Sensor Discovery on Linked Data

Joshua Pschorr, Cory Henson, Harshal Patni, and Amit Sheth

Kno.e.sis Center, Department of Computer Science and Engineering,
Wright State University, Dayton, Ohio, USA
pschorr.2@wright.edu, {cory, harshal, amit}@knoesis.org

Abstract. There has been a drive recently to make sensor data accessible on the Web. However, because of the vast number of sensors collecting data about our environment, finding relevant sensors on the Web is a non-trivial challenge. In this paper, we present an approach to discovering sensors through a standard service interface over Linked Data. This is accomplished with a semantic sensor network middleware that includes a sensor registry on Linked Data and a sensor discovery service that extends the OGC Sensor Web Enablement. With this approach, we are able to access and discover sensors that are positioned near named-locations of interest.

Keywords: Linked Data, Architectures and Middleware for Semantic Sensor Networks, Semantic Web, Sensor Discovery, Sensor Web Enablement

1 Introduction

There are millions of sensors collecting data about our environment. Many of these sensors and their observations are now becoming accessible on the Web. While such accessibility is a great achievement, it also poses new challenges. One such challenge involves the ability to discover sensors on the Web that are relevant and useful for the needs of a particular application or user. For example, sensors near an object, event, or situation of interest are more relevant than those located farther away. While this may seem an overly obvious example, current solutions are still often unsatisfying.

Consider the following scenario: You are interested in finding temperature and precipitation sensors near *Wright State University* so that you can decide whether to take a coat and umbrella to school. Executing this query against current Sensor Web services [1][2] requires the user to input bounding-box coordinates (e.g., *N 39° 45′ 32″, W 84° 11′ 29″*) referring to the location of interest. The use of specific coordinates to represent a location can often be unintuitive and cumbersome for naïve users in comparison to the more semantically relevant term, *Wright State University*.

Several projects, such as *GeoNames* [3] and *LinkedGeoData* [4], have begun publishing expressive descriptions of spatial data and named locations on the Web as Linked Data [5][6]. Relating descriptions of sensors to nearby locations defined within these open spatial datasets will allow more intuitive sensor discovery queries through named locations. We have generated several such datasets on Linked Data containing sensor information with links to named locations, as discussed in section 3.2. The first dataset, *LinkedSensorData*, contains descriptions of over 20,000 weather

stations located in North America. The second dataset, *LinkedObservationData*, contains descriptions of observations from these weather stations and includes over one-billion triples.[1]

It is often the case, however, that users and application developers who may want to write sensor discovery queries are unaware of Semantic Web technologies such as RDF[2], SPARQL[3] and Linked Data. To accommodate such users, it will be necessary to integrate commonly used Sensor Web technologies with Semantic Web technologies to enable access to more expressive semantic descriptions of sensor data and locations found on Linked Data. The Open Geospatial Consortium (OGC) Sensor Web Enablement (SWE) provides a set of standard languages and services commonly used by the sensors community. [7] In particular, the Sensor Observation Service (SOS) provides a standard interface for accessing sensor descriptions and observations. [2] By making SOS semantically aware, we can take advantage of semantic descriptions of named locations to allow more intuitive sensor discovery queries. This semantically aware SOS is termed SemSOS. [8]

A synergistic integration of Semantic Web and Sensor Web technologies promises the ability to meaningfully access, discover, and query the Web of sensors and observations. Through meaningful semantic annotation, integration with existing spatial knowledge bases, and support for current Sensor Web services, we can provide a middleware for semantic sensor networks capable of providing the ability to access and discover sensors in a more intuitive manner. In this paper, we describe our approach to developing such a middleware and demonstrate how it allows us to solve the problem of sensor discovery on the Web. In particular, our contributions include:

- Semantic description of sensor data and integration with Linked Data to support sensor discovery based on named locations.
- Semantic enablement of the SWE Sensor Observation Service to support access to sensor descriptions on Linked Data.

2   Background

It has been predicted that by 2015 nearly every artifact in our environment will contain sensors and be connected to the Web. [9] As we progress towards this goal, there is a greater need for discovery, access, querying, and reasoning over sensor data on the Web. The OGC's Sensor Web Enablement [7] project has developed a set of standard languages and Web service interfaces for managing Web accessible sensor data. The SWE languages are XML-based and thus provide syntax-level interoperability but lack the semantic-level interoperability needed for advanced integration and analysis. In order to address this challenge, there has been a recent attempt to combine Sensor Web and Semantic Web technologies into a Semantic Sensor Web. [10] The World Wide Web Consortium (W3C) has also recognized this challenge and has initiated the development of the Semantic Sensor Network Incubator Group (SSN-XG). The goal of the SSN-XG is *to begin the formal process*

---

[1] http://wiki.knoesis.org/index.php/SSW_Datasets
[2] http://www.w3.org/RDF/
[3] http://www.w3.org/TR/rdf-sparql-query/

of producing ontologies that define the capabilities of sensors and sensor networks, and to develop semantic annotations of a key language used by services based sensor networks. [11] In this capacity, the Semantic Sensor Networks Incubator Group has recognized sensor discovery as a top-priority use-case to showcase the benefits of integrating Sensor Web and Semantic Web technologies.[4] This use-case provides the motivation for this paper.

2.1 Sensor Discovery on the Web

With huge amounts of sensor data now available on the web, discovery of the sensors and observations of interest becomes a more important and challenging problem. In the past, consumers of sensor data often also fulfilled the role of sensor data producers with a strong tie between application and sensor network. This tie is beginning to break with producers now often unaware of where and how their data is being used, and consumers often unaware of where and how to find relevant data. Of particular importance to many consumers is location-based discovery of sensors.

To support the type of rich, location-based semantics that we wish to use for sensor discovery, it is necessary to annotate the sensor descriptions and their observations with useful metadata. The SWE Sensor Model Language (SensorML) does this by encoding metadata about the coordinate-based geometric characteristics of sensors and sensor systems. While this type of metadata makes it possible to determine the geospatial point in which a sensor operates, it requires an extra step to determine whether the specified coordinates of each sensor fall within the user's target location. The prevailing solution to the above problem is to use a registry for discovery, but this approach has issues of its own. Registry approaches have run into problems of scalability in the Web Services community. The complexity necessary to support the various types of metadata and the centralized nature of registries makes them difficult for consumers to query and for providers to update. The latter issue is of particular importance, as we would like new sensors represented in discovery results as soon as possible. Furthermore, existing solutions, such as the OGC's Catalog Service (CS-W) are often too general, and while they deal well with relatively static GIS data, do not handle the dynamic nature of sensor data. [12]

2.2 Sensor Observation Service

The SWE standards currently enjoy wide-spread use within the Sensor Web community. The OGC standard *Observations & Measurements* (O&M) defines a model for encoding sensor observations, while the *Sensor Model Language* (SensorML) defines a model for sensor systems' observational and geometric characteristics. The OGC standard API for retrieving sensor and observation data is known as the *Sensor Observation Service* (SOS). A broadly useful solution to the problem of discovery must be able to support clients of these specifications, as they are the de facto standard ways to access the Sensor Web.

The Sensor Observation Service (SOS) is an OGC-SWE standard which defines a web service interface for providing *access to observations from sensors and sensor*

---

[4] http://www.w3.org/2005/Incubator/ssn/wiki/Use_cases

systems in a standard way that is consistent for all sensor systems including remote, in-situ, fixed and mobile sensors. SOS groups observations made by related sensor systems into *Observation Offerings*. An *Observation Offering* is a logical collection of sensors and sensor systems that, generally, are located in proximity to one another and sample their environment at shared intervals.

SOS defines four service profiles: *core*, *transactional*, *enhanced*, and *entire* (which includes all functions from the previous three). For a standards compliant SOS service, only support for the core profile is mandatory, while all other profiles are optional. The core and enhanced profiles provide support for consumers of sensor data. A consumer client of sensor data requires methods for obtaining information about the service itself and requesting observations, sensor descriptions, features, etc. over some spatial and temporal context. This information is useful in applications such as visualization, data fusion, situation awareness, and sensor discovery. The transactional profile supports publishers of sensor data. Such publisher clients are responsible for acting as intermediaries between sensor networks generating observations and the SOS service that inserts sensor descriptions and observations into its repository. The *core* profile includes three operations: *GetCapabilities*, *DescribeSensor*, and *GetObservation*. The *GetCapabilities* function provides a means to request a description of the service and is of particular importance for sensor discovery. This description includes information such as service identification (service name, keywords, etc.), provider, and most importantly, metadata that allows for the discovery of the capabilities of the service. The capability description includes metadata about all supported functions of the service (including valid values and ranges for query parameters), filtering capabilities (logical operators that may be supplied with query parameters), and a full list of all *Observation Offerings* (including the parameters: sensor systems, time, phenomenon, location, etc.) defined within the service.

2.3 Semantic Web and Linked Data

Beyond the Semantic Web languages and technologies of RDF, RDF-S, OWL, and SPARQL, significant recent progress in the realization of the vision of Semantic Web is the emergence of Linked Data. Linked Data is a large and growing collection of interlinked public datasets, encoded in RDF, and spanning diverse areas such as life sciences, nature, science, geography, and entertainment. In the sensors domain, sources of geospatial information such as GeoNames and LinkedGeoData are of particular importance. *The GeoNames geographical dataset contains over eight million geographical names and consists of 7 million unique features including 2.6 million populated places and 2.8 million alternate names.* [3] In section 3.2, we will introduce two new sensor datasets, *LinkedSensorData* and *LinkedObservationData*, with links to locations defined in GeoNames.

3 Sensor Discovery on Linked Data

Our approach to supporting the goal of sensor discovery on the Web begins with exploiting the strengths of the Semantic Web. We define ontologies to model sensor data in order to support rich reasoning and query. We show how our models may be integrated with Linked Data in order to exploit already existing sources of spatial or thematic data. Finally, we draw upon these ideas to leverage Linked Data as our decentralized alternative to typical, insular registries, by describing sensor discovery over our linked datasets.

3.1 Semantic Representation of Sensor Data

By committing to an ontological model, applications may benefit from a shared semantics of sensor data, thus leading to improved interoperability. Our ontologies are fashioned after the SWE models of sensor descriptions and observations. This provides a well-understood model of sensor data and the ability to interoperate with existing SWE clients and services.

Ontology Model of Sensor Data. Within the O&M standard, an observation (*om:Observation*) is defined as an *act of observing a property or phenomenon, with the goal of producing an estimate of the value of the property*, and a feature (*om:Feature*) is defined as an *abstraction of real world phenomenon* [13]. (Note: *om* is used as a prefix for Observations and Measurements). The major properties of an observation include feature of interest (*om:featureOfInterest*), observed property (*om:observedProperty*), sampling time (*om:samplingTime*), result (*om:result*), and procedure (*om:procedure*). Often these properties can be complex entities that may be defined in an external document. For example, *om:FeatureOfInterest* could refer to any real-world entity such as a coverage region, vehicle, or weather-storm, and *om:Procedure* often refers to a sensor or system of sensors defined within a SensorML document. Therefore, these properties are better described as relationships of an observation. We have developed an encoding of the Observations and Measurements language in OWL, called O&M-OWL. In this ontology, we have defined the previous relations, and more, in a form that may be queried and reasoned over effectively in order to derive actionable knowledge of the environment from sensor observations. The translation between O&M in OWL and O&M in XML is straightforward.

Semantic Annotation of Sensor Data. While encoding sensor data in OWL is useful for advanced analysis and reasoning, SOS services are, in practice, implemented using XML. However, it is often useful to also embed semantic terminology defined in an ontology model into an XML document. This technique is called semantic annotation and is used for greater semantic interoperability of data encoded in XML, which provides only syntactic interoperability. Ontology terms are embedded in XML documents through model references, or URIs of concepts defined in an ontology [14]. The OGC-SWE standards already provide several mechanisms to reference concepts that are external to the document. Such concepts are either defined in another XML document and accessed through an XLink element or defined in a

registry and accessed through the *swe:definition* attribute. Using either mechanism, we can embed a model reference that will provide more meaningful description and thus enhanced semantic interoperability. This technique is also applied within the *GetCapabilities* operation in order to embed high-level *om:Feature* concepts that may otherwise be unavailable in an SOS *GetCapabilities* response. This is necessary to inform a SemSOS client of the precise description of concepts that may be used to query the knowledgebase.

3.2 Linked Sensor Data

Using the sensor model outlined above, we have generated several sensor datasets and made them available as Linked Data. The datasets contain sensor descriptions and observations collected from weather stations within the United States. These datasets provide links to GeoNames in order to support location-based sensor discovery.

Linked Data as a Sensor Registry. An ideal mechanism for sensor discovery on the Sensor Web should include facilities for expressive query against semantically meaningful user criteria, simple procedures for the inclusion of new sensors and observations, and the ability to extend and build upon existing data. These requirements are all fulfilled by Linked Data, while they highlight weaknesses of traditional service registries. As such, we position Linked Data as an alternative to more conventional registry approaches.

A registry for sensors can expect to have new sensors added occasionally, but must assume additional observation data will be added on a continuous basis. A traditional centralized registry system does not scale to the amount of sensor and observational data that we can expect sensor systems to generate. Linked Data, however, presents a decentralized approach to publishing sensor data by creating relations to existing data and providing dereferenceable URIs.

Extending existing data sets with new relationships is great advantage of using Linked Data as a registry for sensor information. Sensor datasets can make use of temporal, spatial, and thematic concepts published elsewhere in Linked Data. Just as important, however, sensors and observations created by one publisher may be extended by another simply by the generation of new relationships referencing the existing facts. The open and decentralized nature of Linked Data allows rich interaction between sensor and thematic data that is often absent or prohibitively complex given conventional, insular registries.

Sensor Descriptions on Linked Data. Using the model presented in section 3.1, we have generated a dataset of sensor descriptions called *LinkedSensorData*. This dataset is derived from data collected by MesoWest, a project within the Department of Meteorology at the University of Utah. [15] MesoWest continually collects data from over 20,000 weather stations phenomena within North America. On average, there are about five sensors per weather station measuring phenomena such as temperature, visibility, precipitation, pressure, wind speed, humidity, etc. In addition to location attributes such as latitude, longitude, and elevation, *LinkedSensorData* also contains links to locations in GeoNames. This dataset is now published as Linked Data.

Sensor Observations on Linked Data. Another dataset, called *LinkedObservationData*, has been generated that contains expressive descriptions of sensor observation data. This dataset is also based on data collected by MesoWest. The observations include measurements of phenomena such as temperature, visibility, precipitation, pressure, wind speed, humidity, etc. The dataset consists of observations made within the United States during the time periods in which several major storms were active (e.g. Hurricane Katrina). These observations were generated by the weather stations described in our sensor descriptions dataset, which they reference. Table 1 describes the storms, date ranges, and size of the *LinkedObservationData* dataset which currently contains over one billion RDF triples and is now published as Linked Data.

Table 1. LinkedObservationData statistics

Name	Storm Type	Date	Number of Triples	Number of Observations
Bill	Hurricane	Aug. 17-22, 2009	231,021,108	21,272,790
Gustav	Hurricane	Aug. 25-32, 2008	258,378,511	23,792,818
Bertha	Hurricane	July 6-17, 2008	278,235,734	25,762,568
Wilma	Hurricane	Oct. 17-23, 2005	171,854,686	15,797,852
Katrina	Hurricane	Aug. 23-30, 2005	203,386,049	18,832,041
Charley	Hurricane	Aug. 9-15, 2004	101,956,760	9,333,676
	Blizzard	April 1-6, 2003	111,357,227	10,237,791

Sensor Locations on Linked Data. Once sensor data is encoded in RDF and published as Linked Data, the next step is to leverage the vast spatial information already present on Linked Data. GeoNames provides the type of spatial data necessary not only to relate user-friendly location names to coordinate information, but also to associate contextual information such as region containment and distance from location. Fig. 1 shows the overall structure of our datasets and the relationships between them, including links to GeoNames.
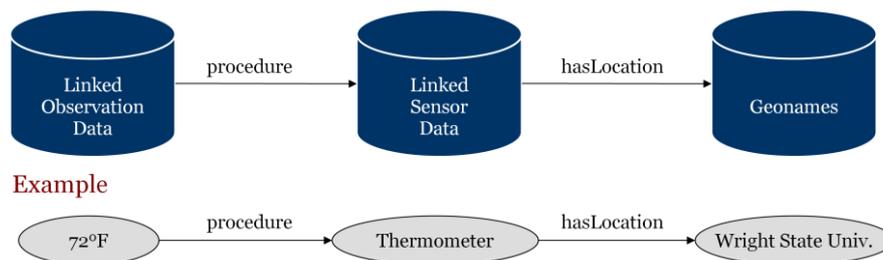

Fig. 1. Relationships between sensor datasets on Linked Data

For each sensor in our knowledge base, we use the *findNearby*[5] service provided by GeoNames to determine the geographically closest named location, or feature, within the GeoNames dataset. This location is then linked with a sensor through the

[5] http://www.geonames.org/export/web-services.html#findNearby

'near' relationship. This relationship describes not only the location of the sensor, but also contextual information regarding the sensor's distance from the location.

GeoNames classifies locations according to containment (e.g. Wright State University is within the city of Dayton) as well as feature classes and codes (e.g. the feature class of Wright State University is a "spot, building or farm" and its feature code is "school"). This provides an extensive source of semantic spatial information that allows us to construct an intuitive mechanism for finding sensor data by region. In addition to feature hierarchy, each GeoNames location provides a *nearbyFeature* relationship that links to a set of locations that are near the original location. The *nearbyFeature* relationship provides another way to find locations near a sensor.

In order to encode these relations between a sensor and the nearest GeoNames location, sensors are annotated with a link to *LocatedNearRel*. The *LocatedNearRel* concept encodes information about the 'near' relationship that holds between a sensor and a named location. More specifically, it contains the closest GeoNames location and its distance from the sensor. The structure is illustrated in Fig. 2.
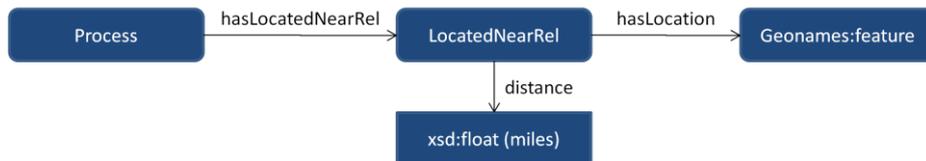

Fig. 2. Concepts and relations linking sensors (or processes) described in LinkedSensorData to features described in GeoNames

3.3 Sensor Discovery Query over Linked Data

With sensor and observation data published with relationships to spatial datasets on Linked Data, discovery simply becomes a matter of querying RDF data. In our implementation, we perform SPARQL queries over a cached version of the relevant portions of Linked Data, particularly named locations in GeoNames and sensor descriptions in *LinkedSensorData* described above. Currently, we support discovery of sensors based on GeoNames locations through two basic operations:

- Find the named location closest to a given sensor
- Find all sensors near a given named location

Fig. 3 shows an example query asking the following question: *Find sensors near Wright State University that can tell me about temperature and precipitation*. The results from this query will include sensors near the specified location and the associated distance between the sensor and location.

4 Sensor Discovery on Semantic Sensor Observation Service

Many people and organizations in the sensors community, both producers and consumers of sensor data, are already heavily invested in the SWE suite of

specifications from the OGC. To support this existing community and evaluate the validity of our approach, we must interoperate with SWE technologies.

The SWE specifications represent a well-reasoned model of the basic structure and characteristics necessary for sensor and observation descriptions. However, as previously discussed, they are syntactic models, and therefore we have chosen to integrate Semantic Web technologies into the existing SWE framework by creating a Semantic Sensor Observation Service, or SemSOS. SemSOS extends the open source 52North SOS implementation [1] with methods for accessing an ontological knowledge base in order to provide queries of high-level features, such as named-locations.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX geonames:<http://www.geonames.org/ontology#>
PREFIX om-owl:<http://knoesis.wright.edu/ssw/sensor-observations.owl#>
PREFIX weather: <http://knoesis.wright.edu/ssw/weather.owl#>

SELECT DISTINCT ?sensor ?dist
WHERE {
        ?sensor rdf:type om-owl:System .
        ?sensor om-owl:hasLocatedNearRel ?near .
        ?sensor om-owl:parameter weather:AirTemperature .
        ?sensor om-owl:parameter weather:Precipitation .
        ?near om-owl:distance ?dist .
        ?near om-owl:hasLocation ?location .
        ?location geonames:name "Wright State University" .
};
```

Fig. 3. Example discovery query of LinkedSensorData

4.1 Overview of SemSOS

52North's SOS implementation is designed to be highly modular, and adaptable to arbitrary suitable sensor data sources, transport protocols, etc. The larger enclosed box in Fig. 4 shows the high-level architecture of the 52North SOS.

The Visualization Layer shown in Fig. 4 is not part of the SOS itself, but rather corresponds to external clients that interact with the SOS, such as publishers or consumers. The Presentation Layer of 52North's architecture defines the SOS's interface to the outside world. The default implementation is an HTTP server, but this can be replaced to support other transport mechanisms and protocols. The Business Layer receives requests from the Presentation Layer, handles them as appropriate, and returns a response. The Business Layer contains the logic for decoding requests and encoding responses, both in SWE formats. The main entry-point from the Presentation Layer is the *RequestOperator* object, which validates incoming requests, determines the type of request, and dispatches accordingly. Each operation supported by the SOS (*GetCapabilities*, *GetObservation*, etc.) is embodied by a Listener object which handles the corresponding incoming request (resp. *GetCapabilitiesListener*, *GetObservationListener*, etc.). The Listener objects may be configured externally during deployment of the service. The individual Listeners handle high-level translation of the request into an internal format which is then used to query the

respective object in the Data Layer and compose the response. The final layer of the 52North architecture is the Data Layer. The Data Layer is an abstraction of a sensor data source through Data Access Objects (DAO). Each DAO represents a particular interface to the sensor data from the point of view of one of the SOS's operations. For each Listener object in the Business Logic Layer, there is a corresponding DAO object in the Data Layer. The DAO objects are used by their respective Listener objects to obtain the data pertaining to a query. The abstraction provided by the DAOs and the Data Layer is what allows the 52North's SOS implementation to be so easily adapted to new sources of sensor data. For each operation that must be supported, all that is required is a new DAO that works with the data source. The default implementation shipped with 52North uses a PostGIS database with a custom database schema to store observation data, while sensor descriptions are stored on the file system in XML files (using SensorML).
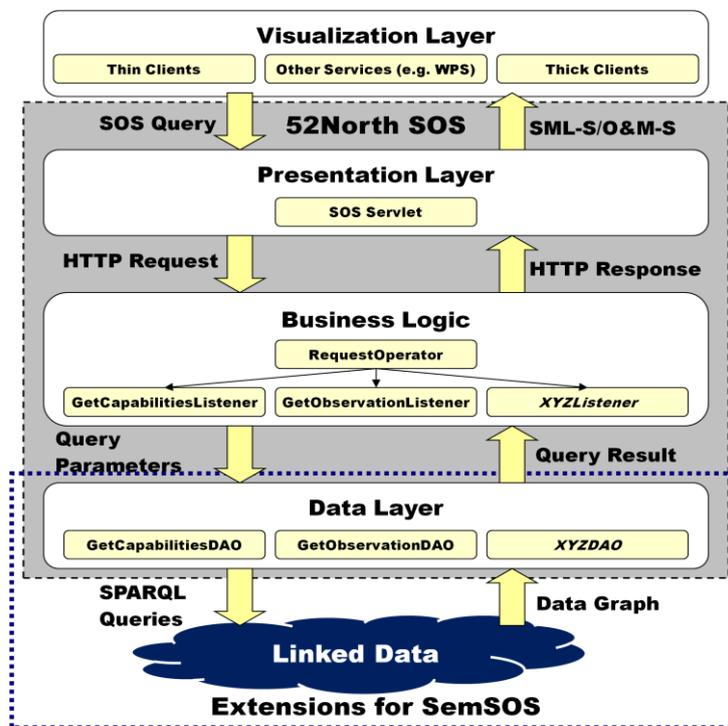

Fig. 4. SemSOS extensions to 52North SOS Architecture

The box surrounding the bottom third of Fig. 4 denotes the extensions made to 52North's SOS in order to implement SemSOS. The modular nature of the 52North implementation allowed us to leave the request routing, encoding/decoding, and similar details in place, while replacing the data access implementation with our own. The DAOs for the operations specified in the SOS core profile (*GetCapabilities*, *etc.*) were replaced with implementations that support access to sensor data on Linked Data.

Specifically, SemSOS uses the RDF2Go[6] and Sesame[7] libraries to access the *LinkedSensorData*. The sensor descriptions are accessed via SPARQL queries that are generated from the incoming SOS query parameters. In order to generate the SPARQL queries, the syntactic form of the SOS query parameters (such as date, time, magnitude, etc.) are transformed into triple patterns conforming to the O&M-OWL ontology. In addition, query filters (such as location, comparison operators, etc.) are transformed into SPARQL-style filters and relational operations.

The result of a SPARQL query evaluates to a set of triples representing an RDF graph, annotated with concepts from O&M-OWL. This graph is then transformed into the internal 52North result structure and returned to the Business Logic Layer. Now, the previous translation to convert SOS queries into SPARQL must be performed in reverse. O&M-OWL concepts instantiated within RDF triples are translated into the original XML encoding of O&M.

The results of SemSOS client queries are valid SOS results. SemSOS also provides richer semantic interoperability for clients that are semantically-aware through semantic annotation of the SWE result documents with ontology terms. This is achieved by using model references, as described in section 3.1.

The integration of SemSOS with Linked Data is achieved in several ways. The global use of model references as identifiers in all SWE query and response documents allows external clients to access the data on Linked Data. In addition, the sensor registry exists as an RDF graph stored as Linked Data.

4.2 Sensor Discovery Service Extension of SemSOS

The final piece of our framework is the support of discovery for SWE clients. Accomplishing this requires that we provide a method for exploiting the expressive nature of our datasets. Just as important, however, is the ability to access this data through a SWE-compatible interface, such as the GetCapabilities operator of the SemSOS service. Current SWE catalog services periodically harvest information from the capabilities documents returned by SOS services' GetCapabilites requests. We have implemented an example SemSOS registry that shows how a SWE catalog service could be extended to make use of semantic model references to Linked Data.

The response to a *GetCapabilities* query is an XML document describing all the information provided by the SemSOS service through a set of *offerings*. Each *offering*, which often represents a "constellation" of sensors, includes information about related *procedures* (sensors), *parameters* (phenomena), and *features of interest*. A *feature of interest* is intended to represent an identifiable ("real-world") object or event about which the sensor system is making observations. We encode the locations defined in GeoNames as *featureOfInterest* model references in resulting SemSOS GetCapabilities documents.

These model references allow for sensor discovery through a query over Linked Data. For example, the SemSOS discovery service can take the URI of a *featureOfInterest* and use it within a query. An excerpt from an example *GetCapabilities* response document is shown in Fig. 5.

```
<sos:Capabilities xmlns:sos="http://www.opengis.net/sos/1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:gml="http://www.opengis.net/gml" xmlns:xlink="http://www.w3.org/1999/xlink" version="1.0.0"
    xsi:schemaLocation="http://www.opengis.net/sos/1.0 http://schemas.opengis.net/sos/1.0.0/sosAll.xsd">
 <sos:Contents>
  <sos:ObservationOfferingList>
   <sos:ObservationOffering gml:id="C1988">
    <gml:name>C1988</gml:name>
    <gml:boundedBy>
     <gml:Envelope srsName="urn:ogc:def:crs:EPSG:4326">
      <gml:lowerCorner>-88.1535 43.0857</gml:lowerCorner>
      <gml:upperCorner>-88.1535 43.0857</gml:upperCorner>
     </gml:Envelope>
    </gml:boundedBy>
    <sos:time />
    <sos:procedure xlink:href="http://knoesis.wright.edu/ssw/System_C1988" />
    <sos:observedProperty xlink:href="http://knoesis.wright.edu/ssw/weather.owl#WindDirection" />
    <sos:observedProperty xlink:href="http://knoesis.wright.edu/ssw/weather.owl#RelativeHumidity" />
    <sos:observedProperty xlink:href="http://knoesis.wright.edu/ssw/weather.owl#AirTemperature" />
    <sos:observedProperty xlink:href="http://knoesis.wright.edu/ssw/weather.owl#DewPoint" />
    <sos:observedProperty xlink:href="http://knoesis.wright.edu/ssw/weather.owl#WindSpeed" />
    <sos:observedProperty xlink:href="http://knoesis.wright.edu/ssw/weather.owl#WindGust" />
    <sos:observedProperty xlink:href="http://knoesis.wright.edu/ssw/weather.owl#Precipitation" />
    <sos:featureOfInterest xlink:href="http://sws.geonames.org/5248611/" />
   </sos:ObservationOffering>
  </sos:ObservationOfferingList>
 </sos:Contents>
</sos:Capabilities>
```

Fig. 5. Example GetCapabilites response document

The *sos:procedure*, *sos:observedProperty* and *sos:featureOfInterest* fields illustrate how we encode model references into standard SWE documents. Of particular interest for discovery is the *sos:featureOfInterest* attribute which references a particular GeoNames feature. A client or discovery service wishing to make use of this reference simply has to retrieve the model reference, perhaps through an XPath[8] expression, such as: *//sos:featureOfInterest/attribute::xlink:href*.

After retrieving the referenced GeoNames features, a SPARQL query over Linked Data retrieves metadata about the specified location. In this case, we are interested in the name of a given location. Fig. 6 shows an example of such a query.

```
PREFIX geonames:<http://www.geonames.org/ontology#>
SELECT DISTINCT ?loc_name
WHERE {
    <http://sws.geonames.org/4528766/> geonames:name ?loc_name .
};
```

Fig. 6. SPARQL query to determine reference location names

Using this approach, we have implemented a prototype SemSOS discovery service which uses *LinkedSensorData* as a registry for sensor information. In particular, a consumer of the service may input a named location and find all sensors and SemSOS services which reference that location as a feature of interest. A discovery request for

---

[8] http://www.w3.org/TR/xpath

our prototype discovery service takes the form of a REST [16] query. An example query is given in Fig. 7 and an example response giving a list of GeoNames features matching the requested query is given in Fig. 8.

```
http://knoesis.wright.edu/sosdis/discovery?q=mapping&name=Wright%20State%20University
```

Fig. 7. Example discovery query

```
<?xml version="1.0" encoding="UTF-8"?>
<GeoNames xmlns="http://knoesis.wright.edu/discovery">
 <GeoName>
  <GeoName>Wright State University</GeoName>
  <URI>http://sws.geonames.org/4528766/></URI>
  <Services>
   <Service>
    <SOS>http://knoesis1.wright.edu/WSUSOSv2/sos</SOS>
    <Sensor>http://knoesis.wright.edu/ssw/System_C1988</Sensor>
    <Sensor>http://knoesis.wright.edu/ssw/System_C1989</Sensor>
   <Service>
  </Services>
 </GeoName>
</GeoNames>
```

Fig. 8. Example discovery response

5 Related Work

The drive to integrate Sensor Web and Semantic Web technologies has been gaining momentum for the past few years. Only recently, however, have we seen the emergence of sensor data on Linked Data. We believe that this integration provides a solid framework for sensor discovery on the Web. Despite this recent emergence, there has already been much discussion on this issue which provides evidence for the validity of our approach. A few examples of such work are described below.

Le-Phouc and Hauswirth [17] have developed an infrastructure, called SensorMasher, for publishing sensor data on Linked Data and a user interface for exploring sensor data and building Web mashups. SensorMasher provides the ability for non-technical users to access and manipulate sensor data on the Web in an intuitive and useful fashion.

Sequeda and Corcho [18] have introduced the concept of Linked Stream Data, which describes how Linked Data principles can be applied to stream data generally, and streaming sensor data specifically. This is an important discussion since most data on Linked Data is static. Sensor data has several attributes that set it apart from the majority of data on Linked Data. For example, sensor data is dynamic (streaming), primarily numerical (phenomenal measurements), highly reliant on spatiotemporal properties, and is often noisy, untrustworthy, inaccurate, and incomplete.

Page et al. [18] have designed a high-level API for semantic mashups and web applications using sensor observations from the Channel Coastal Observatory in the UK. This implementation is based on three objectives: (1) to publish sensor

observations as Linked Data, (2) to access sensor observations through REST services that support GML schema, and (3) to support clients familiar with either Linked Data or GML. This work is probably the most similar to our own, since we are also utilizing both Linked Data and OGC technologies in order to provide the benefits of Semantic Web to those clients familiar with OGC. The distinction is that we have generated large datasets that are now on Linked Data and have extended the OGC Sensor Observation Service to support sensor discovery queries.

In addition to providing sensor data as Linked Data, there has also been work on supporting semantics and sensor discovery within SWE. Janowicz et al. [21] are designing a semantic enablement layer within the SWE standards. Also, a Sensor Instance Registry (SIR) [12] has been developed as part of the OSIRIS project. Its goal is to support discovery of individual sensors and SWE services that encapsulate them. SIR uses a method similar to our prototype registry for harvesting sensor information from GetCapabilities documents. It handles discovery queries via a custom XML-based syntax. On the backend, SIR uses the SWEET ontology[9] for basic disambiguation, but does not expose model references or other semantic information.

6 Conclusion and Future Work

We have introduced a semantic sensor network middleware that allows for effective discovery of sensors on the Web. Specifically, intuitive discovery via named locations is shown to follow from leveraging the power of the Semantic Web and the existing datasets found on Linked Data.

However, there are many opportunities to improve sensor discovery. A reasonable extension to our work on sensor discovery through named-locations involves leveraging the hierarchical relationships found in geographic datasets such as GeoNames and LinkedGeoData. The approach described by Jain et al. [19] that uses SPARQL query rewriting to determine spatial relationships and containment would fit naturally with our use of Linked Data and allow queries about named regions (cities, states, etc.) in addition to low-level features.

Currently, our approach works with fixed-location sensors, but ignores the large and growing number of mobile sensors. Mobile extensions might involve linking locations to observations as well as sensors or providing sample time relations to sensor locations. This may provide another opportunity for incorporating links to LinkedGeoData, which contains finer-grained entities, such as traffic lights and roads.

Even in its current prototypical state, our semantic sensor network middleware approach has realized the important use-case of sensor discovery on the Web. By leveraging Linked Data as a sensor registry and integrating with existing standards, we have shown that practitioners of both Sensor Web and Semantic Web can participate and benefit.

---

[9] http://sweet.jpl.nasa.gov/

References

1. 52North Sensor Observation Service, http://52north.org/maven/project-sites/swe/sos/index.html
2. Sensor Observation Service, http://www.opengeospatial.org/standards/sos
3. GeoNames, http://www.geonames.org/
4. LinkedGeoData, http://linkedgeodata.org
5. Linked Data, http://linkeddata.org/
6. Bizer, C., Heath, T., and Berners-Lee, T.: Linked Data – The Story So Far. International Journal on Semantic Web and Information Systems, 5(3):1–22, 2009.
7. Botts, M., et al.: OGC Sensor Web Enablement: Overview and High Level Architecture (OGC 07-165), Open Geospatial Consortium white paper, 28 Dec. 2007.
8. Henson, C., Pschorr, J., Sheth, A., and Thirunarayan, K.: SemSOS: Semantic Sensor Observation Service. In Proceedings of 2009 International Symposium on Collaborative Technologies and Systems (CTS 2009). Baltimore, MD (2009)
9. Raskino, M., Fenn, J., and Linden, A.: Extracting Value From the Massively Connected World of 2015, Gartner Research, 1 April 2005, http://www.gartner.com/resources/125900/125949/extracting_valu.pdf
10. Sheth, A., Henson, C., and Sahoo, S.: Semantic Sensor Web, IEEE Internet Computing, July/August 2008, p.78-83.
11. W3C Semantic Sensor Network Incubator Group, http://www.w3.org/2005/Incubator/ssn/
12. Jirka, S., Broring, A., and Stasch, C.: Discovery Mechanisms for Sensor Web in Sensors, pp. 2661-2681, (2009).
13. Observations and Measurements (O&M), http://www.opengeospatial.org/standards/om
14. Semantic Annotations for WSDL, http://www.w3.org/2002/ws/sawsdl/
15. MesoWest, http://mesowest.utah.edu/index.html
16. Fielding, R. T.: Architectural Styles and the Design of Network-based Software Architectures, Doctoral dissertation, University of California, Irvine (2000)
17. Le-Phuoc, D., Hauswirth, M.: Linked open data in sensor data mashups, In Second International Workshop on Semantic Sensor Networks Workshop (SSN2009), in conjunction with the 8th International Semantic Web Conference (ISWC2009), October 26, 2009.
18. Sequeda, J. and Corcho, O.: Linked Stream Data: A Position Paper, In Second International Workshop on Semantic Sensor Networks Workshop (SSN2009), in conjunction with the 8th International Semantic Web Conference (ISWC2009), October 26, 2009.
19. Page, K., De Roure, D., Martinez, K., Sadler, J., and Kit, O.: Linked Sensor Data: RESTfully serving RDF and GML, In Second International Workshop on Semantic Sensor Networks Workshop (SSN2009), in conjunction with the 8th International Semantic Web Conference (ISWC2009), October 26, 2009.
20. Jain, P., Yeh, P.Z., Verma, K., Henson, C., and Sheth, A.: SPARQL Query Re-writing for Spatial Datasets Using Partonomy Based Transformation Rules, In Third International Conference on Geospatial Semantics (GeoS 2009), Mexico City, Mexico, December 3-4, 2009
21. Janowicz, K. Schade, S., Bröring, A., Keßler, C., and Stasch, C.: A Transparent Semantic Enablement Layer for the Geospatial Web, Terra Cognita 2009 Workshop, in conjunction with the 8th International Semantic Web Conference (ISWC 2009), October 26, 2009.