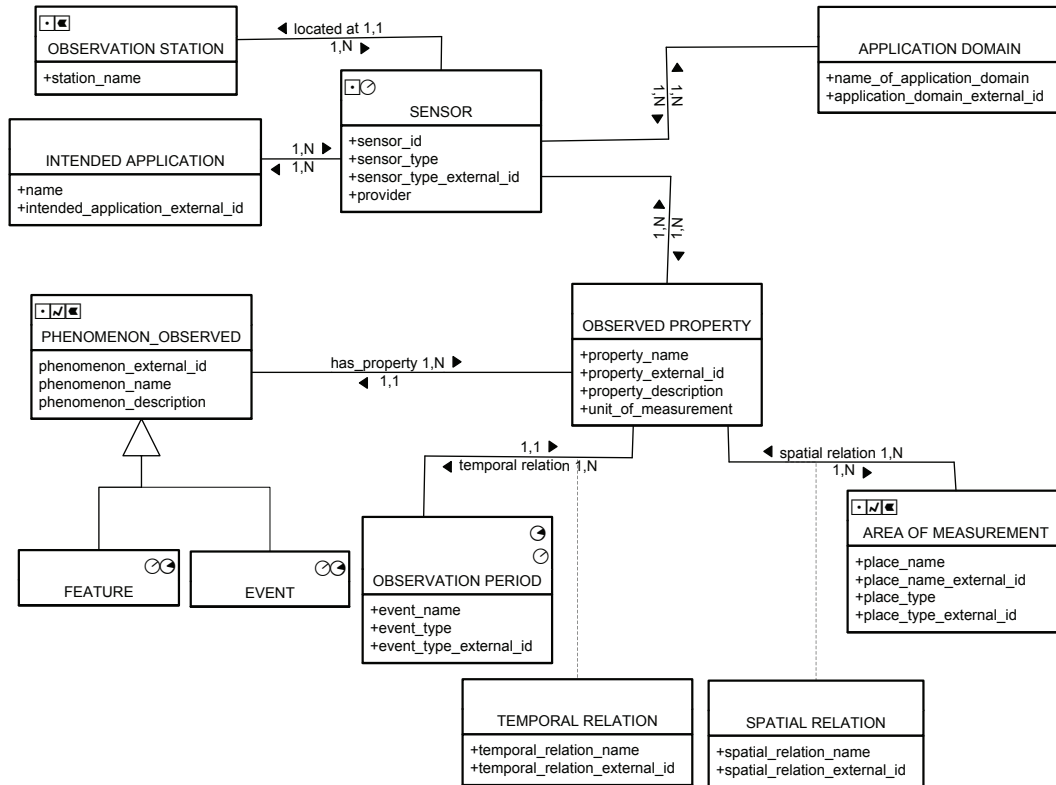


Figure 1: Sensor metadata model.

A sensor is described by the sensor type, the provider of the sensor, its localization, and the observation station. We associate the sensor with its *intended application* and its *application domain* (e.g., hydrology, meteorology, etc.). The intended application is the type of activity that is originally meant to be performed with the data (e.g., soil moisture monitoring, toxic gas dissemination monitoring, etc.). In the O&M sensor data model, each sensor is linked to one or several observations performed on a phenomenon. In our model, the phenomenon is abstracted with the class *phenomenon observed*. The phenomenon observed can be a physical object, represented with the class *feature* (e.g., watercourse, building, etc.), or an *event* (e.g., a storm, an earthquake, etc.). Observed properties are qualities of the phenomenon that can be measured, such as soil moisture, wind speed, etc. The *observed property* class is linked to the *area of measurement* class, which represents the point, line, or polygon where the observations were made. The *area of measurement* is associated with a *place*, which is the name of the location (e.g., University of Calgary) and the type of place (e.g., university). This alternative way (besides spatial coordinates) of providing the location of the measurements is offering several ways of specifying the location of interest.

The observed *property* class is linked to the area of measurement with a *spatial relation*. For example, the observations could have been made near the University of Calgary. For this research, we have considered the spatial relations provided in the OpenCyc spatial properties and relations ontology [15]. The observed property class is also linked to an observation period. This period covers the time during which the measurements were made; it can be a time interval or an event. Specifying the observation period allows current data as well as archived sensor data to be dealt with. Just as spatial coordinates can be counter-intuitive when compared to the place of interest (e.g., name of a city) [45]; so a time interval can be less intuitive to specify than the corresponding event (e.g., hurricane Katrina). The observed property class is linked to the observation period with a temporal relation. For example, a set of observations could have been performed before the hurricane. Here again, we use the temporal relations provided in the OpenCyc temporal ontology. Of note also is that temporal relations allow users to specify temporal relations between different sensor data sets in their queries (e.g., data on gas density gathered “after” data on temperature increases). OpenCyc is an open-source project that aims at creating a comprehensive ontology of common sense knowledge to support human-like reasoning in artificial intelligence applications. The OpenCyc spatial properties and relations ontology contains spatial relations of direction and orientation, relative position of objects, and mereological relations. Including spatial relations in the model allows the user to specify more intuitive queries involving places of interest.

The *observed property* class is also linked to an *observation period*. This period covers the time during which the measurements were made; it can be a time interval or an event. Specifying the observation period allows dealing with current data as well as archived sensor data. Just as spatial coordinates can be less intuitive compared to the place of interest (e.g., name of a city) [45], a time interval to represent the time frame of interest can be harder to specify than the corresponding event (e.g., hurricane Katrina). The *observed property* class is linked to the *observation period* with a temporal relation. For example, a set of observations could have been performed *before* the hurricane. Here again, we use the temporal relations provided in the OpenCyc temporal ontology. Of note also is that temporal relations allow users to specify temporal relations between different sensor data sets in their queries (e.g., data on gas density gathered “after” data on temperature increase).

In order to better compare the context elements of different sensors during the semantic mapping process, it is useful to identify different types of context elements. We propose the following categorization of the types of contexts elements:

- functional context element: a function, task or role associated with the sensor, for example, intended application is a functional context element;
- situational context element: something that describes the situation, for example, dryness is a possible situational context of the phenomenon observed “soil”;
- classification context element: a category that a class is a member of; for example, “bridge” can have as classification context “transport infrastructure” or “hazard to air navigation”;
- spatial context element: a geographic area or place, which may be described with a spatial relation of proximity, topology, or orientation (for example, a sensor close to a floodplain); and
- temporal context element: a period of time or an event (for example, an historical period or the event of earthquake).



Because each sensor is made available by different providers, sensor descriptions are semantically heterogeneous. To resolve these heterogeneities, the terms used in sensor descriptions are referenced to a common and formal vocabulary, i.e., a semantic reference system, or reference ontology [6]. According to Kuhn and Raubal [38], the semantic reference system consists of a semantic datum, which is the basic vocabulary used to describe a given universe of discourse; a semantic reference frame (SRF), which is a concept structure defining the conceptualization underlying the use of this vocabulary; and a function that links a term used in an application (for example, a term used within the description of a sensor) to a concept in the SRF. We propose to reference the terms used within the sensor descriptions to the SWEET (Semantic Web for Earth and Environmental Terminology) ontologies [43]. SWEET ontologies contain categories such as *realm*, which corresponds to the *application domain* and includes the sub-categories *ocean*, *atmosphere*, etc. SWEET also includes an ontology for *observations*, *phenomena*, *human activities*, and *processes* (which encompasses the concepts falling under *intended application*) and temporal concepts. Similarly, places are referenced to GeoNames [23], which is a geographical dataset that contains over 8 million geographical names. In GeoNames, location names are associated with coordinates as well as with a type of place (building, city, etc.). Places in GeoNames are also linked by inclusion relations.

### 3.2 Representing dynamic variability context

To represent the variability of sensor context and the dependencies between context elements, we employ Horn-like rules that relate two or more context elements of a given observation set. A rule that formalizes the association between context elements is a logical implication of the form:

$$\text{Head} \rightarrow \text{Body}.$$

The head is also called the antecedent of the rule, and the body the consequent of the rule. The rule expresses a logical implication, i.e., it indicates that if the head is verified, then the body is also verified. For instance, a context rule written in natural language could be:

*intended use is assessing interior air quality*  $\rightarrow$   
*area of measurement of sensor observation is "inside building"*  
*intended use is assessing outside air quality*  $\rightarrow$   
*area of measurement of sensor observation is "outside building"*

However, for rules to be processed by reasoning engines, they cannot be expressed in natural language; they must be expressed with formal syntax and semantics. We express rules with the syntax of the Semantic Web Rule Language (SWRL), which expresses rules in terms of OWL concepts [28]. In this formalism, the body and head of the rule are formed with atoms. Atoms are basic rule elements that can be of the following forms:

1. **Concept atoms:**  $c(x)$  means that individual  $x$  is an instance of concept  $c$ ; for example,  $Obs(x)$  means that  $x$  represents a set of sensor observations, while  $PhenomenonObserved(Air)$  means that "Air" is the value of  $PhenomenonObserved$ .
2. **Property atoms:**  $p(x, z)$  means that the value of property  $p$  for individual  $x$  is  $z$ . (e.g.,  $inside(AreaOfMeasurement, building)$ ).

Using this formalism, the above-mentioned rules for a sensor  $s$  write as follows:

$$\begin{aligned} & \text{Sensor}(s) \wedge \text{IntendedApplication}(s, \text{assess interior air quality}) \rightarrow \\ & \quad \text{Sensor}(s) \wedge \text{AreaOfMeasurement}(s, a) \wedge \text{inside}(a, \text{building}) \\ & \text{Sensor}(s) \wedge \text{IntendedApplication}(s, \text{assess outside air quality}) \rightarrow \\ & \quad \text{Sensor}(s) \wedge \text{AreaOfMeasurement}(s, a) \wedge \text{inside}(a, \text{building}). \end{aligned}$$

Such rules allow expressing the dynamic nature of the context, i.e., when a first context element value is modified, another context element must be modified accordingly. According to [55], context can be acquired using at least two different approaches: explicitly, when the user manually specifies the context-related information; or implicitly, when technologies such as sensors are used to gather context information. However, it is out of the scope of this paper to address the issue of context acquisition.

As explained by Zander and Schandl [55], the represented context is not an accurate and comprehensive representation of the surrounding real world context, but an abstraction in which the degree of accuracy varies and which may focus on different aspects of reality, depending on factors such as the users' intentions and perception of the real world. Accordingly, it is likely that multiple representations of the same situation, differing in the level of precision and the aspects they focus on, may coexist. To resolve the resulting context heterogeneity, we propose the following semantic mediation system, which is based on the presented sensor metadata model for representing context and context variability rules.

## 4 Context-aware semantic mediation system

The contribution of this context-aware semantic mediation system is to propose a system that has the capacity to process context descriptions of sensor observations; to provide a more accurate mapping result that will support sensor data discovery, retrieval, and integration; and to integrate the dynamic nature of sensor context. The system also goes beyond existing semantic mediation systems that are dedicated to static semantics, by introducing the impact of dynamic context on the semantic mapping process. The dynamic context plays a dual role: reflecting the dynamic nature of the environment and situation within which sensor observations are made, and providing context-dependent semantic mappings.

In this section, we present the semantic mapping system and its core components. Figure 2 illustrates its architecture. The system is composed of the following components: 1) a context acquisition component; 2) a user interface that allows the user to specify requested sensor observations, to manage the semantic mappings of different sensor observation sets, and to visualize the context-dependent semantic mappings; 3) a light mapping component; and 4) a complex mapping component. The context acquisition component is responsible for gathering the relevant information on the context of sensor observations and instantiating the context model to populate the context base with context descriptions. As indicated above, in this paper, we do not address the problem of context acquisition, which is a complex issue, but only indicate that context could be user-defined or automatically produced with context-aware system approaches. The semantic mapping process is divided in two main phases: a light mapping phase and a complex mapping phase. The light mapping

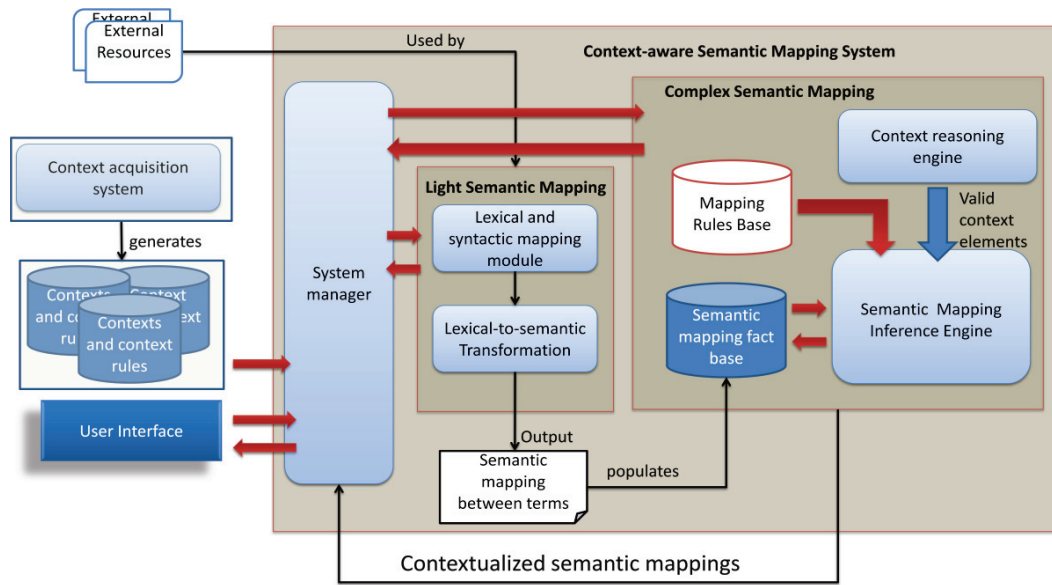


Figure 2: Architecture of the dynamic context-aware semantic mediation system.

component's role is to compare the individual terms (words) used in the compared context descriptions using external, lexical resources, and syntactic techniques. During this phase, the structure of the context descriptions is not considered. The complex mapping component's role is to compare the constructs that form the compared context descriptions, and which are composed of the previously-mentioned terms, using semantic-web reasoning techniques. The system's final output is a set of context-dependent semantic mappings between sensor observation sets. The user can modify the value of context elements and visualize the impact on the semantic mappings.

#### 4.1 Semantic relations produced by semantic mapping system

The principle underlying the proposed semantic mapping system is based on set theory, where each class definition is the intentional definition of a set of objects: that is, the real world objects that can be classified as instances of the class. Therefore, the meaning of the semantic expression "two classes are overlapping" is that these two classes can share a common set of instances. This is a strict condition, since two classes that share several properties but for which one of the common properties have disjoint ranges of values would be considered as non-overlapping (disjoint). For example, if a class C1 is "a river which width is between 7 and 10 m" and a class C2 is "a river which width is between 11 and 20 m," C1 and C2 would be considered as disjoint classes according to this definition. While it is true that no rivers can be an instance of both C1 and C2, in several cases those classes would be considered as close enough in meaning, especially if the user does not need to consider a spatial constraint on the width of the rivers. Consequently, we consider that classes that cannot share any instance but that have at least one non-disjoint element are "weak-overlapping," while pairs of classes that verify the strict condition are "strong-

overlapping.” Table 1 shows the meaning of each possible semantic relation that the semantic mapping system can infer between classes and the context descriptions formed by these classes.

EQUIVALENT (C1, C2)	INCLUDES (C1, C2)	INCLUDED IN (C1, C2)	STRONG OVERLAP (C1, C2)	WEAK OVERLAP (C1, C2)	COMPATIBLE (C1, C2)	DISJOINT (C1, C2)
All instances of C1 are instances of C2, and all instances of C2 are instances of C1	All instances of C2 are instances of C1, but not all instances of C1 are instances of C2	All instances of C1 are instances of C2, but not all instances of C2 are instances of C1	Some but not all instances of C2 are instances of C1, and some but not all instances of C1 are instances of C2	No instances of C2 are instances of C1, and no instances of C1 are instances of C2, but C1 and C2 have some common elements	No instances of C2 are instances of C1, and no instances of C1 are instances of C2, but C1 and C2’s names are synonyms (applies only to properties and classes, which are the only elements to have a name)	No instances of C2 are instances of C1, and no instances of C1 are instances of C2

Table 1: Interpretation of semantic relations

The semantic mapping procedure is performed by the light semantic mapping component (LSMC) and the complex semantic mapping component (CSMC).

## 4.2 Light semantic mapping component

The role of the light semantic mapping component (LSMC) is to compute semantic relations between the terms used in different context descriptions. Terms include any label such as the name of a concept, and names of properties and relations. Several semantic mapping approaches also include a similar matching phase, which is often called pre-processing, for example in S-Match [24]. In addition, several of these approaches rely on external resources (for example, WordNet, an application-independent terminological database for the English language) to find lexical relations between terms. However, some comprehensive and application-independent terminological databases such as WordNet provide more than one meaning for an entry term, depending on the context. For example, a “stream” could be a stream of data, or a type of watercourse. Therefore, the innovation of the LSMC, with respect to existing approaches, is to propose a solution to help identify the proper meaning that corresponds to a term taken from a context description. To do so, we propose to use the other terms that are part of the same context description to identify the appropriate, intended meaning of the term.

Also, another contribution is to use not only a single external resource, such as WordNet, but also other types of external resources that are suitable for the terms denoting spatial and temporal concepts. We refer to these external resources as “global ontologies,” as they are formalizations of conceptualizations that are intended to be independent from any application domain. In the current implementation, we have used, in addition to WordNet, two global ontologies, which will be presented below: the OpenCyc spatial properties and relations ontology, as well as the OpenCyc temporal ontology. Meanwhile, this choice does not exclude eventually integrating more ontologies in our system to maximize the mapping results.

The light semantic mapping process' principle is to find, in the appropriate global ontology, the concepts that correspond to the terms employed in the compared context descriptions, in order to make the terms semantically comparable. Because global ontologies have formalized vocabularies and semantics, they allow the specification of the meaning of terms within a common semantic reference framework. The light semantic mapping process includes the following steps:

1. *Parsing*: First, the terms that are composed of multiple words are decomposed with a text parser into a list of words, which will be processed separately before being recombined.
2. *Normalization*: Each resulting word is transformed into a basic, normalized form, through a process called lemmatization. For example, the lemmatization of "lakes" results in "lake." The purpose of this step is to enable the recognition of the word in the global ontology. Note that step 1 and 2 are performed for all terms in a context description before moving on to step 3. This is because normalized words will be used in step 4.
3. *Querying the global ontology*: Each resulting normalized word is used to query the appropriate global ontology, in order to retrieve the matching global ontology's term (with its set of meanings). To select the appropriate global ontology (either WordNet, OpenCyc spatial properties and relations ontology or OpenCyc temporal ontology), we use tags associated with each context element in the context description. At the context-modeling stage, each context element is associated with a tag which value can be "spatial," "temporal," or "thematic." For terms tagged as "thematic," we select WordNet as the global ontology; for those tagged as "spatial," we select OpenCyc spatial properties and relations ontology; for those tagged as "temporal," we select OpenCyc temporal ontology.
4. *Selection of the appropriate meaning*: The term retrieved in step 3 might be associated with several meanings. In WordNet, for example, each meaning is described with one or more sentences, which we hereafter refer to as "the meaning definition." As another example, in OpenCyc spatial properties and relations ontology, the term "above" can have different meanings illustrated in Figure 3. To select the meaning that matches the normalized word, we employ the terms used in the context description in the following manner: for all normalized words of a context description (which were produced in step 2), we attempt to match them with the terms contained in each meaning definition. The meaning with the definition that contains the greatest number of matched words is selected as the appropriate meaning. Of note is that in the current implementation, the LSMC was able to automatically retrieve the appropriate meaning in approximately 60 percent of cases, which means that human intervention is required to validate the automated selection of the appropriate meaning. This is partly due, on the one hand, to the presence in the meaning definitions of several commonly used words. On the other hand, it is also due to context descriptions being composed of words that significantly differ from words used in WordNet meaning definitions. Further research on linguistic context recognition is required to improve the autonomy of the LSMC.
5. *Creation of semantic annotations*: When the appropriate meaning corresponding to a term of a context description is determined, a semantic annotation is created between the term and the meaning. A semantic annotation is a binary relation  $a(El, (c, m))$  between an element  $El$  of a semantic description (usually a database schema, in this

- case, a context description) and the meaning  $m$  of an ontology's concept  $c$ . The purpose of semantic annotations is to formally identify the meaning of  $El$ .
6. *Computation of the relations*: Once semantic annotations have been created for each term being part of the two compared context descriptions, we aim to find the relations between each pairs of terms of the same type (whether they are part of a spatial, temporal, situational, functional, or classification context element). To do so, we retrieve the relation between the meanings  $m$  of the terms and transpose this relation between the terms.
  7. *Transformation of the relations*: The relations issued in step 6 can be of different kinds, depending on the global ontology being used. For example, to compare thematic types of terms, we use WordNet, which issues synonym, hypernymy, and hyponym relations. However, the OpenCyc spatial properties and relations ontology issues relations such as "specialization of" and "generalization of." The semantic relations issued by the basic mapping component must be uniform, since they will be reused by the complex mapping inference engine. Therefore, we provide a transformation function that establishes correspondences between the types of relations contained in global ontologies and the semantic relations defined in Table 1. For example, the transformation of the lexical relations in WordNet is defined in Table 2.

Transformation rules
$t(x) \wedge t(y) \wedge \text{synonym}(x, y) \Rightarrow \text{equivalent}(x, y)$
$t(x) \wedge t(y) \wedge \text{hypernym}(x, y) \Rightarrow \text{includes}(x, y)$
$t(x) \wedge t(y) \wedge \text{hyponym}(x, y) \Rightarrow \text{included in}(x, y)$
$t(x) \wedge t(y) \wedge \text{disjoint}(x, y) \Rightarrow \text{disjoint}(x, y)$

Table 2: Transformation rules.

This transformation is based on the one proposed by Serafini et al. [49]. It relies on the principle that lexical relations have set-theoretic implications. Assuming that a term has an extensional definition (i.e., the set of real world objects that it represents), then a first term  $x$  (e.g., water body) which is an hypernym of a second term  $y$  (e.g., lake) includes  $y$ , since all objects that  $y$  represents can also be classified as instances of  $x$ .

Many semantic mapping approaches have focused on the thematic aspects of the compared concepts, leaving aside the complexities of their spatial and temporal aspects. Therefore, they use only general external resources, such as WordNet, which are not suitable to retrieve semantic relations between names of spatial elements. For example, WordNet does contain the term "above," but does not distinguish between "above-directly," "above-touching," or "above-higher" (Figure 3). For instance, an antenna can be "above-touching" a building; a road sign can be "above-overhead" a street; while hazard to air navigation can be "above-higher" the ground.

The case of temporal elements raises a similar concern. While, for example, temporal relations were formally defined by Allen, many other temporal concepts are employed to describe geospatial concepts' temporal elements. For example, there exist different meanings of the temporal relation "temporal bound intersect," such as "temporally intersect," "ends during," and "starts during." When the relation "temporal bound intersect" is verified between two time intervals, it could mean that the time intervals are intersecting ("temporally intersect"); that the end of the first time interval happens during the second time interval



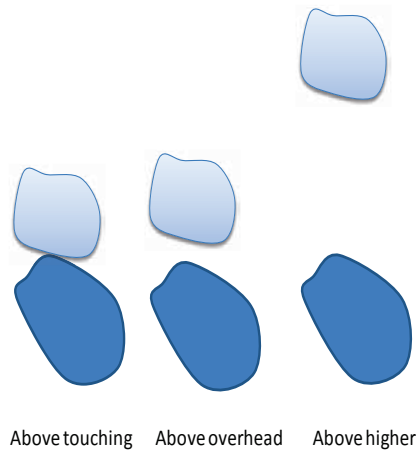


Figure 3: Different meanings of the spatial relation “above.”

(“ends during”); that the beginning of the first time interval happens during the second time interval (“starts during”); and so on.

Therefore, the LSMC must rely on comprehensive global ontologies of spatial and temporal concepts. Ontologies of spatial (or temporal) concepts describe spatial (or temporal) concepts in general, such as shapes, spatial relations, and temporal relations, regardless of the application domain. For the purpose of this approach, we used as global spatial ontology the OpenCyc spatial properties and relations ontology, as well as the temporal component of OpenCyc. However, we note that the principle of our approach is, from the perspective of the methodology, independent of the chosen external resource.

### 4.3 Complex semantic mapping component

The function of the complex semantic mapping component (CSMC) is to compute semantic relations between complex constructs of context descriptions. In this paper, a complex construct is a term employed to designate any structure that is part of a context description and that is composed of two or more terms, including context elements that are represented as OWL properties or OWL classes.

The contribution of the CSMC is to use more advanced semantic reasoning tools to infer semantic mappings, as existing semantic mapping tools rely only on subsumption reasoning [40] or similarity-based reasoning [30]. As such, we use rule-based reasoning, which allows finding more than subsumption relations, but also relations included in Table 1. More specifically, a rule-based inference system takes as input facts from a fact base, and produces new facts inferred by rules (which are stored in a rule base) in a recursive manner. Therefore, the CSMC is based on the following principles:

- The CSMC considers the semantic relations between terms issued by the LSMC as input facts.
- The CSMC can access a rule base that contains a set of mapping rules. A mapping rule is a Horn-like implication that expresses the condition for a semantic relation between two complex constructs to be true.

- The problem of computing the semantic relations between the complex constructs that are composed of those terms is formulated as the problem of verifying a set of mapping rules.

A semantic mapping rule consists of a mapping rule antecedent and a mapping rule consequent:

1. The mapping rule antecedent is a conjunction of rule statements that must be verified. Rule statements can be simple (composed of a single statement) or composite (composed of several statements related with logical Boolean expressions “and” ( $\wedge$ ), and “or” ( $\vee$ )). There are two different types of simple rule statements:
  - (a) Element type statements are statements about the nature of a complex construct; for example, the statement  $P(x)$  indicates that  $x$  is a property.
  - (b) Mapping statements are the affirmation of a semantic relation  $r$  between two terms declared through elements type statements, and which are of the form  $r(x, y)$ , where  $x, y$  are terms that participate to complex constructs of different context descriptions. The relation  $r$  can be derived from the LSMC or from the CSMC.
2. The mapping rule consequent is the consequence of the antecedent. It is a semantic relation  $r$  that holds between two complex constructs which participate in different context descriptions.

Table 3 provides the complete set of mapping rules that were developed for the CSMC. To construct Table 3, the definitions of the semantic relations provided in Table 1 were translated into formal rule statements expressing relations between terms and constructs. In all cases, the nature of the relation between constructs (properties, classes) is purely semantic, i.e., expressing a relationship between the intentional sets of objects that they represent. In other words, an overlap or inclusion relation between two spatial properties does not have the same meaning as the spatial relation “overlap” between two polygons. Rather it means that two classes solely defined by this property (whether spatial, temporal, or thematic) would have overlapping sets of instances. Of note is that these mapping rules, in contrast with context rules introduced in Section 3.2, are independent of the application domain.

Four sets of statements are provided to support the following comparisons:

- Comparison of properties (R1 to R7) based on the relations between their names and ranges of values.
- Comparison of mixed properties (properties composed of two properties, such as spatiotemporal properties, R8 to R13), based on the relations between the properties that compose them, where these relations are inferred with rules R1 to R7. The relations must hold between properties of the same nature, since properties with different types of ranges cannot be compared. For example, thematic properties cannot be compared to spatial properties.
- Comparison between classes (R14 to R21), based on the relations between the properties that compose them, and the relation between their names.
- Comparison between context descriptions (R22 to R26), based on the relations between the classes that compose them.

$P(x)$  means that  $x$  is a property;  $name(x, np1)$  means that  $np1$  is the name of  $x$ , and  $range(x, rp1)$  means that  $rp1$  is the range of  $x$ .  $equivalent(np1, np2)$  means that  $np1$  is equivalent to  $np2$ , and similarly for the other semantic relations. The  $\neg$  sign indicates negation. For example, the following is an example of the application of rule R4 to compare two observed properties *WaterLevel* of a context description  $C1$  and *LevelOfWater* of a context description  $C2$ :

$$\begin{aligned} & p(C1 : WaterLevel) \wedge p(C2 : LevelOfWater) \wedge name(C1 : WaterLevel, WaterLevel) \wedge \\ & name(C2 : LevelOfWater, LevelOfWater) \wedge range(C1 : WaterLevel, medium \sqcup high) \wedge \\ & range(C2 : LevelOfWater, high) \wedge equivalent(WaterLevel, LevelOfWater) \wedge \\ & includes(medium \sqcup high, high) \rightarrow \\ & includes(C1 : WaterLevel, C2 : LevelOfWater) \end{aligned}$$

The rule integrates the result computed by the LSMC, which issued that *WaterLevel* and *LevelOfWater* are equivalent in meaning (synonymous), with the statement  $equivalent(WaterLevel, LevelOfWater)$  (which applies to the terms and not to the observed properties themselves, which are defined by their name (term) and their range). In addition to this equivalence relation, the inclusion of the ranges allows to infer inclusion of *LevelOfWater* into *WaterLevel* (i.e., *LevelOfWater* is more specific in meaning than *WaterLevel*).

Table 3: Semantic mapping rules.

---

**Mapping rules for properties  $p(x), p(y)$ :**

---

R1:  $p(x) \wedge p(y) \wedge name(x, np1) \wedge name(y, np2) \wedge range(x, rp1) \wedge range(y, rp2) \wedge equivalent(np1, np2) \wedge equivalent(rp1, rp2) \Rightarrow equivalent(x, y)$

R2:  $p(x) \wedge p(y) \wedge name(x, np1) \wedge name(y, np2) \wedge range(x, rp1) \wedge range(y, rp2) \wedge [\neg disjoint(np1, np2) \vee overlap(rp1, rp2)] \Rightarrow weak\_overlap(x, y)$

R3:  $p(x) \wedge p(y) \wedge name(x, np1) \wedge name(y, np2) \wedge range(x, rp1) \wedge range(y, rp2) \wedge [\neg disjoint(np1, np2) \vee overlap(rp1, rp2)] \Rightarrow strong\_overlap(x, y)$

R4:  $p(x) \wedge p(y) \wedge name(x, np1) \wedge name(y, np2) \wedge range(x, rp1) \wedge range(y, rp2) \wedge [includes(np1, np2) \vee equivalent(np1, np2)] \wedge [includes(rp1, rp2) \vee equivalent(rp1, rp2)] \Rightarrow includes(x, y)$

R5:  $p(x) \wedge p(y) \wedge name(x, np1) \wedge name(y, np2) \wedge range(x, rp1) \wedge range(y, rp2) \wedge [included\_in(np1, np2) \vee equivalent(np1, np2)] \wedge [included\_in(rp1, rp2) \vee equivalent(rp1, rp2)] \Rightarrow included\_in(x, y)$

R6:  $p(x) \wedge p(y) \wedge name(x, np1) \wedge name(y, np2) \wedge range(x, rp1) \wedge range(y, rp2) \wedge [\neg equivalent(np1, np2) \vee disjoint(rp1, rp2)] \Rightarrow disjoint(x, y)$

R7:  $p(x) \wedge p(y) \wedge name(x, np1) \wedge name(y, np2) \wedge range(x, rp1) \wedge range(y, rp2) \wedge [equivalent(np1, np2) \vee disjoint(rp1, rp2)] \Rightarrow compatible(x, y)$

---

**Mapping of mixed properties (where variables with same index are of the same nature, spatial, temporal or thematic):**

---

R8:  $p(x1) \wedge p(x2) \wedge p(y1) \wedge p(y2) \wedge equivalent(x1, y1) \wedge equivalent(x2, y2) \Rightarrow equivalent(x1 \vee x2, y1 \vee y2)$

R9:  $p(x1) \wedge p(x2) \wedge p(y1) \wedge p(y2) \wedge [overlap(x1, y1) \vee overlap(x2, y2)] \Rightarrow weak\_overlap(x1 \vee x2, y1 \vee y2)$

R10:  $p(x1) \wedge p(x2) \wedge p(y1) \wedge p(y2) \wedge [overlap(x1, y1) \wedge overlap(x2, y2)] \Rightarrow strong\_overlap(x1 \vee x2, y1 \vee y2)$

R11:  $p(x1) \wedge p(x2) \wedge p(y1) \wedge p(y2) \wedge [includes(x1, y1) \vee equivalent(x1, y1)] \wedge [includes(x2, y2) \vee equivalent(x2, y2)] \Rightarrow includes(x1 \vee x2, y1 \vee y2)$

R12:  $p(x1) \wedge p(x2) \wedge p(y1) \wedge p(y2) \wedge [included\_in(x1, y1) \vee equivalent(x1, y1)] \wedge [included\_in(x2, y2) \vee equivalent(x2, y2)] \Rightarrow included\_in(x1 \vee x2, y1 \vee y2)$

R13:  $p(x1) \wedge p(x2) \wedge p(y1) \wedge p(y2) \wedge [disjoint(x1, y1) \vee disjoint(x2, y2)] \Rightarrow disjoint(x1 \vee x2, y1 \vee y2)$

---

**Mapping of classes (where c1, c2 are the compared classes):**

---

R14:  $\forall p, p' \in c1, \exists p', p' \in c2 | equivalent(p, p') \Rightarrow equivalent(c1, c2)$

R15:  $\forall p, p' \in c1, \exists p', p' \in c2 | [equivalent(p, p') \vee includes(p, p')] \Rightarrow includes(c1, c2)$

R16:  $\forall p', p' \in c2, \exists p, p \in c1 | [equivalent(p, p') \vee includes(p, p')] \Rightarrow included\_in(c1, c2)$

R17:  $\exists p, p \in c1, \exists p', p' \in c2 | strong\_overlap(p, p') \Rightarrow strong\_overlap(c1, c2)$

R18:  $[\forall p, p \in c1, \forall p', p' \in c2 | [disjoint(p, p') \vee weak\_overlap(p, p')]] \wedge [\exists p, p \in c1, \exists p',$

$R19: p' \in c2 | weak\_overlap(p, p')] \Rightarrow weak\_overlap(c1, c2)$

R20:  $name(c1, nc1) \wedge name(c2, nc2) \wedge [\forall p, p \in c1, \forall p', p' \in c2 | disjoint(p, p')] \wedge$

$\neg equivalent(nc1, nc2) \Rightarrow disjoint(c1, c2)$

R21:  $name(c1, nc1) \wedge name(c2, nc2) \wedge [\forall p, p \in c1, \forall p', p' \in c2 | disjoint(p, p')] \wedge$

$equivalent(nc1, nc2) \Rightarrow compatible(c1, c2)$

---

**Mapping of context descriptions (ctx1, ctx2 are the compared context descriptions):**

---

R22:  $\forall c, c \in ctx1, \exists c', c' \in ctx2 | equivalent(c, c') \Rightarrow equivalent(ctx1, ctx2)$

R23:  $\forall c, c \in ctx1, \exists c', c' \in ctx2 | [equivalent(c, c') \vee includes(c, c')] \Rightarrow includes(ctx1, ctx2)$

R24:  $\forall c', c' \in ctx2, \exists c, c \in ctx1 | [equivalent(c, c') \vee includes(c, c')] \Rightarrow included\_in(ctx1, ctx2)$

R25:  $\exists c, c \in ctx1, \exists c', c' \in ctx2 | strong\_overlap(c, c') \Rightarrow strong\_overlap(ctx1, ctx2) [\forall c, c \in ctx1, \forall c', c' \in ctx2 | [disjoint(c, c') \vee weak\_overlap(c, c')]] \wedge$

$[\exists c, c \in ctx1, \exists c', c' \in ctx2 | weak\_overlap(c, c')] \Rightarrow weak\_overlap(ctx1, ctx2)$

R26:  $\forall c, c \in ctx1, \forall c', c' \in ctx2 | disjoint(c, c') \Rightarrow disjoint(ctx1, ctx2)$

---

#### 4.4 Managing context variability during semantic mapping

As explained in Section 3 of this paper, a context element can vary when another context element's value is modified (e.g., when the phenomenon observed changed from "atmospheric pollution" to "interior pollution," the area of measurement must change from "outside" to "inside building"). This change will have an impact on the value of the semantic mapping between context descriptions. The context reasoning engine uses context rules (which are different from mapping rules) to infer the current context description based on the input values of context element. The context rules are application-dependent and must be defined by experts from the application domain. Therefore, in contrast with mapping rules, the number of context rules can vary. For the context variability rules to be valid, they need to be consistent (not contradict with each other). We are currently in the process of developing a rule consistency checker that would relieve the users of performing the consistency check manually. The rule consistency checker will be based on techniques for the automatic validation of logical consistency of ontologies.

In the current version of the proposed semantic mapping system, the user interface allows the user to select a context element value (e.g., "interior pollution"); as a result, the

context reasoning engine uses the context rules to infer the other context elements' values. To do so, the context reasoning engine uses the Jess rule engine to infer these values. Jess is a SWRL rule-based reasoning engine that automatically converts an OWL knowledge base into Jess assertions, and can infer facts based on SWRL rules [20]. The context reasoning engine then provides the semantic mapping inference engine with the current values of the contexts, and infers the semantic mappings that correspond to this context. The user can visualize the resulting semantic mappings in different contexts. This functionality is more concretely demonstrated in the following case study on the fusion of heterogeneous sensor data.

## 5 Heterogeneous multi-sensor data fusion

Sensor networks provide us with raw data obtained by sensing a large variety of real world features' measurable qualities. However, it is usually the information derived from sensor data that is valuable and that can be relied upon to make decisions. In addition, to obtain relevant information, it is often necessary to fuse data produced by several sensors. Simple sensor data streams often can only be used to answer simple queries, such as "what is the air temperature at  $x$ ." However, simple sensor data streams are not sufficient to answer complex queries, such as queries that require aggregation of similar data over a given area, or queries that require combination of different data to infer events or higher-level facts.

Several data fusion approaches assume that the user knows which streams can be fused in order to perform a given task. However, this assumption is neither suitable when large volumes of sensor data are available, nor when the set of available sensors is dynamic. Therefore, in this context, semantics can help to give explicit meaning to sensor data. Semantics can then be used to identify sensor observations that can be fused in a meaningful fashion. More specifically, the semantic mediation system proposed in this paper aims at reasoning with dynamic and variable contexts of sensor observations to identify the observations that can be fused in a given context.

In the following, we present a use-case scenario of sensor data fusion for environmental monitoring that demonstrates the features and contributions of the semantic mediation system. The use case scenario is representative of different application domains where sensor data is being used. Consider the problem of monitoring the presence of hazardous substances in human-inhabited areas. The areas of interest are covered by a network of heterogeneous sensors, including, for instance, mobile cameras linked to processing services equipped with image recognition capacity in order to identify gas plumes, as well as sensing devices that can detect toxic substances and their concentration. In order to identify toxic gas plumes, observations produced by mobile cameras and toxic gas monitoring devices need to be fused, i.e., the camera can identify the shape and density of the gas plume, while the sensing device can identify the toxicity. Then, security services can identify whether the situation is critical and choose the appropriate intervention on the field. In the following, we use SWRL to model this situation. The sensor services are identified with variables  $S1$  and  $S2$ , as instances of the class *SensorService*. Output values of sensor services are identified with variables  $x1$ ,  $x2$ , and the areas of measurement of sensors (considered as spatial context elements) are identified with variables  $a1$ ,  $a2$ . The following SWRL rule indicates when an alert for toxic gas plume detection should be released:

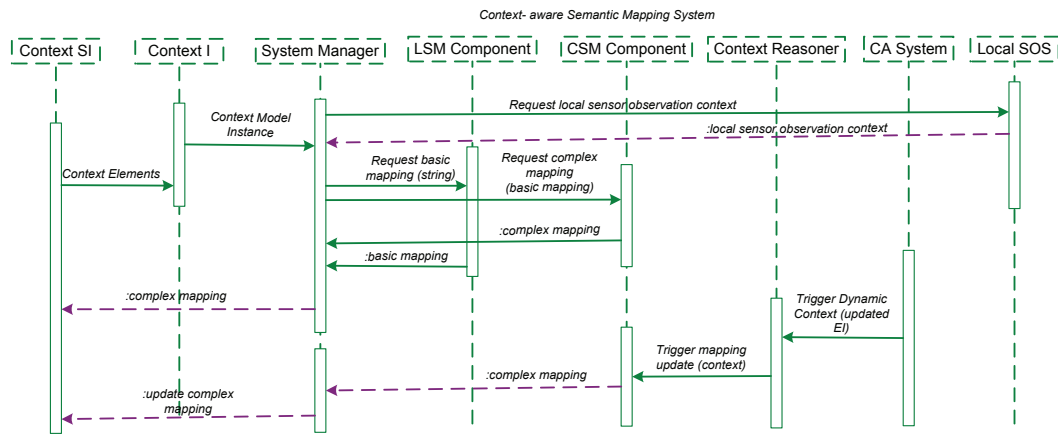


Figure 4: Sequence diagram of the system.

$$\begin{aligned}
 & \text{SensorService}(?S1) \wedge \text{Density}(?x1) \wedge \text{OutputOf}(S1, x1) \wedge \text{SensorService}(?S2) \wedge \\
 & \quad \text{Toxicity}(?x2) \wedge \text{OutputOf}(S2, x2) \wedge \text{AreaOfMeasurement}(a1) \wedge \\
 & \quad \text{HasAreaOfMeasurement}(S1, a1) \wedge \text{AreaOfMeasurement}(a2) \wedge \\
 & \quad \text{HasAreaOfMeasurement}(S2, a2) \wedge \text{SpatialOverlap}(a1, a2) \wedge \\
 & \quad \text{MoreThan}(x1, \text{DensityThreshold}) \wedge \text{MoreThan}(x2, \text{ToxicityThreshold}) \rightarrow \\
 & \quad \text{ToxicGasPlumeAlert}(?y)
 \end{aligned}$$

The atom *ToxicGasPlumeAlert* (?y) triggers the creation of an instance *y* of the class *ToxicGasPlumeAlert*.

In order to process such a semantic rule, the monitoring system that assists security services in their task of identifying hazards due to toxic gas plumes would have to browse the network of heterogeneous sensors to identify those that capture the relevant observations on density and toxicity of gases. The semantic mapping system that we have proposed was designed to support such task. Figure 4 shows the sequence diagram of the semantic mapping system in the described environment.

The sequence is initiated by the user, who specifies the context of the required sensor observations, through an interface where the template of the context model can be reproduced (Figure 5). The user can select the types of context elements (among possible context elements represented in the sensor metadata model of Figure 1) for which he or she wishes to specify a value. For example, in Figure 5, the user has specified values for “sensor observation” and “phenomenon observed.” The sensor context specification interface allows the user to easily specify the elements of the query without knowing how to use the SQWRL language.

The context specification interface (Context SI in Figure 4) sends the specified context elements to the query formalization module (ContextI on Figure 4), which transforms the context specified by the user into a SQWRL query, for example:



Use this interface to define the context of your query.

**SELECTION OF CONTEXT ELEMENTS**

What type of sensor observation are you looking for?

Specify the type of phenomenon observed:

**Add selected type of context element**

CONTEXT ELEMENT TYPE	CONTEXT ELEMENT SUB-TYPE
spatial context element	location of phenomenon observed
spatial context element	area of measurement
spatial context element	feature at proximity
situational context element	characteristic of the environment
situational context element	quality of observed feature

<Back    Next>    Cancel

Figure 5: Sensor context specification interface.

$$\begin{aligned}
 & \text{SensorService}(?s) \wedge \text{SpatialCtxEl} : \text{AreaOfMeasurement}(?a) \wedge \text{IsAreaOfMeasurement}(?a, ?s) \wedge \\
 & \text{Building}(?o) \wedge \text{Inside}(?a, ?o) \wedge \text{SensorObservation}(?obs) \wedge \text{ProducedBy}(?obs, ?s) \wedge \\
 & \text{SameAs}(?obs, \text{Toxicity}) \wedge \text{FeatureCtxEl} : \text{PhenomenonObserved}(?p, ?s) \wedge \\
 & \text{SameAs}(?p, \text{Gas}) \wedge \text{FunctionalCtxEl} : \text{IntendedUse}(?i) \wedge \text{IsIntendedUse}(?i, ?s) \wedge \\
 & \text{SameAs}(?i, \text{DetectHealthHazard}) \wedge \text{ClassificationCtxEl} : \text{ClassOfObservation}(?c) \wedge \\
 & \text{IsClassOfObservation}(?c, ?s) \wedge \text{SameAs}(?c, \text{QualityOfSubstance}) \rightarrow \\
 & \text{sqwrl} : \text{select}(?s)
 \end{aligned}$$

When the Query Formalization Module receives the user-selected values for context elements, it tags each context element with the associated type of context (spatial, temporal, functional, etc.) and generates an atom of the form:

$$\text{TypeOfContextElement} : \text{ContextElement}(?a).$$

The type of context element associated with each context element is predefined in the system. For example, *SpatialCtxEl : AreaOfMeasurement(?a)* is an atom generated by the query formalization module, where the user had selected the spatial context element *AreaofMeasurement* and specified the value *a* for this context element. In addition, for each context element value specified by the user, the query formalization module generates a property atom that links the context element value to the researched sensor service (e.g., *IsAreaOfMeasurement(?a, ?s)*) using predefined properties (e.g., *IsAreaOfMeasurement*, associated with the element *AreaOfMeasurement*, and *ProducedBy*, associated with the element *Producer*). Predefined built-in operators already formalized within the SQWRL language are also used to generate the query (e.g., *SameAs*). Once all atoms are generated from values specified by the user, the query formalization module assembles the atoms into a single query and generates the consequent, which is the same for all queries (i.e., *sqwrl : select(?s)*).

The SQWRL query produced by the query formalization module is forwarded to the system manager, which is responsible for managing the semantic mapping process. Firstly, the system manager requests local sensor observation contexts from available SOS. Upon receiving these contexts, the system manager extracts individual terms from the contexts and from the query. The terms are forwarded to the LSMC (LSM Component in Figure 4), which performs the light semantic mapping phase. The semantic relations produced by the LSMC are stored in a mapping fact base as instances of the OWL class *LightMapping*, with OWL object properties *HasSourceTerm* and *HasTargetTerm*, and OWL datatype property *StatesRelation*. Figure 6 shows the UML representation of the model that we have developed for implementing the mapping fact base.

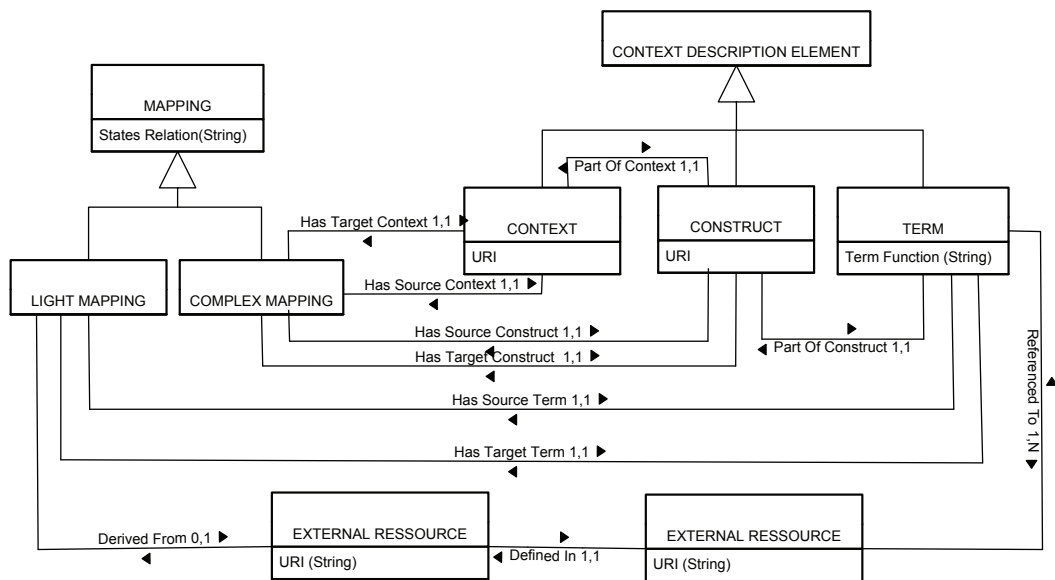


Figure 6: UML model of the mapping fact base.

The system manager then requests the CSMC (see Figure 4) to compute the complex mappings, based on the light mappings stored in the mapping fact base. The CSMC processes the mapping rules stored in the mapping rule base against the facts stored in the

mapping fact base, and it infers the semantic relations between the context elements of the query and corresponding context elements of sensor observations. Similarly to light semantic mappings, the semantic relations produced by the CSMC are stored in the mapping fact base as instances of the OWL class *ComplexMapping*, with OWL object properties *CM:Relation*, *CM:Source*, and *CM:Target*.

The semantic mappings between the query's context elements and the sensor observation's context elements are added to the mapping fact base. The CSMC automatically infers the semantic relations between the query's and sensor observations' global contexts using these new facts.

In the following, we provide two scenarios that continue the above case study: a scenario where the context of sensor observations is static, and a scenario where the context of sensor observations is dynamic. We highlight the possibilities that the proposed context-aware semantic mediation system allows for, and also the challenges that are raised when attempting to deal with dynamic semantics.

## 5.1 Static context scenario

In the static context scenario, we consider that sensors are static and that the value of context elements remains unchanged in time. Figure 7 shows the interface where the user can visualize the results of the query (i.e., the retrieved sensor services and their respective relation with the query) when sensor observation contexts are static.

The screenshot displays the 'QUERY RESULTS' application window, which is divided into several sections:

- QUERY:** A table mapping context element sub-types to their values.
 

CONTEXT ELEMENT SUB-TYPE	CONTEXT ELEMENT VALUE
Area of measurement	Inside(Building)
Sensor observation	Toxicity
Phenomenon observed	Gas
Intended use	Detect health hazard
Class of observation	Quality of substance
- QUERY RESULTS:** A tree view showing the hierarchy of retrieved sensor observations, including categories like 'EQUIVALENT TO Query', 'INCLUDING Query', and 'STRONG OVERLAP WITH Query'. A specific URI is highlighted: `http://app.semsensorproject.ca:20112/sos`.
- CONTEXT DESCRIPTION:** A text area showing the selected sensor observation's context description, including declarations, class assertions, and object property assertions from the 'SampleOntology'.
- SENSOR DATA FUSION:** A section containing a 'SENSOR DATA FUSION RULE' (a complex logical expression involving SensorService, Density, OutputOf, AreaOfMeasurement, SpatialCtEI, and MoreThan) and a 'Select retrieved sensor observations as input for the sensor fusion rule' area. This area contains two selected observations:
  - S1:** `http://app.semsensorproject.ca:20545/sos`
  - S2:** `http://app.semsensorproject.ca:20112/sos`

Figure 7: Visualization of results in a static context scenario.

Query results are communicated to the user in the form of a hierarchy where the retrieved, relevant sensor services are classified according to their semantic relation with the requested context, displayed on left-hand side of Figure 7. The context description of the retrieved sensor services is also displayed, so that the user can interpret the results to select the most suitable sensor observations. For example, the context-aware semantic mapping system has inferred that sensor observations on “cytotoxicity” provided by a retrieved sensor service (identified, for the purpose of this implementation, as <http://app.semsensorproject.ca:20344/sos>) are weakly overlapping the query. The context is determinant in establishing the relevance of sensor observations, given that even if “cytotoxicity” is semantically included in “toxicity,” the area of measurement is “inside greenhouse,” and intended use is “detect hazard for plants.” Therefore, the context of sensor observations is crucial to discard related but irrelevant sensor observations.

In addition, the related fusion rule is also displayed (right-hand side of Figure 7). In the example, the fusion rule contains variables (e.g., S1 and S2) that correspond to sensor services we are looking for. The user can easily browse the semantic mapping results (left-hand side of Figure 7) to select the sensor service(s) that will replace S1 and S2 in the fusion rule (bottom-right side of Figure 7). In the above example, the semantic mapping query was meant to find sensor services that could replace S2 (to measure toxicity). As a result, the toxic gas plume alert could be triggered when the output of the user-selected sensor services satisfies the sensor data fusion rule. This approach represents a seamless way for users to gather and combine data coming from various and heterogeneous sensor services. The limitations of the static scenario, however, arise when we consider that the context of sensors observations is likely to be dynamic (e.g., the sensor is no longer “within building,” but “outside building”).

## 5.2 Dynamic context scenario

In a dynamic context scenario, context element values can be modified (e.g., through a context-aware system which issues notifications, or following users’ or agents’ instructions). The impact of the modification of a context element’s value is reflected with a SWRL rule, which formalizes the dependencies between context elements. The context-aware semantic mapping system maintains a dynamic context rule repository that stores the context variability rules and enables the user to edit the rules of a given sensor observation service (top of Figure 8). For instance, a rule that relates the *AreaOfMeasurement* (e.g., “outsideBuilding”) to the *IntendedUse* of sensor data (e.g., “AssessExteriorPollution”) for a sensor service (identified, for the purpose of this implementation, as <http://app.semsensorproject.ca:20112/sos>) that was part of the query results on Figure 7. In the static scenario, the semantic relation of this sensor service with the query was *includes*, meaning that the sensor service output was more general than the query. When the spatial context of the sensor observations provided by this service is modified from “insideBuilding” to “outsideBuilding,” the context reasoner infers the values of other related context elements (e.g., *IntendedUse*), as per the rules stored in the dynamic context rule repository (Figure 8).

As a result, the semantic mappings are updated, reflecting a new semantic relation with the current query (“weak overlap”), which affects the relevance of the service for the user. The affected sensor service is reclassified in the query results hierarchy (bottom of Figure 8). As another example, when the *AreaOfMeasurement* is changed from “insideBuilding” to “outsideBuilding,” the types of gases monitored can be modified because when assessing

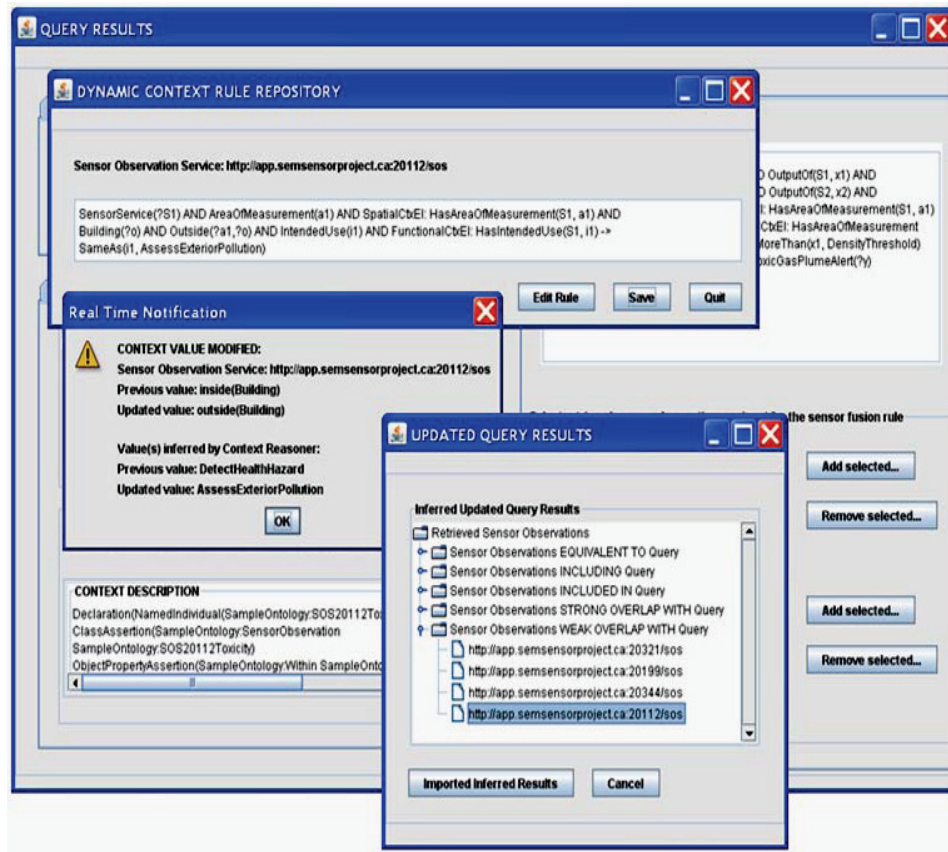


Figure 8: Visualization of results in a dynamic context setting.

outdoor pollution, health authorities do not focus on the same substances as when assessing indoor pollution. The following simplified rules express the dependency between the area of measurement and the type of gas which density is being measured (sensors can monitor more than one type of gas):

$$Sensor(s) \wedge AreaOfMeasurement(s, a) \wedge outside(a, building) \rightarrow \\ Sensor(s) \wedge PhenomenonObserved(s, carbon\ dioxide)$$

$$Sensor(s) \wedge AreaOfMeasurement(s, a) \wedge inside(a, building) \rightarrow \\ Sensor(s) \wedge PhenomenonObserved(s, carbon\ monoxide)$$

The notification of a change of a context element's value can be automated with real time context acquisition system. However, this requires further research to automate the semantic annotation of real world events that can be detected by sensors to the application

ontology employed to describe dynamic context rules and sensor observations contexts. Notably, to extend the current context-awareness capabilities of the semantic mediation service, we are currently investigating the use of Open Geospatial Consortium (OGC)'s Sensor Web Enablement (SWE) standard service interfaces to detect events from sensor data. More specifically, we are developing a context acquisition service based on the Sensor Event Service (SES). The SES is a proposed SWE standard that allows users or applications to subscribe to events of interest captured by sensors [9]. The events of interest are specified via constraints on the values observed by sensors. The SES is based on a publish-subscribe system: it monitors the observations produced by registered sensors and notifies users or applications when the subscribed event occurs. In the above example, the SES would notify the Context Reasoner of the Semantic Mediation System that the area of measurement of a sensor has changed from "insideBuilding" to "outsideBuilding." In this case, it means that the area of measurement is also an observed property of the sensor, since the SES monitors changes in the value of observed properties. However, it is not a constraint of our system that a context element can also be an observed property (either by the sensor itself or by another related sensor).

## 6 Conclusions and perspectives

This paper has addressed the challenges related to semantic interoperability in the sensor web, and more specifically, the issue of finding semantic mappings for sensor data. We focused on the need for providing comprehensive semantics describing the context of sensor observations and their context, and we have provided a metadata model accordingly. The proposed metadata model is adapted to sensor observations and we argue that such a model is needed to complement the SensorML standard, the loose structure of which is not sufficient to support reasoning with multiple sensor descriptions and resolve semantic heterogeneities. We also focused on the challenges posed by dynamic metadata, i.e., the fact that sensor observation context is likely to evolve in real time, especially for mobile sensors. We have described how SWRL rules can be used to represent and deal with the dynamic nature of context. Then, we have described how the dynamic context model can serve as a basis for context-aware computation of semantic mappings to reconcile heterogeneous sensor observations, using semantic web technologies to support rule-based reasoning. Finally, we have demonstrated how the proposed context-aware semantic mediation system is useful to support fusion of sensor data produced by multiple sensors through fusion rules. This use-case scenario was intended to illustrate one of the possible uses of the proposed system. However, we argue that the system is also useful to support other semantic interoperability tasks, such as query propagation in heterogeneous sensor networks. As such, this work aims at contributing to the long term vision of the semantic sensor web, by also setting the focus on dynamic semantics as a new but increasingly widespread type of semantics that future systems will have to deal with.

This research has revealed several additional issues that call for future work to further improve the proposed system. Firstly, we note that while end-users of the system can use a simple context specification interface to submit their query (without required knowledge on the SQWRL language), expert users are needed to formulate the rules on the dynamic context relevant to their application domain. This requires experts to be familiar with the SWRL language. Therefore, we are currently planning to extend the system with a simpli-



fied rule editor. Such rule editor would enable to create rules without knowledge of the underlying language but also to verify the syntactical validity of the generated rules, as well as contradictions among specified rules. The same rule editor will be employed to improve the visualization of fusion rules. A second challenge noted during this research is the identification by the LSMC, among the several possible meanings of a term provided by an external resource (lexicon of global ontology), of the more appropriate meaning. The implementation has demonstrated that human intervention is required to validate the automated selection of the appropriate meaning. It was concluded that further research on linguistic context recognition is required to improve the autonomy of the LSMC. Eventually, more external resources (a larger variety of global ontologies) would be required to improve the ability of the system to identify the meaning of a large variety of terms.

Another of the main challenges raised in this work is related to the issue of matching real world features and events captured by sensors to ontological models representing sensor observation contexts. In this perspective, human intervention [10] and a rule-based strategy [36] are proposed as two of the solutions to support this type of annotation. As future work, we are also investigating how knowledge extraction techniques can be integrated into the system to support this task, and the theoretical and technological challenges that it raises. Among these challenges, we highlight the need to develop and integrate integrity constraints specific to sensor data into the proposed metadata model, and to develop reasoning languages to deal with these constraints during interoperability processes among sensor data.

## Acknowledgments

This research was made possible by an operating grant from Microsoft Research and Alberta Innovate Technology Future. We also wish to thank the anonymous reviewers for their valuable comments which contributed to the improvement of this paper.

## References

- [1] AGARWAL, P. Ontological considerations in GIScience. *International Journal of Geographical Information Science* 19, 5 (2005), 501–536. doi:10.1080/13658810500032321.
- [2] ANAGNOSTOPOULOS, C., TSOUNIS, A., AND HADJIEFTHYMIADES, S. Context awareness in mobile computing environments: A survey. In *Mobile e-Conference, Information Society Technologies* (2004).
- [3] ARBONA, A., BENKNER, S., ENGELBRECHT, G., FINGBERG, J., HOFMANN, M., KUMPF, K., LONSDALE, G., AND WOEHRER, A. A service-oriented grid infrastructure for biomedical data and compute services. *IEEE Transactions on NanoBioscience* 6, 2 (2007), 136–141. doi:10.1109/TNB.2007.897438.
- [4] BAKILLAH, M., AND LIANG, S. Discovering sensor services with social network analysis and expanded SQWRL querying. In *2012 Web and Wireless Geographical Information Systems Conference*, S. Di Martino, A. Peron, and T. Tezuka, Eds., vol. 7236 of *Lecture Notes in Computer Science*. Springer, Berlin, April 2012, pp. 221–238. doi:10.1007/978-3-642-29247-7\_16.

- [5] BAKILLAH, M., AND MOSTAFAVI, M. G-map semantic mapping approach based on augmented geospatial service description to improve semantic interoperability of distributed geospatial web services. In *Fourth International Workshop on Semantic and Conceptual Issues in Geographic Information Systems*, J. Trujillo, G. Dobbie, H. Kangassalo, S. Hartmann, M. Kirchberg, M. Rossi, I. Reinhartz-Berger, E. Zimányi, and F. Frasin-car, Eds., vol. 6413 of *Lecture Notes in Computer Science*. Springer, Berlin, November 1–4 2010, pp. 12–22. doi:10.1007/978-3-642-16385-2\_3.
- [6] BITTNER, T., DONNELLY, M., AND WINTER, S. Ontology and semantic interoperability. In *Large Scale 3D Data Integration: Challenges and Opportunities*, S. Zlatanova and D. Proserpi, Eds. CRC Taylor and Francis, 2008.
- [7] BOTTS, M., PERCIVALL, G., REED, C., AND DAVIDSON, J. OGC sensor web enablement: overview and high level architecture. Tech. Rep. OGC 07-165r1, Open Geospatial Consortium, 2007. doi:10.1109/AUTEST.2007.4374243.
- [8] BRODEUR, J., BÉDARD, Y., EDWARDS, G., AND MOULIN, B. Revisiting the concept of geospatial data interoperability within the scope of human communication process. *Transactions in GIS* 7, 2 (2003), 243–265. doi:10.1111/1467-9671.00143.
- [9] BRÖRING, A., ECHTERHOFF, J., JIRKA, S., SIMONIS, I., EVERDING, T., STASCH, C., LIANG, S., AND LEMMENS, R. New generation sensor web enablement. *Sensors* 11 (2011), 2652–2699. doi:10.3390/s110302652.
- [10] BRÖRING, A., JANOWICZ, K., STASCH, C., AND KUHN, W. Semantic challenges for sensor plug and play. In *Web and Wireless Geographical Information Systems* (Berlin, 2009), J. Carswell, A. Fotheringham, and G. McArdle, Eds., vol. 5886 of *Lecture Notes in Computer Science*, Springer, pp. 72–86. doi:10.1007/978-3-642-10601-9\_6.
- [11] CHI, Y. Ontology-based curriculum content sequencing system with semantic rules. *Expert Systems with Applications* 36, 4 (2009), 7838–7847. doi:10.1016/j.eswa.2008.11.048.
- [12] COMPTON, M., HENSON, C., LEFORT, L., NEUHAUS, H., AND SHETH, A. A survey of semantic specifications of sensors. In *Semantic Sensor Networks* (2009), pp. 17–32.
- [13] CORCO, O., AND GARCIA-CASTRO, R. Five challenges for the semantic sensor web. *Semantic Web* 1 (2010), 121–125.
- [14] CUADRA, D., CALLE, J., AND RIVERO, J. How to manage spatio-temporal events in relational databases. *Journal of Research and Practice in Information Technology* 43, 4 (2011), 329–344.
- [15] CYCORP. OpenCyc. <http://www.opencyc.org>. Accessed 29 May 2013.
- [16] DEY, A. Understanding and using context. *Personal and Ubiquitous Computing* 5, 1 (2001), 4–7. doi:10.1007/s007790170019.
- [17] DING, L., WORKS, K., AND RUNDENSTEINER, E. Semantic stream query optimization exploiting dynamic metadata. In *IEEE 27th International Conference on Data Engineering* (11–16th April 2011). doi:10.1109/ICDE.2011.5767840.

- [18] EGENHOFER, M. Toward the semantic geospatial web. In *GIS'02* (McLean, Virginia, 8–9 November 2002). doi:10.1145/585147.585148.
- [19] EID, M., LISCANO, R., AND SADDIK, A. A universal ontology for sensor networks data. In *IEEE International Conference on Computational Intelligence for Measurement Systems and Applications* (2007). doi:10.1109/CIMSA.2007.4362539.
- [20] ERIKSSON, H. Using JessTab to integrate Protégé? and Jess. *IEEE Intelligent Systems* 18, 2 (2004), 43–50. doi:10.1109/MIS.2003.1193656.
- [21] EUZENAT, J., AND SHVAIKO, P. *Ontology Matching*. Springer, Berlin, 2007.
- [22] EUZENAT, J., AND VALTCHEV, P. Similarity-based ontology alignment in OWL-lite. In *15th European Conference on Artificial Intelligence* (2004), ECAI 2004, pp. 333–337.
- [23] GEONAMES.ORG. GeoNames. <http://www.geonames.org>. Accessed 29 May 2013.
- [24] GIUCHIGLIA, F., SHVAIKO, P., AND YATSKEVICH, M. S-Match: An algorithm and an implementation of semantic matching. In *European Semantic Web Symposium* (Berlin, 2004), vol. 3053 of *Lecture Notes in Computer Science*, Springer, pp. 61–75. doi:10.1007/978-3-540-25956-5\_5.
- [25] GOMEZ, M., PREECE, A., JOHNSON, M., DE MEL, G., VASCONCELOS, W., GIBSON, C., BAR-NOY, A., BOROWIECKI, K., PORTA, T., AND PIZZOCARO, D. An ontology-centric approach to sensor-mission assignment. In *16th International Conference on Knowledge Engineering and Knowledge Management* (2008). doi:10.1007/978-3-540-87696-0\_30.
- [26] GRUBER, T. A translation approach to portable ontology specification. Tech. Rep. KSL 92-71, Knowledge Systems Laboratory, Stanford University, California, 1993.
- [27] HERZOG, A., JACOBI, D., AND BUCHMANN, A. Predefined classification for mixed mode/environments. Tech. rep., TU Darmstadt, 2009.
- [28] HORROCKS, I., PATEL-SCHNEIDER, P., BOLEY, H., TABET, S., GROSO, F., AND DEAN, M. SWRL: A semantic web rule language combining OWL and RuleML. <http://www.w3.org/Submission/SWRL>, 2004. Accessed 29 May 2013.
- [29] HU, W., AND QU, Y. Falcon-AO: A practical ontology matching system. In *Web Semantics: Science, Services and Agents on the World Wide Web 6* (2008), pp. 237–239. doi:10.1016/j.websem.2008.02.006.
- [30] JANOWICZ, K. Similarity-based retrieval for geospatial semantic web services specified using the web service modeling language (WSML-core). In *The GeoSpatial Web* (Berlin, 2006), Springer. doi:10.1007/978-1-84628-827-2\_24.
- [31] JEONG, Y., SONG, E., CHAE, G., HONG, M., AND PARK, D. Large-scale middleware for ubiquitous sensor networks. *IEEE Intelligent Systems* 25, 2 (2010), 1541–1672. doi:10.1109/MIS.2010.52.
- [32] JIRKA, S., BRÖRING, A., AND STASCH, C. Discovery mechanisms for the sensor web. *Sensors* 9, 2 (2009), 2661–2681. doi:10.3390/s90402661.

- [33] KALFOGLOU, Y., AND SCHORLEMMER, M. Ontology mapping: The state of the art. *The Knowledge Engineering Review* 18, 1 (2003), 1–31. doi:10.1017/S0269888903000651.
- [34] KESSLER, C., RAUBAL, M., AND JANOWICZ, K. The effect of context on semantic similarity measurement. In *OTM 2007: On the Move to Meaningful Internet Systems Workshops* (2007), pp. 1274–1284. doi:10.1007/978-3-540-76890-6\_55.
- [35] KIM, J. H., KWON, H., KIM, D. H., KWAK, H. Y., AND LEE, S. J. Building a service-oriented ontology for wireless sensor networks. In *Seventh IEEE/ACIS International Conference on Computer and Information Science* (2008), ICIS 08, pp. 649–654. doi:10.1109/ICIS.2008.100.
- [36] KLIEN, E. A rule-based strategy for the semantic annotation of geodata. *Transactions in GIS* 11, 3 (2007), 437–452. doi:10.1111/j.1467-9671.2007.01054.x.
- [37] KNAPPMAYER, M., BAKER, N., LIAQUAT, S., AND TONJES, R. A context provisioning framework to support pervasive and ubiquitous applications. In *4th European Conference on Smart Sensing and Context*, EuroSSC'09. Springer, Berlin, 2009, pp. 93–106. doi:10.1007/978-3-642-04471-7\_8.
- [38] KUHN, W., AND M., R. Implementing semantic reference systems. In *6th AGILE Conference on Geographic Information Science*, M. Gould, R. Laurini, and S. Coulondre, Eds., AGILE 2003. Presses Polytechniques et Universitaires Romandes, Lyon, France, 2003, pp. 63–72.
- [39] LI, L., AND TAYLOR, K. A framework for semantic sensor network services. In *International Conference on Service Oriented Computing* (2008), ICSOC'08, pp. 347–361. doi:10.1007/978-3-540-89652-4\_27.
- [40] LUTZ, M., AND KLIEN, E. Ontology-based retrieval of geographic information. *International Journal of Geographical Information Science* 20 (2006), 233–260. doi:10.1080/13658810500287107.
- [41] MASSMANN, S., ENGMANN, D., AND RAHM, E. COMA++: Results for the ontology alignment contest OAEI 2006. In *International Workshop on Ontology Matching, collocated with the 5th International Semantic Web Conference (ISWC-2006)* (2006), vol. 225 of OM-2006, CEUR Workshop Proceedings, pp. 107–114.
- [42] MELNIK, S., GARCIA-MOLINA, H., AND RAHM, E. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *18th International Conference on Data Engineering* (2002), ICDE02, pp. 117–128. doi:10.1109/ICDE.2002.994702.
- [43] NASA JPL. Semantic Web for Earth and Environmental Terminology (SWEET). <http://sweet.jpl.nasa.gov/ontology>. Accessed 29 May 2013.
- [44] PATHAN, M., TAYLOR, K., AND COMPTON, M. Semantics-based plug-and-play configuration of sensor network services. In *3rd International Workshop on Semantic Sensor Networks (SSN)* (Shanghai, China, 2010), K. Taylor, A. Ayyagari, and D. de Roure, Eds.
- [45] PSCHORR, J., HENSON, C., PATNI, H., AND SHETH, A. Sensor discovery on linked data. Tech. rep., Knoesis Center, 2010.

- [46] RODRIGUEZ, A., AND EGENHOFER, M. Determining semantic similarity among entity classes from different ontologies. *IEEE Transactions on Knowledge and Data Engineering* 15, 2 (2003), 442–456. doi:10.1109/TKDE.2003.1185844.
- [47] RUSSOMANNO, D., KOTHARI, C., AND THOMAS, O. Sensor ontologies: From shallow to deep models. In *Thirty-Seventh Southeastern Symposium on System Theory* (2005), SSST'05, pp. 107–112. doi:10.1109/SSST.2005.1460887.
- [48] SAHA, D., AND MUKHERJEE, A. Pervasive computing: a paradigm for the 21st century. *IEEE Computer Society* 36, 3 (2003), 25–31. doi:10.1109/MC.2003.1185214.
- [49] SERAFINI, L., BOUQUET, P., MAGNINI, B., AND ZANOBINI, S. An algorithm for matching contextualized schemas via SAT. Tech. Rep. 0301-06, Istituto Trentino di Cultura, Trento, Italy, 2003.
- [50] SHETH, A., HENSON, C., AND SAHOO, S. Semantic sensor web. *IEEE Internet Computing* 12, 4 (2008), 78–83. doi:10.1109/MIC.2008.87.
- [51] TEO, H. An activity-driven model for context-awareness in mobile computing. ACM, pp. 545–546. doi:10.1145/1409240.1409342.
- [52] USCHOLD, M., CLARK, P., DICKEY, F., FUNG, C., SMITH, S., UCZEKAJ, S., WILKE, M., BECHHOFER, S., AND HORROCKS, I. A semantic infosphere. In *2003 International Semantic Web Conference* (2003), ISWC'03, pp. 882–896. doi:10.1007/978-3-540-39718-2\_56.
- [53] VAN KRANENBURG, H., BARGH, M., IACOB, S., AND PEDDEMORS, A. A context management framework for supporting context-aware distributed applications. *IEEE Communications Magazine* 44, 8 (2006), 67–74. doi:10.1109/MCOM.2006.1678112.
- [54] ZAFEIROPOULOS, A., SPANOS, D., ARKOULIS, S., KONSTANTINOU, N., AND MITOUR, N. Data management in sensor networks using semantic web technologies. In *Data Management in Semantic Web*, H. Jin and Z. Lv, Eds. Nova Science Publishers, 2009.
- [55] ZANDER, S., AND SCHANDL, B. Context-driven RDF data replication on mobile devices. *Semantic Web* 2012, 1 (1), 131–155. doi:10.3233/SW-2011-0043.