

Sensor Observation Service Semantic Mediation: Generic Wrappers for In-Situ and Remote Devices

Manuel A. Regueiro¹, José R.R. Viqueira^{1(✉)}, Christoph Stasch²,
and José A. Taboada¹

¹ Computer Graphics and Data Engineering Group (COGRADE),
Centro de Investigación en Tecnoloxías da Información (CITIUS),
Universidade de Santiago de Compostela (USC), Santiago de Compostela, Spain
{manuelantonio.regueiro,jrr.viqueira,manel.cotos}@usc.es

² 52° North Initiative for Geospatial Open Source Software GmbH,
Martin-Luther-King-Weg 24, 48155 Muenster, Germany
c.stasch@52north.org

Abstract. In-situ and remote sensors produce data that fit different data modeling paradigms, namely, Entity/Relationship paradigm for the former and Multidimensional Array paradigm for the latter. Besides, different standardized data access services are used in practice. Therefore their integrated access is still a major challenge. This paper describes a solution for the development of generic semantic data access wrappers for observation datasets generated by in-situ and remote sensing devices. Those wrappers are key components of data mediation architectures designed for the semantic integrated publishing of observation data.

Keywords: Semantic mediation · Interoperability · Data integration · Observation data · Environmental data

1 Introduction

The amount of environmental observation datasets generated nowadays is increasing due to the advances in sensing technologies. In-situ devices, like meteorological stations, generate data that fit well the Entity/Relationship paradigm and relevant relational technologies. Remote devices, like radars, generate array data, which are generally managed with ad-hoc implementations on top of standardized array file formats.

This article is based upon work from COST Action KEYSTONE IC1302, supported by COST (European Cooperation in Science and Technology). It has been partially funded by the Galician Government (Xunta de Galicia) and FEDER funds of the EU under the Consolidation Program of Competitive Research Units (Network ref. R2014/007).

The implementation of Spatial Data Infrastructures (SDIs) demands from data providers standardized data access services. The Open Geospatial Consortium (OGC) proposes the Sensor Observation Service (SOS) specification to provide access to collections of observations. Informally, *Observations* provide values of *Properties* of specific entities (*Feature of Interest - FOI*), which are generated by some observation *Process*. Beyond the above metadata and the observed value, an observation must also record temporal data and some other optional metadata, including the unit of measure (uom), quality information and some other parameters. Mandatory operations of the SOS interface include *DescribeSensor* and *GetObservation*. The former provides a Sensor Modeling Language - SensorML description of a specific *Process*. The later retrieves observation data that matches specific criteria, including filters on space, time and metadata. To minimize the probability of getting an empty result in a *GetObservation* request, the observations of each *Process* of a SOS are grouped into collections called *Offerings*. Mandatory operation *GetCapabilities* provides appropriate metadata of each available *Offering*.

Integrated access to in-situ and remote observation data sources through SOS has already been reported in [10]. A semantic mediation solution of SOS data sources has also been developed as previous work of these authors [9], where a well-known mediator/wrapper architecture [12] is combined with the use of ontologies. Basic SOS related concepts are defined in a *Core Ontology* as specializations of relevant W3C Semantic Sensor Network (SSN) [5] concepts. *Data Source Ontologies* represent SOS metadata of each dataset by specializing relevant concepts of the *Core Ontology*. Data source classes may be annotated with relationships to classes of some well-known top-level application domain ontology like SWEET [8]. The definition of a *Mediator Ontology* enables the expert to specify required semantic integration knowledge, in the form of relationships between global and local concepts. Those relationships are used to determine which data sources must be queried and which criteria has to be used during global *GetObservation* evaluations. The implementation of the wrappers of the different data sources is always ad-hoc. However, many similarities exist between the different relational sensor observation datasets, and the same applies to those recording array observation data.

Based on the above, this paper describes the implementation of two generic data access wrappers: (i) A wrapper for in-situ geospatial observation data sources, recorded in spatial relational DBMSs and (ii) A wrapper for remote geospatial observation data sources, accessible through NetCDFSubset¹ standardized array data services.

The remainder of this paper is organized as follows. Section 2 discusses on some related pieces of work. The design and implementation of the in-situ sensor observation data wrapper is described in Sect. 3. Section 4 is devoted to the remote sensor observation data wrapper. Finally, Sect. 5 concludes the paper.

¹ <http://www.unidata.ucar.edu/software/thredds/current/tds/reference/NetcdfSubsetServiceReference.html>.

2 Related Work

Most of the current SOS implementations are specialized on observations generated by in-situ devices, recorded in relational databases under specific data models (see the 52° North SOS for a representative example²). Only [2] supports array data sources generated by remote sensing devices.

Semantic sensor data discovery and integration are identified as major challenges in [3], in the scope of the Semantic Sensor Web [11] and the Linked Sensor Data. In the *Model Based Mediation* approach for scientific data sources [7], each data source exports its semantics within relevant ontologies and the mediator combines data source ontologies with data integration knowledge provided by the domain expert. An extension of a conventional conceptual model with constructs that incorporate observation semantics is defined in [1]. The result data modeling framework may be used to annotate data sources with observation semantics.

In [4] the semantic annotation of SensorML documents is the base for the semantic registration of sensing devices in SOS services, which enables subsequent semantically integrated access. A semantic SOS (SemSOS) implementation is reported in [6], where sensor data is semantically annotated and transformed to RDF to be recorded with semantic data storage technologies. Next, SPARQL is used to implement SOS requests. It noticed that none of the above approaches intend to provide semantic data mediation between various existing data sources.

3 In-Situ Sensor Observation Data Wrapper

A generic wrapper was developed that enables SOS access to any database of in-situ observations recorded in a spatially enabled DBMS. To illustrate this, let us first describe two real data sources, which were used during the evaluation of the proposed solution.

*Meteorological Stations*³ (Fig. 1(a)): Observation data is generated every 10 min (*10MinutesData*), daily (*DalyData*) and monthly (*MonthlyData*). Each *Measurement* represents the fact that a sensing device (*Sensor*) that measures a given property (*Parameter*) is installed in a *Station* at a given *Elevation* above the soil and an aggregation process (*Function*) is next applied with a given time frequency (*Interval*). *Sensors* are classified by *SensorTypes* whereas *Stations* are integrated in *Networks*.

*CTD Profiles*⁴ (Fig. 1(b)): Each data element (*Data*) records a value, a sea depth level and a reference to a *Measurement*. A *Measurement* references a measured property (*Parameter*) and a *Profile*, which represents the use of a specific *CTD-Device* at a given time instant and at a given location in the sea (*Station*).

A uniform view of any database is provided through a generic data model (See Fig. 2).

² <http://52north.org/communities/sensorweb/sos/index.html>.

³ <http://www2.meteogalicia.es/galego/observacion/estacions/estacions.asp>.

⁴ <http://www.intecmar.org/Ctd/Default.aspx>.

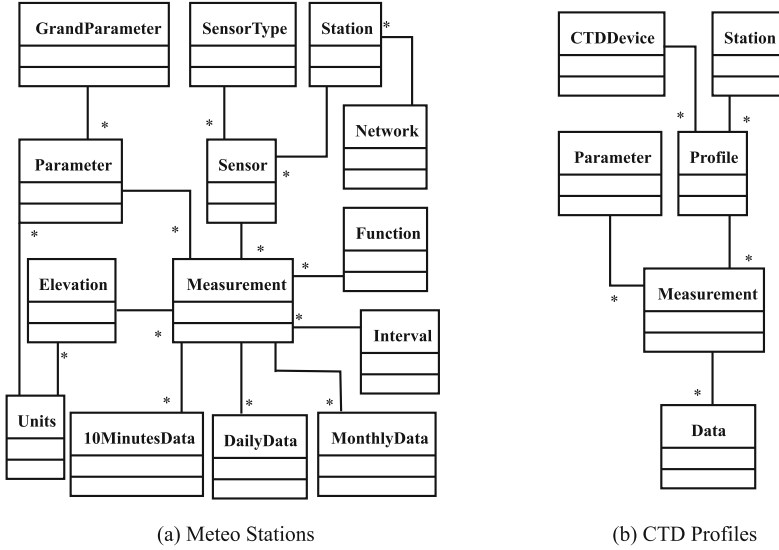


Fig. 1. Conceptual models of meteorological station and CTD data sources.

The model enables both the generation of the required *Data Source Ontology* and the implementation of the SOS *GetObservation* operation. At the top of the diagram, three UML classes enable the representation of the *Process*, *Property* and *FOI* OWL classes that might be available in the data source (*SensorType*, *GrandParameter* and *Network* elements in the case of meteorological stations). The URI of each class is constructed concatenating its identifier (id) with the data source identifier. Relationships with the selected well-known top-level application domain ontology (SWEET in our case) are also provided. Finally, each OWL class has also a reference to its superclass in the model. This enables the creation of OWL class hierarchies from the data source data.

Individuals of the above classes are represented by relevant UML classes. *ProcessDescriptionTime* represents the temporal evolution of the SensorML description of each *Process*. Finally, the observations of each *Process* and *Property* at each *FOI* are represented by UML class *ObservationInstance*. *ObservationInstanceLatest* is used to enable more efficient access to the last observations, which is a typical data need in many real applications.

The SQL code of the *ProcessInstance* view for the data source of meteorological stations is given below.

```
SELECT CAST(p.id AS VARCHAR)||"_"||replace(p.name, " ", "-")||_||
        CAST(e.id AS VARCHAR)||"_"||replace(e.elevation, " ", "-") AS id,
        CAST(gp.id AS VARCHAR)||"_"||gp.name AS propertyClass
FROM Paramter AS p, GrandParameter AS gp, Measurement AS m, Elevation AS e
WHERE p.grandParameter=gp.id AND m.parameter=p.id AND m.elevation=e.id
```

Identifiers are generated concatenating appropriate keys of the database elements with other attributes that can be better interpreted by humans. Thus,

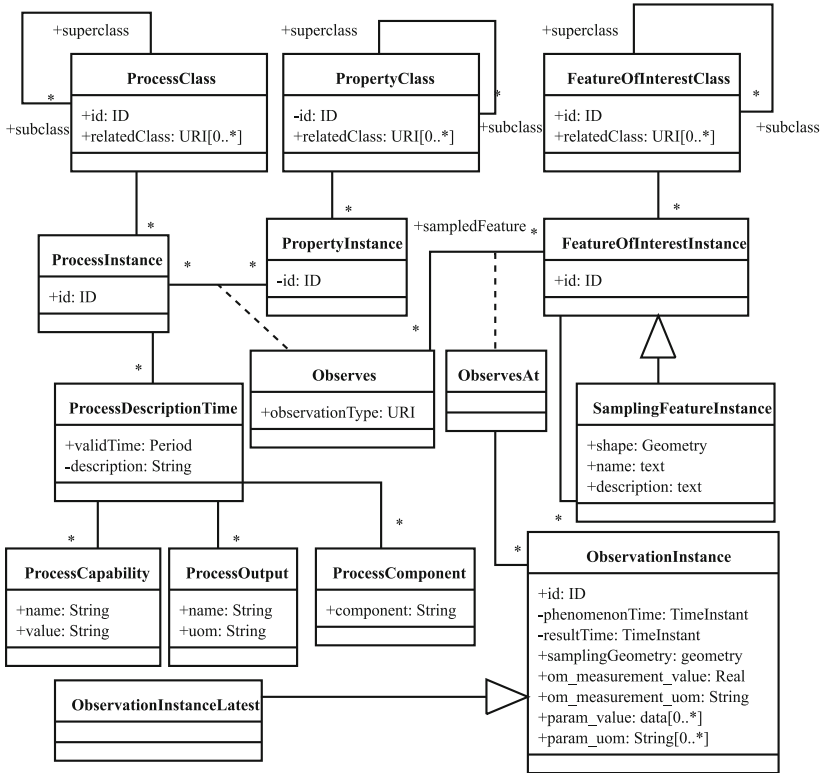


Fig. 2. Generic conceptual model for in-situ observation databases.

Parameter “Temperature” (id = 25), measured at “10 m’ (Elevation identifier 15) has identifier “25_Temperature_15_10-m”.

A *GetObservation* request that retrieves all the observations of a *Property* with identifier *prop* generated by a *Process* with identifier *proc*, during the period defined by instants *s* and *e* at FOIs located inside a given rectangle *b* is implemented with the following SQL statement⁵.

```

SELECT oi.*
FROM ObservationInstance oi JOIN
    SamplingFeatureInstance sfi ON (oi.foi = sfi.id)
WHERE oi.process = proc AND oi.property = prop
    AND oi.phenomenonTime BETWEEN s AND e
    AND st_intersects(b, sfi.shape)

```

The above initial implementation offered very slow response times. This is due to the fact that potential indexes of the underlying database are not used, because of the way identifiers are constructed. To overcome this problem, the application domain expert must provide the positions inside each identifier

⁵ Spatial SQL standard ISO/IEC 13249-3:2011 must be supported by the underlying DBMS.

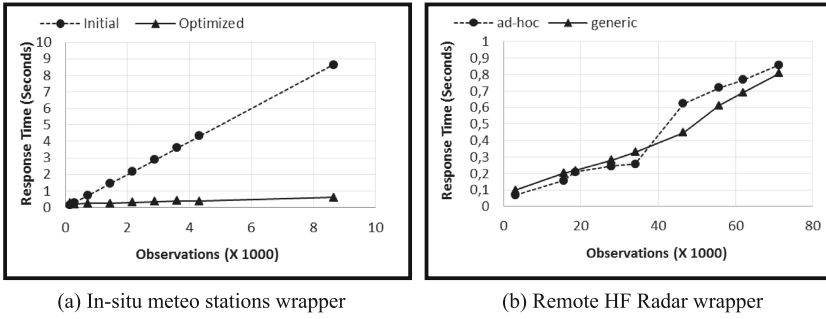


Fig. 3. Performance evaluation.

occupied by key attributes (indexed ones). Thus, the restriction “oi.property = ‘25_Temperature.15_10-m’” may be replaced by a more efficient “oi.paramId = 25 and oi.elevId = 15”. The gain in performance is shown in Fig. 3(a).

4 Remote Sensor Observation Data Wrapper

A generic wrapper was developed that enables the semantically integrated access to array datasets produced by remote sensors and published through NetCDF-Subset services. A specialization of the *Raster Core Ontology*, whose main elements are depicted in Fig. 4, is used by the expert to provide required metadata of each such dataset.

Processes that generate the array data are represented by individuals of core#Process. Each *Offering* of the data source will be defined normally as

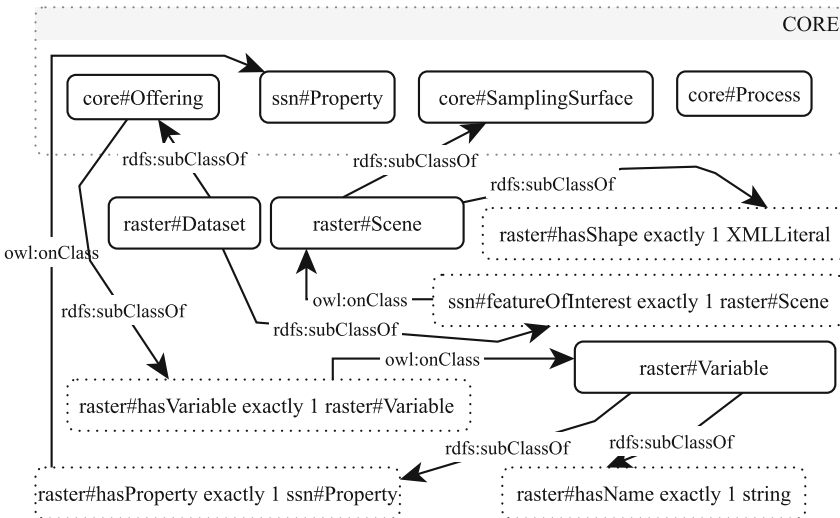


Fig. 4. Raster core ontology.

a subclass of *core#Offering*, specifying with a relevant restriction the reference to its *Process* and with relevant annotation the reference to the specific catalog of the specific THREDDS data server. Variables of the server are defined as individuals of *raster#Variable*, referencing their related SOS *Property*. An algorithm is periodically executed to update such ontology with metadata obtained from the THREDDS data server, which is required to solve future *GetCapabilities* and *GetObservation* requests.

SOS *GetCapabilities* requests are implemented using SPARQL over the above ontology. *GetObservation* requests are solved in two steps. First, a SPARQL query is executed to obtain the relevant *raster#Dataset* classes of the ontology and next a *NetCDFSubset* request is performed for each such dataset to obtain the required array data. Regarding performance evaluation, it is noticed that the a main difference between the current generic implementation of the wrapper and an ad-hoc one would be given by the time to access the ontology. However, such time is two low in comparison to the time to access the datasets. This comparison between the generic and ad-hoc implementations is given in Fig. 3(b).

5 Conclusion

The design and implementation of generic data access wrappers for in-situ and remote sensor observation data sources was discussed. Those wrappers are key components of a mediator/wrapper architecture for sensor observation semantic data mediation. Generic models and ontologies are designed and based on them SOS operations are implemented. The expert can concentrate now on semantic issues related to the datasets, decreasing this way the development cost of data wrappers, without a sensitive impact in the system performance.

References

1. Bowers, S., Madin, J., Schildhauer, M.: A conceptual modeling framework for expressing observational data semantics. In: Li, Q., Spaccapietra, S., Yu, E., Olivé, A. (eds.) *Conceptual Modeling - ER 2008*. LNCS, vol. 5231, pp. 41–54. Springer, Heidelberg (2008)
2. Bridger, E., Bermudez, L.E., Maskey, M., Rueda, C., Babin, B.L., Blair, R.: *Oostethys - open source software for the global earth observing systems of systems*. In: *American Geophysical Union Fall 2009 Meeting* (2009)
3. Bröring, A., Echterhoff, J., Jirka, S., Simonis, I., Everding, T., Stasch, C., Liang, S., Lemmens, R.: *New generation sensor web enablement*. *Sensors* **11**(3), 2652 (2011)
4. Bröring, A., Maué, P., Janowicz, K., Nüst, D., Malewski, C.: *Semantically-enabled sensor plug & play for the sensor web*. *Sensors* **11**(8), 7568 (2011)

5. Compton, M., Barnaghi, P., Bermudez, L., García-Castro, R., Corcho, O., Cox, S., Graybeal, J., Hauswirth, M., Henson, C., Herzog, A., Huang, V., Janowicz, K., Kelsey, W.D., Phuoc, D.L., Lefort, L., Leggieri, M., Neuhaus, H., Nikolov, A., Page, K., Passant, A., Sheth, A., Taylor, K.: The SSN ontology of the W3C semantic sensor network incubator group. *Web Semant. Sci. Serv. Agents World Wide Web* **17**, 25–32 (2012)
6. Henson, C.A., Pschorr, J.K., Sheth, A.P., Thirunarayan, K.: Semos: semantic sensor observation service. In: *Proceedings of the 2009 International Symposium on Collaborative Technologies and Systems, CTS 2009*, pp. 44–53. IEEE Computer Society (2009)
7. Ludascher, B., Gupta, A., Martone, M.: Model-based mediation with domain maps. In: *17th International Conference on Data Engineering, Proceedings*, pp. 81–90 (2001)
8. Raskin, R.G., Pan, M.J.: Knowledge representation in the semantic web for earth and environmental terminology (SWEET). *Comput. Geosci.* **31**(9), 1119–1125 (2005)
9. Regueiro, M.A., Viqueira, J.R., Stasch, C., Taboada, J.A.: Semantic mediation of observation datasets through sensor observation services. Technical report, CITIUS, Universidade de Santiago de Compostela, January 2016
10. Regueiro, M.A., Viqueira, J.R., Taboada, J.A., Cotos, J.M.: Virtual integration of sensor observation data. *Comput. Geosci.* **81**, 12–19 (2015)
11. Sheth, A., Henson, C., Sahoo, S.: Semantic sensor web. *IEEE Internet Comput.* **12**(4), 78–83 (2008)
12. Wiederhold, G.: Mediators in the architecture of future information systems. *Computer* **25**(3), 38–49 (1992)