Domain-Independent Data Processing in an Ontology Based Data Access Environment Using the SOSA Ontology

Cornelis Bouter[1], Han Kruiger[1] and Jack Verhoosel[1]

[1] *TNO, Department Data Science, Anna van Buerenplein 1, 2595DA The Hague, The Netherlands*

Abstract

Data pre-processing, analysis, and quality checks are constantly tailored to a specific data set. Ontology-Based Data Access (OBDA) can provide the interoperability layer to apply the same procedure on heterogeneous data sets. The SOSA ontology, for example, provides a domain-independent generalization of sensor measurements, and is already employed in various OBDA applications. Data pre-processing procedures or visualization tools that operate on the SOSA structure can be generally applicable if the data is also structured following the ontology. We have developed a tool to show how to apply simple data analysis and visualizations using SPARQL queries generated real-time. The tool was initially centered around the horticultural domain, but in this demonstration we show how to generalize the technique across domains. The demonstration therefore contributes to the OBDA goals of enabling data quality verification and data analysis by presenting how to apply the same interoperability layer across domains.

Keywords

ontology based data access, semantic sensor network ontology, SOSA ontology, interoperability, data analytics,

1. Introduction

It is a well-known problem that sensors expose their data in different formats and with various meanings. To tackle this problem and use their measurements for data analysis, the naming and formatting of the measured parameters need to be aligned with each other. A promising strategy for this problem is the definition and use of a common model or ontology that provides this alignment, usually called Ontology-Based Data Access (OBDA). The structure provided by an ontology can be used as a way of pre-processing the data for further analysis. Another part of this strategy is that of providing insight in the quality of the data in terms of, e.g., indicating incorrect values, missing values, unreasonable outliers, and time series misalignment.

In the horticultural domain, for example, many different competitors offer sensors that measure temperature and humidity (T/Rv sensors) in a greenhouse. A T/Rv-sensor usually is small such that it can be easily positioned at a specific location to produce local measurements. In

FOIS 2021 Demonstrations, held at FOIS 2021 - 12th International Conference on Formal Ontology in Information Systems, September 13-17, 2021, Bolzano, Italy

cornelis.bouter@tno.nl (C. Bouter); han.kruiger@tno.nl (H. Kruiger); jack.verhoosel@tno.nl (J. Verhoosel)
https://www.hankruiger.com (H. Kruiger)
0000-0002-0901-202X (H. Kruiger)

addition, it is cheap and therefore a large set of them can be placed to cover the entire greenhouse. Various other companies offer climate computers that produce a wealth of measurements of different parameters inside the greenhouse, especially focused on indoor climate and outdoor weather conditions, but increasingly also around the status of the crop and energy usage.

The different formats inhibit data exchange, comparison, and analysis in various ways:

- Comparing the climate computer data with the T/Rv measurements within a single greenhouse;
- Comparing the T/Rv-measurements of sensors of different manufacturers; and
- Comparing data of different climate computers by different vendors.

To realise the described strategy, we have been working on a Common Greenhouse Ontology[1] (CGO) [1, 2]. This CGO extends the domain-independent SOSA ontology with concepts from the horticultural domain. On top of that, a Data Analysis Facility (DAF) has been developed to achieve further pre-processing of, and provide insight in data to be analyzed. In this demonstration we show how to generalize our approach across domains to contribute to domain-independent data processing and analysis.

2. State of the Art

The fields of Ontology-Based Data Access (OBDA) define an interoperability layer that utilises an ontology for communication among a heterogeneous set of databases [3, 4]. A domain-independent ontology that may function as the interoperability layer is the Semantic Sensor Network (SSN) ontology [5, 6] or its lightweight version the Sensor, Observation, Sample and Actuator (SOSA) ontology [7]. These ontologies have been applied various times from the industry [8] and the IoT environment [9] to our work in the horticultural domain, but without an attempt to generalize the SOSA extension to similar use cases beyond the respective domains. Extending OBDA to, e.g., data quality checks and data analytics has been identified as further research [3].

3. DAF Demonstration

The Data Analysis Facility (DAF) is a tool intended to interpret data structured via the SOSA ontology consisting of several components (fig. 1). The *sensors* provide data in a foreign format which are transformed into RDF by *mappers* [4]. The data is stored into various triple stores in the *interoperability layer* using a common language that uses SOSA, which is in our use case the CGO. The interoperability layer contains a triple store for each data set, e.g., for each set of sensors and for each climate computer in a greenhouse. Each triple store is exposed via a SPARQL endpoint. The implementation is an extension of Apache Jena Fuseki [10, 1]. The *DAF server* retrieves this data using SPARQL, and provides a REST API that offers functionality to create flattened views into the linked data. The *DAF front-end* is a web application that uses the REST API to enable the user to perform analyses on the data.

[1] Available at https://ontology.tno.nl

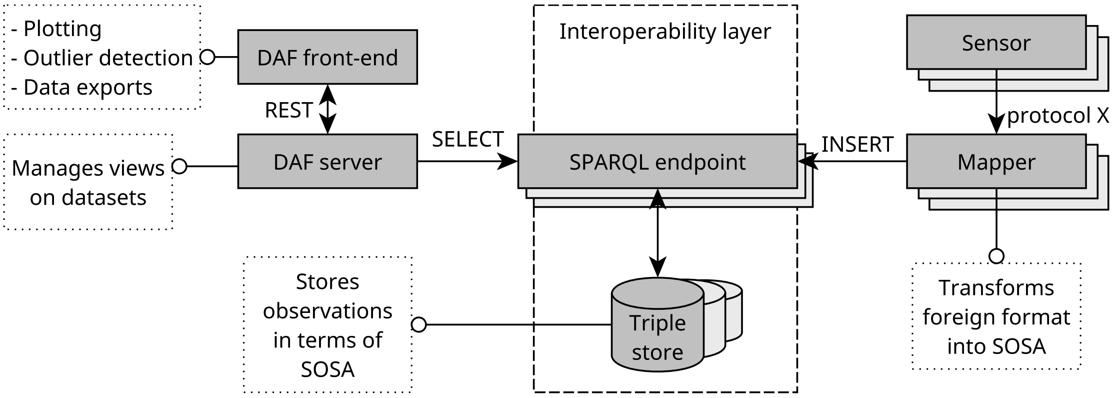

Figure 1: DAF architecture

The differentiating factor we want to demonstrate is that we built a domain-independent interoperability layer despite the domain-specific use case. This is made possible because the ontology strictly separates domain-independent SOSA concepts from domain-specific concepts added by the CGO (fig. 2). The DAF tool SPARQL queries can then retrieve domain-*specific* data through domain-*independent* queries. Our demonstration will consist of these two parts: the ontology design and the SPARQL queries instantiating the data visualizations.

3.1. Ontology

The first component we demonstrate is the ontology model describing the structure of the data available in the triples. The CGO extends the SOSA ontology by providing domain-specific subclasses for, a.o., `sosa:FeatureOfInterest` (FoI) and `sosa:ObservableProperty` (OP). As an example, we show how the CGO [2], and by extension SOSA, represents observations of, e.g., the length of a flower, the photosynthesis of a crop, or the temperature of a greenhouse. We have developed a mapper from T/Rv sensor or climate computer to the data model for various vendors.

The main demonstration then follows on generalizing the CGO structure to other domains, such that the same architecture containing the same SPARQL queries can operate on data from multiple domains. First, we show how measurements in, e.g., industry or education can be represented analogously. A machine is the feature of interest that has its energy consumption (OP) observed. In the education domain the student is the feature of interest whose attendance rate (OP) is observed. This analogy shows that the ontology design structure is sufficient to align a data set with the DAF; namely, a domain-specific extension (fig. 2) of the `sosa:FeatureOfInterest`, the `sosa:ObservableProperty`, and the `sosa:Sensor`. This description can be easily implemented by ontology developers. As an example we show this implementation for another domain, such as industry or education.

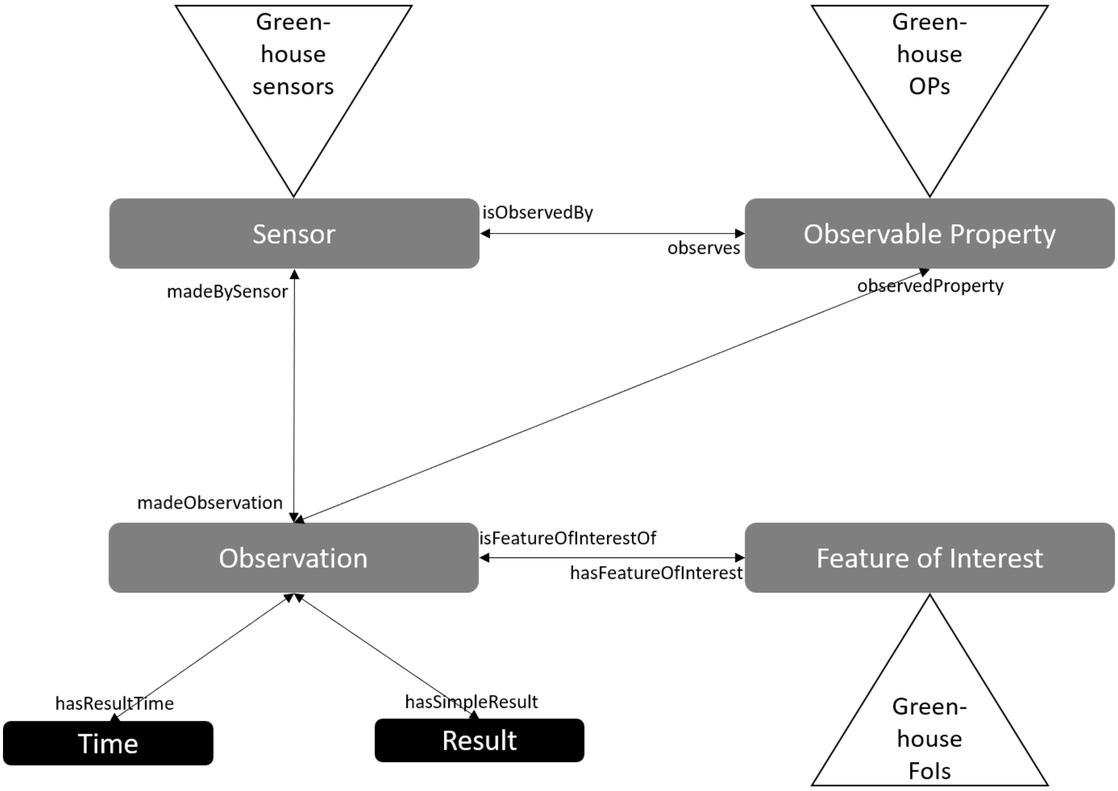

Figure 2: SOSA structure with domain-specific extensions in blue triangles. The R-Tv sensor would be contained in the sensors extension, the temperature and humidity in the observable properties extension, and the greenhouse air in the features of interest extension. Figure adapted from SSN website[2].

3

3.2. DAF Server and Front-End

At this point in the demonstration we have shown that the DAF can be used across domains requiring for each application a domain-specific SOSA extension. The continued demonstration presents a software architecture based on semantic web protocols together with its implementation. Its main takeaway should be that the ontology described in the previous section enables writing of domain-*independent* SPARQL queries to retrieve domain-*specific* data.

As configuration, the DAF application needs one or more SPARQL endpoints that provide access to datasets using SOSA. Upon initialization of a dataset, the DAF issues domain-independent SPARQL queries to retrieve the domain-specific data (fig. 3):

1. Which sensors are in the dataset?
2. Which properties of which features of interest do the sensors observe?

[3] https://www.w3.org/TR/vocab-ssn/

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX sosa: <http://www.w3.org/ns/sosa/>

SELECT DISTINCT
  ?featureOfInterest ?featureOfInterestType ?label ?typeLabel
WHERE {
  ?featureOfInterest rdf:type sosa:FeatureOfInterest .
  ?featureOfInterest rdf:type ?featureOfInterestType .

  OPTIONAL { ?featureOfInterest rdfs:label ?label . }
  OPTIONAL { ?featureOfInterestType rdfs:label ?typeLabel . }
}
```

Figure 3: A domain-independent SPARQL query that retrieves the domain-specific features of interest, their types, and their labels

This domain-specific metadata is presented in the front-end to a user, who can select combinations of sensors and measurements (identified as the combination of a feature of interest and an observable property) that they want to use as model features in an analysis. We show that the SPARQL queries transferring data from the triple store to the DAF server are domain-independent.

For example, a user may select two model features: the temperature of the air in the greenhouse, as measured by sensor X, and the humidity of the air in the greenhouse, as measured by sensor Y. We demonstrate that a SPARQL query is generated on-the-fly based on the selected model features and sensors. The query result is a flattened *view* into the dataset with, in this example, three colums: the (normalized) observation time, the temperature, and the humidity of the greenhouse air.

We conclude the presentation by showing the data quality visualizations in the front-end: an outlier detection analysis visualised through scatter plots and several boxplots. Because the data is flattened, we can leverage the *pandas* [11] and *scikit-learn* [12] packages for the analyses we offer. The user can also download the selected data to apply more sophisticated data analysis techniques. These visualizations can inspire further applications and ideas. During this final part we stress that the visualizations are based on data accessed via the ontology.

4. Discussion & Conclusion

Our demonstration will show a unified view of how to apply the SOSA ontology in sensor data application across domains. The ontology is already being applied in various use cases, but each time it is implemented differently in another architecture despite the similar goal of automatically interpreting heterogenous data. Our work presents the initial version of an architecture that can be employed across use cases involving the SOSA ontology.

The demonstration additionally works towards increased functionality for OBDA systems. Data quality and analytics powered by ontologies were identified by [3] as directions for further research. Our tool demonstrates a direct data pipeline from the ontology based data to data visualizations with underlying outlier detection algorithms. We thereby demonstrate an initial direction for domain-independent data processing techniques.

References

[1] J. Verhoosel, B. Nouwt, R. Bakker, A. Sapounas, A. Slager, A datahub for semantic interoperability in data-driven integrated greenhouse systems, in: Efita Conference 27-29 Juni 2019, Rhodes Island, Greece, 1-6, 2019.

[2] R. Bakker, R. v. Drie, C. Bouter, L. v. Rooijen, S. v. Leeuwen, J. Top, The Common Greenhouse Ontology: an ontology describing components, properties, and measurements inside the greenhouse, in: Efita Conference 2021, 2021.

[3] G. Xiao, D. Calvanese, R. Kontchakov, D. Lembo, A. Poggi, R. Rosati, M. Zakharyaschev, Ontology-based data access: A survey, International Joint Conferences on Artificial Intelligence, 2018.

[4] O. Corcho, F. Priyatna, D. Chaves-Fraga, Towards a new generation of ontology based data access, Semantic Web 11 (2020) 153–160.

[5] M. Compton, P. Barnaghi, L. Bermudez, R. García-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, K. Janowicz, W. D. Kelsey, D. Le Phuoc, L. Lefort, M. Leggieri, H. Neuhaus, A. Nikolov, K. Page, A. Passant, A. Sheth, K. Taylor, The SSN ontology of the W3C semantic sensor network incubator group, Journal of Web Semantics 17 (2012) 25–32.

[6] K. Taylor, A. Haller, M. Lefrançois, S. J. Cox, K. Janowicz, R. García-Castro, D. Le Phuoc, J. Lieberman, R. Atkinson, C. Stadler, The semantic sensor network ontology, revamped., in: JT@ ISWC, 2019.

[7] K. Janowicz, A. Haller, S. J. Cox, D. Le Phuoc, M. Lefrançois, SOSA: A lightweight ontology for sensors, observations, samples, and actuators, Journal of Web Semantics 56 (2019) 1–10.

[8] M. Obitko, V. Jirkovskỳ, Big data semantics in Industry 4.0, in: International conference on industrial applications of holonic and multi-agent systems, Springer, 2015, pp. 217–229.

[9] T. Elsaleh, M. Bermudez-Edo, S. Enshaeifar, S. T. Acton, R. Rezvani, P. Barnaghi, IoT-stream: A lightweight ontology for internet of things data streams, in: 2019 Global IoT Summit (GIoTS), 2019, pp. 1–6. doi:10.1109/GIOTS.2019.8766367.

[10] The Apache Software Foundation, Apache Jena Fuseki, 2021. URL: https://jena.apache.org/documentation/fuseki2/.

[11] T. pandas development team, pandas-dev/pandas: Pandas, 2020. URL: https://doi.org/10.5281/zenodo.3509134. doi:10.5281/zenodo.3509134.

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.