

Spatio-temporal Conceptual Schema Development for Wide-Area Sensor Networks

Mallikarjun Shankar¹, Alexandre Sorokine¹, Budhendra Bhaduri¹,
David Resseguie¹, Shashi Shekhar², and Jin Soung Yoo²

¹ Oak Ridge National Laboratory

PO Box 2008 MS 6017 Oak Ridge, TN 37831-6017

{shankarm,sorokina,bhadurib1,resseguiedr}@ornl.gov

² Department of Computer Science and Engineering, University of Minnesota

200 Union ST SE 4192S Minneapolis, MN 55414

{shekhar,jyoo}@cs.umn.edu

Abstract. A Wide-Area Sensor Network (WASN) is a collection of heterogeneous sensor networks and data repositories spread over a wide geographic area. The diversity of sensor types and the regional differences over which WASNs operate result in semantic interoperability mismatches among sensor data, and a difficulty in agreeing on methods for sensor data access and exchange. We assume that sensors and their associated data have an explicit spatio-temporal basis (or tagging) in their representation. In this paper, we describe a spatio-temporal loosely-coupled federated database model for the WASN data storage problem - that of unifying query and data representation given a heterogeneous WASN - and propose a conceptual schema to ease the problem of integration of sensor data representations. This is a continuing and critical challenge as sensor networks become more ubiquitous and data interoperation becomes increasing vital for a variety of applications (such as homeland security, transportation, environmental monitoring, etc.). We employ a top-down ontology-driven software development methodology. We use the SNAP/SPAN ontology as a sample framework for the conceptual schema. We compare our methodology of conceptual schema development with a bottom-up entity-oriented schema construction and discuss the differences in the two approaches. A unique contribution is the discussion of deployment experiences to evaluate proposed approaches in the context of a concrete WASN testbed.

1 Introduction

We address the problem of unifying sensor data representation (e.g., for queries and storage) in Wide-Area Sensor Networks (WASNs). A WASN is a collection of heterogeneous sensor networks and data repositories spread over a wide geographic area (e.g., a deployment across one or more states). We aim to find commonality in the representation of the data records and their semantic description given the heterogeneity of the constituent components. We observe that a majority of WASN-related data is explicitly or implicitly spatio-temporally tagged.

In many cases, it is easy to argue for an explicit spatial and temporal context for the WASN data, since the location and time-instant of data measurements directly affect the command and control operations within the WASN. Example domains for such sensor networks include transportation networks and CBRNE (Chemical-Biological-Radiation-Nuclear-Explosive, [1]) detection networks. Sensor networks have been a focus of much research in recent years [2,3]. Much of the research targets resource-constrained wireless sensor networks. WASNs, by contrast, are often not resource constrained. Although the concerns of power efficient computing, networking, and data access are relevant to WASNs near the edges of the network, we address the complementary requirement for WASNs: that they cover significantly larger geographic areas and incorporate a wide range of sensor and application types.

The systematic representation and storage of the spatio-temporal data related to WASNs presents difficulties because such networks typically support a large number of heterogeneous sensors (types and manufacturers) and consist of several autonomous regional domains carrying out sensor data collection and sensor actuation. The diversity of types and the distances over which the network operates results in semantic interoperability mismatches among sensor data and a difficulty in agreeing upon sensor data access and exchange formats and protocols. For example, data from the same type of sensor but from different manufacturers may disagree in data formats, or may be implemented in a way that the database stores the records differently. The difficulty arises in building common structures for data transport, storage, and access that can apply to all or the majority of the individual domains. The problem is exacerbated by the spatio-temporal representation of the data-records which present challenges in treating characteristics (like location and movement) systematically. While we recognize that it will not generally be reasonable to fix or mandate one schema across geographically disparate deployments or sensor modalities, we believe that it is still a reasonable expectation to have a common baseline framework to represent and serve sensor data. Experiences constructing components of a wide-area sensor network for the real-time collection and integration of sensor data [4,5] show that a critical prerequisite to offering sensor data to a wide variety of applications (a model or software component that is a customer of the sensor data as well as a potential actuator of the sensor) is uniformity of representation and storage. Analogous to the interoperable Internet (for accepted access modes such as http-based communication), data providers can participate in the sensor network operation, regardless of the nature of the data source, or of the application model, if they comply with recommended guidelines for data representation and access. Here, we employ a methodology for sensor data characterization that uses ontological principles to guide the definition of the entity categories. Specifically, we use an ontology-driven software development methodology [6] building upon the SNAP/SPAN [7] upper-level spatio-temporal ontology as a foundation for the basic conceptual schema. The ontology-driven design methodology results in a desired common framework, and enables us to construct the sensor network's data plane generically. For purposes of this paper we refer to the ontology-driven

approach as *top-down* in contrast to an entity-oriented approach which we call *bottom-up*.

1.1 Contributions

We make the following contributions. First, we discuss integration concerns in WASNs from the position of the well-studied area of federated databases, and propose a spatio-temporal loosely-coupled federated database model for WASN sensor data storage. Second, we design a high-level conceptual schema for achieving flexible integration of sensor data representations. The conceptual schema we offer, based on a geospatial ontology, is capable of addressing the spatio-temporal aspects of WASN data collection. Third, to evaluate our ontology-driven methodology of conceptual schema design, we qualitatively compare two conceptual schemas developed over the course of the SensorNet architecture design process, and discuss their strengths and weaknesses. One approach is a bottom-up entity-oriented modeling approach that takes advantage of GML [8] concepts and the other is a top-down approach derived from the ontology-driven approach. We find that although the ontology-driven conceptual schema offers the richer semantics that one may need, we do not yet address the implementation gap that conventional entity-oriented methodologies fill effectively.

We consider our deployment experiences in SensorNet [5], a WASN testbed to enable plug-n-play sensors and applications. SensorNet incorporates several standardized techniques for ingesting and disseminating data including standards from the Open Geospatial Consortium (OGC) suite of specifications [8,9,10,11] and the IEEE [12]. These specifications give us a unique and concrete foundation to investigate the state-of-the-art and state-of-the-practice in creating interoperable WASN data models.

1.2 Scope and Outline

This paper addresses the problem of schema integration in the context of WASN data interoperation. We relate our work to conceptual schema construction and work in ontology-driven software development (e.g., [6]) - our goal here is to compare two approaches to the schema construction in the specific domain of WASN. The connection between generic schema construction for wide-area sensor networks has not received very much attention in the literature. While they do not directly address conceptual schema construction, IrisNet [13] is an earlier effort where the focus is a querying mechanism acting on an Extensible Markup Language (XML) based distributed data collection and transmission framework, and, more recently, the Global Sensor Network effort [14] aims to use a virtual sensor abstraction to make sensors accessible uniformly in a middleware. We make references through the paper to related work and approaches we build upon in conceptual schema and ontology-driven software development. More general questions of schema integration are beyond the scope of the paper as are other practical challenges that are rooted in the distributed nature of WASN data such as, for example, the security aspects of access. The remainder

of the paper is organized as follows. Section 2 details general WASN architecture and discusses the problem of data integration in WASN. Section 3 presents our ontology-driven approach to conceptual schema design. In Section 4, we qualitatively compare the developed top-down and the bottom-up schemas. Section 5 contains a summary and avenues for future work.

2 WASN Architecture and Data Integration

In this section, we present general WASN architecture considerations and discuss the data interoperability problem.

2.1 Architecture and Operation

The WASN architecture [5] accomplishes distributed data dissemination from a large variety of sensors using different types of connectivity and data acquisition modes. A regional instantiation of a WASN consists of four primary entities: sensors (and actuators), aggregation nodes, regional data centers, and user applications (Figure 1).

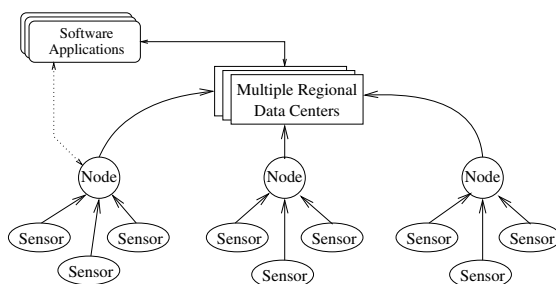


Fig. 1. WASN Architecture

The SensorNet Node is a device which collects data from various kinds of sensors and actuators, provides intermediate storage for the collected data, and transmits the data either to a regional center for permanent storage or a user application for analysis. The Node is built around an embedded PC-compatible computer that runs Linux OS and a Java-based server that implements the Open Geospatial Consortium (OGC) Web Feature Service (WFS) protocol [10]. (WFS was chosen because it was an accepted standard that supported spatio-temporal data transport and querying. Until recently, WFS alone had these characteristics.) Communication with the sensors takes place through a variety of means such as RS232 ports, USB or IEEE 1451 enabled devices [15]. The node is equipped with multiple standard wired and wireless communication devices such as modems and network ports. Nodes can be configured either as stationary platforms installed in the environment or compact transportable devices [16].

The WASN node supports several types of sensors (chemical, radiological, biological, weather and etc.). The Node data processing functions include:

- connecting to sensors,
- collecting data from sensors,
- issuing commands to sensor and actuators,
- preprocessing collected data to perform validation and calibration of the sensor readings,
- tagging the data with location coordinates and time stamps, and
- issuing alerts when a threshold is exceeded.

Applications access data from the regional data centers that act as hubs of the regional federate. Regional data centers thus provide permanent storage and retrieval capabilities for the collected data. Communications between software applications, nodes and regional centers on the WASN are performed using the WFS protocol for data insertions and retrievals. Entities within the message are described as OGC features. An OGC feature is a set of property-value pairs, one of whose properties is an OGC geometry with a geospatial reference [17]. Other feature properties carry information such as time stamps and measurements. Each feature has a feature type description associated with it.

2.2 Data Integration

A WASN imposes a federated database structure in the sense that, on the one hand, it provides unified access to a multitude of sensor data sources while, on the other hand, it allows regional autonomy on the data. For example, a WASN does not define a concrete set of regional centers or specific models of sensor devices that are needed to participate in the WASN. Such an approach allows for great flexibility and extensibility of the information system. It also minimizes the cost of connecting new and legacy sensor networks to a growing WASN. However, this approach brings many challenges typical to any federated and ad-hoc expanding data system.

To address the problems of unifying query and record structure and preserving the semantic integrity of the data, we first discuss database schema transformation and mapping in the context of a WASN. The database schema is a description of the data managed by the database system. Database schemata are composed of descriptions of the databases objects (tables, classes, types, etc.) and the relations between these objects. Schemata may be associated with vocabularies and ontologies to maintain relevant semantic information. In federated databases, compatibility between record structures and query languages is achieved by transforming or mapping component database schemata into a single federated schema and by converting the corresponding data as shown in Figure 2. In loosely-coupled federated databases, component database schemata typically develop independently of the federated schema and the data has to be converted according to several database schemata that play different roles in the federation. Component databases of a federated database system (FDBS) typically store their data using internal schemata that are optimized for the particular

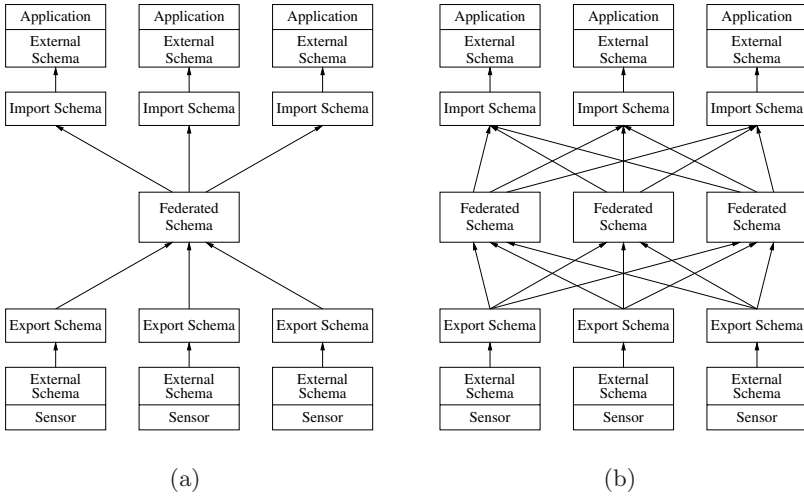


Fig. 2. Schema Translation in WASN

purpose a component database was developed for. Component databases expose their data using external schemata that are a subset of the internal schema and that contain the elements that are relevant for the FDBS.

In a WASN, the role of component databases on the data production side is played by sensors and partly by the WASN nodes. The internal schema of a sensor is either based upon a sensor standard (e.g., the IEEE 1451 Transducer Electronic Data Sheet (TEDS)) or defined during the WASN node setup. A sensor schema may contain data that is not necessarily pertinent to the overall operation of the WASN such as the calibration or sensor health information. Such data is omitted from the external sensor-level schema. On the WASN node, the data is tagged with a spatio-temporal reference that also becomes a part of the external sensor schema. An external schema of a sensor may be specific to that sensor. For a federation of a large number of sensors, the sensor data has to be translated into some common federated schema (“Federated Schema” in Figure 2(a)). We do not preclude multifederation with several federated schemata (Figure 2(b)) but do not address it here for space reasons.

For an application to be able to retrieve WASN data, it must be able to translate the WASN federated schema into its own external schema. In most cases a user will need to develop an import schema that is compatible with the application external schema which represents a subset of the WASN federated schema as shown in Figure 2(a).

3 Conceptual Schema Design

Due to the potentially unlimited variety of applications that can utilize WASN data and the many types of sensors accessible through the WASN, the main

challenge of federated schema development is to create a schema that would be universal enough to accommodate a large variety of potential data sources and data uses and yet be restrictive enough to gain the benefit of a common schema.

The conceptual schema needs to capture the most general entities and relations contained in the system. The role of a conceptual schema is to serve as a basis for the development of the implementation-level schemata expressed in formal languages like SQL data definition language, XML schema or others. A WASN conceptual schema should cover not only entities related to sensors but also the most general notions that can be used by the user applications. Before implementation, the software developer translates entities of the conceptual schema into implementation-level objects of a particular computation platform such as table definitions of relational databases or classes of object-oriented programming languages. We note that there may not be a direct correspondence between conceptual- and implementation-level entities. For example, a sensor on the WASN conceptual schema (Figure 5) may be represented as several inter-linked relational tables.

Different methodologies for the development of conceptual schemata have been discussed in the literature [18,19]. In this study we present two approaches for WASN conceptual schema development in the SensorNet project. One approach is a bottom-up approach based on entities, and the other is a top-down approach using the upper-level SNAP/SPAN ontology.

3.1 Bottom-Up Conceptual Schema

Figure 3 shows the structure for the bottom-up schema [20]. The data interoperation is accomplished through WFS clients located on the WASN nodes and WFS servers at the data centers. For the communications inside the WASN that we consider here, the data is encoded in a GML application schema and each record is represented as an OGC feature tagged with a spatial location and a time stamp.

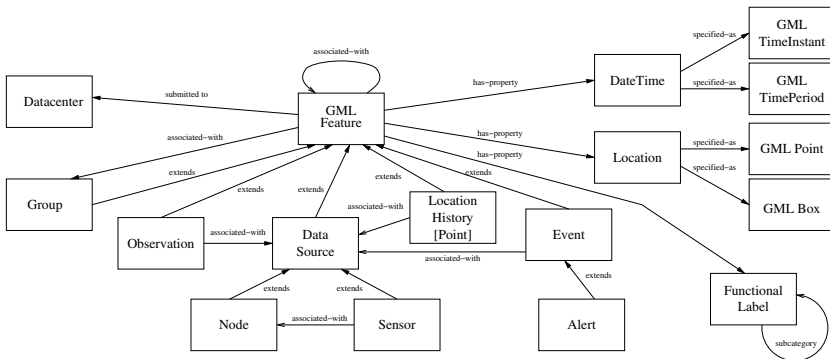


Fig. 3. Bottom-Up SensorNet Concept Map

The central entity of the SensorNet schema is the GML Feature inherited from the GML schema that is a set of property-value pairs, some of whose values represent geometry. Each Feature and its descendants has properties to characterize its spatial and temporal locations. To represent locations, the schema incorporates a limited subset of GML geometries including GML Points, TimeInstants and bounding boxes. Although this set of geometries can potentially be extended to other GML geometry types, the types used satisfy most of the requirements for storage and retrieval of the data in the WASN and limiting to these types significantly simplifies implementation.

Data centers, Observations, Events, Alerts and Data Source all extend the GML Feature and inherit its properties. An Observation is used to store measurements performed by the Sensors. Nodes and Sensors represent Data Sources that can be queried or instructed to perform Observations. Each Observation is always associated with a Data Source. In the WASN architecture, locations are associated with the Nodes which are equipped with global positioning system (GPS) devices (a Node inherits its Location property from the GML feature). After a measurement has been taken, its geographic coordinates are stored in the Node database as a GML Point. Sensor measurements are stored within Observations as XML records whose structure is defined by sensor-specific Data Type Definitions (DTDs). Events and Alerts are intended to communicate information such as sensor observations exceeding certain values. Events represent changes notable from the SensorNet operation point of view, e.g., low battery reading, communication channel failures. Alerts are a narrower category of Events intended for SensorNet users. Alerts are issued in cases of, for example, radiation sensor readings that exceed a certain threshold. Each Alert and Event is associated with a Node and Sensor that it has originated from. Location History is used to represent movements of Sensors and Nodes. Each record of the Location History type describes a point and associated time stamp for the location of a Sensor or a Node. It can be retrieved from a WFS just like any other Feature.

The approach for developing the schema shown in Figure 3 is based on the GML specification and accumulated practical knowledge of developing sensors and sensor platforms. The schema evolved in parallel with the architecture prototype being built. Additional entities (for example, Location History) are added to the schema in order to accommodate the needs of the prototype development and continually translated to an XML structure using standard ER-Modeling based translation.

3.2 Ontology-Driven, Top-Down Approach

The breadth of the potential application area of the conceptual schema necessitates the use of a systematic methodology for the study of conceptualization. To develop such a schema we investigate the conceptualization of the WASN domain area. Our goal is to produce a conceptual schema that later would serve as a foundation for implementing a WASN federated schema following the approach called “ontology-driven development” as proposed in [6]. We build a conceptual

schema using a formal ontology¹ as a tool for comprehending the conceptualization of the WASN domain.

We express our domain conceptualization (or domain ontology) as a repertoire of entities and relations that are necessary to communicate the concepts (as opposed to the artifacts). Examples of such entities within the WASN domain are sensors, measurements, classes of sensors, etc. Relations are defined between entities. Examples of relations in the WASN context would be an association between a measurement and a sensor, or a “kind-of” relation between a specific sensor and a class of the sensor device that it belongs to. To create the catalog of entities, we use a methodology similar to that described in [19]. We study the various narratives describing sensor data acquisition and use. To create a list of entities, we identify nouns and noun phrases specific to the sensor domain. Then we clean the list of synonyms and entities with low importance in the context of a WASN. We obtain the list of relations by analyzing association between the entities and also using general expertise in the domains of sensors and geographic data. After collecting a catalog of entities and relations, the next step we take in the development of the domain ontology is to classify the domain entities according to the broadest top-level ontological categories. Development of an exhaustive but meaningful classification of all entities of some domain would be a significant challenge, and consequently we rely on a sample top-level ontology² discussed in the philosophical literature and described in [7].

SNAP/SPAN Ontology. Here we use the top-level categories of the SNAP/SPAN ontology (Figure 4, [7, page 74, modified and highly simplified]) to build a conceptual schema for the WASN - shown in in Figure 5. This ontology subdivides all entities along class/individual and SNAP/SPAN dimensions (which we describe below) and has advantages such as independence from any particular application domain in the WASN and thus has an intrinsic ability to account for the dynamic nature of the reality. It also has an extensive formalization in first-order predicate logic [31]. The choice of SNAP/SPAN ontology was largely influenced by its explicit support of dynamic entities.

Support for dynamics is critical for WASNs because monitoring change and detection of processes is a goal of any WASN operation. The SNAP/SPAN ontology framework that we adopt here has previously been successfully applied to the design of spatio-temporal applications [32,33,34], analysis of the conceptualization in hydrologic models [26], and the solving of semantic interoperability problems for CAD/GIS data models [35].

¹ Formal ontology is a study of the basic constituents of reality and is deeply rooted in philosophy [21]. Here we use the notion of ontology as an explicit specification of the conceptualization [22]. Ontological methodologies have proved their efficiency in other domains including medical information systems [23,24] and biological classifications [25].

² The use of top-level classifications is common in ontology development [26,27,28]. Examples of systems of top-level categories of entities are IEEE SUMO ontology or John Sowa’s upper-level ontology [29,30].

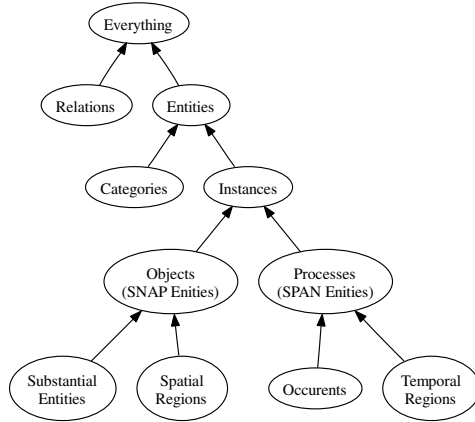


Fig. 4. Top-level Ontological Categories

Top-level ontological categories are systematized as a proper tree with each domain entity instantiating one and only one terminal (leaf-level) category (Figure 4). At the root of the category hierarchy, all entities are subdivided into classes and individuals. Classes represent concepts, while individuals are entities that have specific spatial and/or temporal locations [36]. For example, “temperature sensors” is a class and a particular temperature sensor outside a building is an example of an individual.

The central dichotomy of the SNAP/SPAN ontology in the branch of individuals is the subdivision of all individuals into two non-overlapping classes of entities — SNAP (or object-like) entities and SPAN (or process-like entities) as shown in Figure 4. Occasionally, in the literature, these SNAP and SPAN entities are referred to as endurants and perdurants respectively. SNAP entities are capable of preserving their identity through time. Examples of SNAP entities are humans, planets, cities, crevices, etc. Examples of SNAP entities in the WASN domain are sensors, actuators, nodes or service personnel. By contrast, SPAN entities unfold in time and only a part of a SPAN entity exists at each moment of time. (A perdurant is always dependent upon an endurant, for example, in the pair *(you, your_life)* *you* is an endurant and *your_life* is a perdurant. *your_life* has such parts as childhood, youth, adulthood and old age. At any point of time only one part of your life exists. *your_life* is dependent on *you* in the sense that it does not make sense to discuss *your_life* without referring to *you* and there is nothing like *your_life* if *you* does not exist.) Processes and events (instant occurrences) are kinds of SPAN entities. Other examples of SPAN entities are communication sessions, an alert event (as opposed to an alert message) and many others.

WASN Ontology-Driven Conceptual Schema. The central notion (or entity or concept) of the WASN conceptual model is a *sensor*³, which is a particular

³ *Sensors* can be equipped with *actuators* that control *sensor* position, orientation, etc.

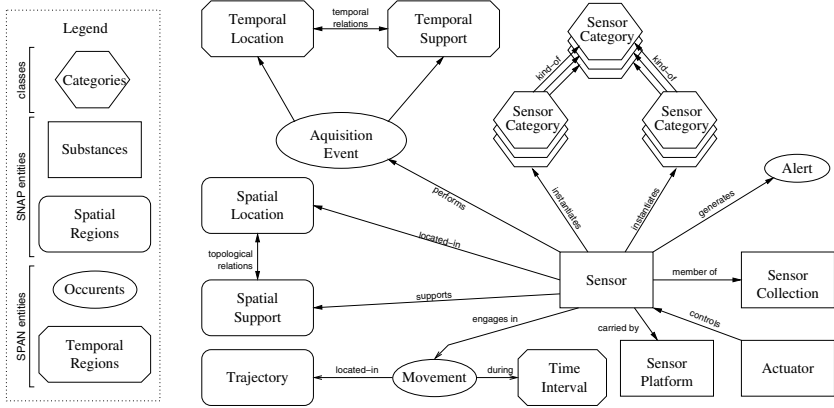


Fig. 5. WASN Ontology-driven Conceptual Schema

device capable of performing measurements of some properties of the environment. To house *sensors*, we require *sensor platforms*. *Sensor platforms* can be stationary, like poles or wall brackets, or mobile. A single *sensor platform* can carry several *sensors*. In this case, *sensors* are said to be arranged into *sensor collections* that share the same *platform*. An example of a *sensor collection* is a WASN node. Individual entities of the conceptual schema correspond to the substantial entities of the top-level ontological categories in Figure 4. Each *sensor*, *sensor collection*, *platform* and *actuator* has a corresponding record in the database of a regional center it belongs to. Each of these entities is first registered with a WASN node. Then the node transmits the descriptive data to a regional center from where it becomes available to the WASN users. The main purpose of these records is to support inventory and discovery of the available equipment that can be used for data collection.

Sensor categories are another facility for WASN resource discovery. They provide users with a class view over available sensors and other resources. *Sensor categories* represent types and classes of sensors and are organized strictly hierarchically. There are several hierarchies intended to capture various dimensions of the category space of *sensors*. For instance, there can be hierarchies created by the type of measured property or measurement technology. A measured property hierarchy is used to query the different kinds of measurements available on the WASN. Another hierarchy can be set by the manufacturer to be used for hardware maintenance tasks. Each sensor belongs to at most one category in a single hierarchy but it can be represented in multiple hierarchies. *Sensor categories* also play another important role by supporting automatic aggregation of the data collected from *sensors*. For example, sensors capable of measuring temperature can be made by different manufacturers. The WASN cannot impose the requirement to use a specific sensor model of a sensor produced by a specific manufacturer. However, the data collected by the models produced by different manufacturers are likely to be comparable and should be visible to the end-user

application as a single aggregated dataset of temperature measurements. Such aggregations can be performed along the specialized sensor category hierarchy. *Sensor categories* are Classes in the top-level ontological categories hierarchy (Figure 5). The system of categories can be extended to entities other than *sensors*.

A *sensor* position within the geographic space is represented through two kinds of locational entities: *spatial location* and *spatial support*. The *spatial location* of a *sensor* is a point on the Earth surface where a *sensor platform* is found at the moment when a *sensor* is active and performs measurements. Static *platforms* have fixed *locations* while *locations* for movable and mobile *platforms* can change. In the WASN, a *sensor location* is determined on the basis of the GPS coordinate of a WASN node. A *sensor* is said to have a *support*, that is, a spatial region that is characterized by the sensor. In most cases a *support* is different from a *sensor location*. For example, a pan-tilt-zoom camera can be installed on a poll with known geographic coordinates. The area that this camera captures is the *support* of this *sensor*. Another example is a temperature sensor whose *support location* is limited to the room in which the sensor is installed. Sensor location and sensor support can be deduced from the properties of the *sensor* itself and the *sensor platform*.

Spatial location and *spatial support* belong to the category of “spatial regions” of the top-level hierarchy (Figure 4). In the WASN database they are recorded as GML Feature Geometry objects [8]. For automatic inferencing, relations between location and support can be expressed using an Egenhofer nine intersection model or region connectedness calculus [37,38].

Each occurrence of a measurement performed by a *sensor* is called an *acquisition event*. An *acquisition event* is a SPAN entity in the top-level hierarchy in Figure 4. Records of *acquisition events* comprise the bulk of data stored in the WASN and are the major target of WASN queries. Each *acquisition event* is associated with the *sensor location* and *support* at the moment that the measurement was performed. As a SPAN entity, an *acquisition event* has a *temporal location* and a *temporal support*, both representing temporal regions from the top-level category hierarchy (Figure 4). *Temporal support* plays the same role in relation to *temporal location* as *spatial support* in relation to *spatial location*, that is, a temporal region representing the time period for which the result of the measurement remains valid. An *acquisition event temporal location* is not necessarily a time instant and may have more than a zero time length.

A *temporal support* is typically confined within a *temporal location*. For example, a camera usually needs some time to adjust to lighting conditions after being switched on. The first few frames typically contain erratic data and will be a part of the *acquisition event temporal location* but not of the *temporal support*.

Movement of the sensor is represented by a special SPAN entity called *Movement*. *Movement* is intended to capture the constantly or intermittently changing location of a *sensor*. *Movement* has two entities associated with it: *Trajectory* is a spatial region to represent trajectory of the movement, and *Time Interval* represents a temporal region during which the movement has occurred.

4 Comparison of Conceptual Schema Constructions

The schemata in Figure 3 and Figure 5 were developed for the same domain but on different premises and they exhibit different sets of entities and relations.

The biggest difference between the schemata is the direct use of the well-defined GML Feature in the bottom-up schema (Figure 3). Most other entities of the schema extend (i.e., inherit their properties from) the GML Feature that enables them to have such properties as Functional Labels and spatial and temporal locations. In terms of object-oriented programming (OOP), the GML Feature represents a superclass of other classes in the schema. The benefit of structuring the schema in this bottom-up way allows implementors to translate the schema to a WFS engine or a database mapping directly. The ontology-driven schema (Figure 5) does not use the extend relation in its OOP sense, leaving that to more implementation-level schemata, thus gaining semantic flexibility for some ambiguity in implementation. As may be expected, the goal of allowing a common semantic structure to describe diverse application concepts comes at the cost of not providing specific mappings for implementors. If WASN architects recommend a specific implementation (e.g., WFS or a web-service implementation) along with an advertised vocabulary, the ontology-driven approach provides semantic commonality (and a top-down systematic structure to build upon) and enables implementation interoperability.

4.1 Representation of WASN Infrastructure

Representation of the WASN infrastructure in the two schemata have important differences stemming from the central component used for the modeling. The GML Feature element provides a common glue that does not easily enable the model to capture the semantic interdependence of the other entities. For the edges in the graph in Figure 3, the named relationships start resembling a part of the notions of a Sensor and Observation (Acquisition Event in Figure 5). The benefit of the latter of course is that the modeler gravitates to these representations at the outset instead of in retrospect. As an example, the SensorNet Node present in the bottom-up schema corresponds to the Sensor Collection in the ontology-driven schema. However, the notion of Sensor Collection is a more general concept and easily extends to other groupings of sensors such as actuators. While an expert designer would inject a systematic structure at the beginning in Figure 3, our claim is that the approach illustrated in the top-down ontology-driven approach orients the designer's mindset to allow systematic model construction.

4.2 Representation of Space and Time

There are significant differences in the models of space and time used in the schemata. The subset of GML geometric primitives that includes Point, Box, TimeInstant and TimePeriod are specified first as building blocks. It takes designer ingenuity to collect and apply them for a WASN use case. In contrast, the

ontology-driven schema orients itself to apply to the domain of interest and the intended meaning. In addition to Sensor Spatial and Temporal Location (Acquisition Time), it introduces the notion of Support, i.e., a spatial or temporal region that is characterized by the sensor. Emphasis on user-level queries and applications in the ontology-driven schema necessitates introduction of such entities as Spatial and Temporal Support. The top-down representation of space and time allows designers to easily include more complex notions of temporal phenomena and their relationship to the spatial components. For example, topological configurations may be described (and discussed) as first-class objects. It is harder to instantiate this structure in the bottom-up model because aggregations of constituent components requires additional entities (e.g., a trajectory as a collection of points).

4.3 Representation of Motion and Dynamics

The schemata take different approaches towards representation of movement and dynamics. In the bottom-up schema, Location History is an add-on entity that stores coordinates and time stamps of sensor locations. The ontology-driven schema uses a domain-aware, and therefore directly applicable entity called Movement that is associated with spatial and temporal regions. Relating *Movement* to an implementation helps a designer in bringing to the surface the true intention of the entity, as opposed to using a construction such as Location History.

5 Summary and Future Work

This study addresses the problem of semantic heterogeneity and semantic interoperability of data collected in WASNs. The study presents an entity-oriented (bottom-up) and an ontology-driven (top-down) methodology for conceptual schema construction and illustrates a set of differences. The top-down conceptual schema offers the potential for better interoperability with user-level software applications because user semantics are taken into account early in the development process. The interoperability advantage comes with the higher level of independence of the conceptual schema from a computing platform because it relies on a very general conceptual framework. The schema can ultimately be implemented using the technologies that the bottom-up approach uses. The bottom-up approach has the benefit of being closer to the implementation and thus allows rapid translation to an implementation structure. An overall conclusion of the study is that ontology-driven methodologies can benefit extensibility and cross-domain interoperability over time but require larger efforts at the early design and development stages.

Several research activities can build upon our work. A direction for future development of the ontology-driven conceptual schema proposed in this paper is to follow through with the prime motivator for ontologies in the first place - formal models in order to identify internal inconsistencies and ambiguities. This

issue becomes particularly relevant as the research community starts formalizing the systematic mapping of semantics (from top-down conceptual schema design) into the design/development of the database schema implementations (the latter remains a gap in the state-of-the-art). Another clear direction for research is evaluating quantitatively the performance differences in the implementations that derive from the two schema constructions.

Acknowledgments

Authors thank Ranga Raju Vatsavai from the Oak Ridge National Laboratory for helpful advice and comments concerning the paper.

Research sponsored by SensorNet Program, Office of Naval Research, and Oak Ridge National Laboratory (ORNL) managed by UT-Battelle, LLC for the U. S. Department of Energy. The submitted manuscript has been authored by a contractor of the U.S. Government under contract DE-AC05-96OR22464. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

References

1. Heyer, R.J.: Introduction to CBRNE Terrorism: An Awareness Primer and Preparedness Guide for Emergency Responders. The Disaster Preparedness and Emergency Response Association. 3 edn. (2006), <http://www.disasters.org>
2. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. *Computer Networks* 38(4), 393–422 (2002)
3. Karl, H., Willig, A.: A short survey of wireless sensor networks. Technical report, Technical University Berlin (2003)
4. Shankar, M., Liu, C., Bhaduri, B.: From static to dynamic models: Enabling real-time geocomputation infrastructures. In: *GeoComputation 2005* (2005), <http://igre.emich.edu/geocomputation2005/>
5. Gorman, B.L., Shankar, M., Smith, C.M.: Advancing sensor web interoperability. *Sensors Online* (4) (2005), <http://sensorsmag.com/articles/0405/14/>
6. Frank, A.U.: Ontology for spatio-temporal databases. In: Sellis, T., Koubarakis, M., Frank, A., Grumbach, S., Güting, R.H., Jensen, C., Lorentzos, N.A., Manolopoulos, Y., Nardelli, E., Pernici, B., Theodoulidis, B., Tryfona, N., Schek, H.-J., Scholl, M.O. (eds.) *Spatio-Temporal Databases*. LNCS, vol. 2520, pp. 9–77. Springer, Heidelberg (2003)
7. Grenon, P., Smith, B.: SNAP and SPAN: Towards dynamic spatial ontology. *Spatial Cognition and Computation* 4(1), 69–104 (2004)
8. Open GIS, Consortium 35 Main Street, Suite 5, Wayland, MA 01778, USA: *Geography Markup Language (GML)*. v3.1 edn (2004) <http://portal.opengeospatial.org/files/>
9. Open, G.I.S.: Consortium: *OpenGIS® Sensor Model Language (SensorML) Implementation Specification*. 1.0 edn (2006), <http://vast.nsstc.uah.edu/SensorML/>
10. Vretanos, P.A.: *Web Feature Service Implementation Specification*. Open GIS Consortium Inc., 35 Main Street, Suite 5, Wayland, MA 01778, USA. Version: 1.0.0 edn. (2002) OGC 02-058

11. Consortium, O.G.: Sensor web enablement initiative. web page, (2006)
<http://www.opengeospatial.org/projects/groups/sensorweb>
12. IEEE Standards Board: Draft Standard for A Smart Transducer Interface for Sensors and Actuators - Common Functions, Communications Protocols and Transducer Electronic Data Sheets (TEDS) Formats (2007)
13. Gibbons, P.B., Karp, B., Ke, Y., Nath, S., Seshan, S.: Irisnet: An architecture for a world-wide sensor web. *IEEE Pervasive Computing* 4 (2003)
14. Aberer, K., Hauswirth, M., Salehi, A.: A middleware for fast and flexible sensor network deployment. *ACM Very Large Databases (VLDB)*, 1199–1202 (2006)
15. IEEE Standards Board: IEEE Std 1451.2-19974: IEEE Standard for a Smart Transducer Interface for Sensors and Actuators (1997)
16. Chin, J.C., Hou, I.H., Hou, J.C., Ma, C., Rao, N.S., Saxena, M., Shankar, M., Yang, Y., Yau, D.K.Y.: Sensornet platforms: Sensor-cyber network testbed for plume detection, identification, and tracking. In: *Information Processing in Sensor Networks* (2007)
17. Open GIS, Consortium 35 Main Street, Suite 5, Wayland, MA 01778 USA: The OpenGISTM Abstract Specification. Topic 1: Feature Geometry. Version 4 edn. (1999)
18. Daniels, J.D., Cook, S.: *Designing Object Systems: Object-Oriented Modelling with Syntropy*. Prentice Hall, NY (1994)
19. Mannino, M.V.: *Database Application Development and Design*. Irwin McGraw-Hill, Boston (2001)
20. Resseguie, D.: Sensornet data center services design document. web page (2006),
<http://www.us.sensornet.gov/SensorNetDocs/>
21. Smith, B.: Ontology and information systems. Technical report, NCGIA (2000),
<http://ontology.buffalo.edu/smith/articles/ontologies.htm>
22. Gruber, T.R.: A translation approach to portable ontologies. *Knowledge Acquisition* 5(2), 199–220 (1993)
23. Rector, A., Rogers, J.: Ontological issues in using a description logic to represent medical concepts: Experience from GALEN. In: *IMIA WG6 Workshop: Terminology and Natural Language in Medicine*, January 1997, Phoenix, Arizona, International Medical Informatics Association (1999)
24. Smith, B., Rosse, C.: The role of foundational relations in the alignment of biomedical ontologies. In: *Proceedings of Medinfo (March, 22–24) San Francisco* (2004)
25. Smith, B.: The logic of biological classification and the foundations of biomedical ontology. In: *Invited Papers from the 10th International Conference in Logic Methodology and Philosophy of Science*, Oviedo, Spain, 19–25 August, Elsevier-North-Holland (2004)
26. Feng, C.C., Bittner, T., Flewelling, D.: Modeling surface hydrology concepts with endurance and perdurance. In: Egenhofer, M.J., Freksa, C., Miller, H.J. (eds.) *GIScience 2004*. LNCS, vol. 3234, pp. 67–80. Springer, Heidelberg (2004)
27. Russomanno, D.J., Kothari, C.R., Thomas, O.A.: Building a sensor ontology: A practical approach leveraging ISO and OGC models. In: Arabnia, H.R., Joshua, R. (eds.) *IC-AI*, CSREA Press, pp. 637–643. CSREA Press (2005)
28. Keet, C.M.: Factors affecting ontology development in ecology. In: Ludäscher, B., Raschid, L. (eds.) *DILS 2005*. LNCS (LNBI), vol. 3615, pp. 46–62. Springer, Heidelberg (2005)
29. Niles, I., Pease, A.: Towards a standard upper ontology. In: *FOIS 2001: Proceedings of the international conference on Formal Ontology in Information Systems*, pp. 2–9. ACM Press, New York (2001)

30. Sowa, J.F.: Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks Cole Publishing Co., Pacific Grove, CA (2000)
31. Bittner, T., Smith, B.: Formal ontologies for space and time (2003), <http://ontology.buffalo.edu/geo/sto.pdf>
32. Worboys, M., Hornsby, K.: From objects to events: GEM, the geospatial event model. In: Egenhofer, M.J., Freksa, C., Miller, H. (eds.) Third International Conference on GIScience. LNCS, vol. 2334, pp. 327–345. Springer, Heidelberg (2004)
33. Worboys, M.F.: Event-oriented approaches to geographic phenomena. *International Journal of Geographical Information Science* 19(1), 1–28 (2005)
34. Worboys, M.: Knowledge discovery using geosensor networks. Technical report, University Consortium for Geographic Information Science (2003), <http://www.ucgis.org/Visualization/whitepapers/Worboys%20paper.pdf>
35. Bittner, T., Donnelly, M., Winter, S.: Ontology and semantic interoperability. In: Prosperi, D., Zlatanova, S. (eds.) Large-Scale 3D Data Integration, pp. 139–160. CRC Press, London (2004)
36. Lowe, E.J.: *A Survey of Metaphysics*. Oxford University Press, Oxford (2002)
37. Randell, D.A., Cui, Z., Cohn, A.G.: A spatial logic based on regions and connection. In: Kaufmann, M. (ed.) 3rd International Conference on Knowledge Representation and Reasoning (1992)
38. Egenhofer, M.J.: A formal definition of binary topological relationships. In: Litwin, W., Schek, H.-J. (eds.) FODO 1989. LNCS, vol. 367, pp. 457–472. Springer, Heidelberg (1989)