FUNCTIONAL INTEGRATION FOR THE OBSERVATION WEB

Sven Schade, Frank Ostermann and Laura Spinsanti
European Commission – Joint Research Centre, Institute for Environment and Sustainability, Ispra, Italy

Keywords: Sensor web, Semantic integration, Interoperability, Algebra, Functional programming, Forest fire.

Abstract: The integration of sensor-based information has been a research topic for many years. International standards, such as the Sensor Web Enablement (SWE) suite of specifications that will soon be released in its second version, have been developed, and major syntactical and structural challenges have been overcome. Solutions for addressing semantic aspects of interoperability have been suggested, but mature applications are still missing. The advent of user generated content for the geospatial domain, Volunteered Geographic Information (VGI), which can be considered as readings of virtual sensors, makes it even more difficult to establish formal systems for the combination of information that is based on heterogeneous sensing methods. This paper proposes a novel approach of integrating conventional sensor information and VGI, which is exploited in the context of detecting forest fire events. In contrast to common logic-based semantic descriptions, we present a formal system using algebraic specifications as a more elegant, illustrative and straight forward solution.

1 INTRODUCTION

The integration of sensor-based information has been a research topic for many years (Schade, 2005; Sheth et al., 2008; Janowicz et al., 2011). Standards, such as the Sensor Web Enablement (SWE) suite of specifications that is currently under major revision[1] (Bröring et al., 2011), help to establish syntactical and structural interoperability. This includes possibilities for integrating information produced by physical sensors and by environmental simulations (Maué et al., 2011).

Several approaches for addressing semantic interoperability have been proposed. Most notably, the W3C recently released a sophisticated ontology on observations and measurements (Janowicz and Compton, 2010), and a light-weight approach for the semantic enablement of the Sensor Web has been proposed (Janowicz et al., 2011). Still, mature solutions that illustrate the use of such work for solving challenges of geospatial information integration are missing.

During the past decade, we have witnessed a surge of geographic information provided by the public to the public via the internet. Resembling virtual sensors, citizens provide this Volunteered Geographic Information (VGI) (Goodchild, 2007) through the web by posting images or videos (e.g. on Flickr or YouTube), blogging or micro-blogging (Twitter), surveying and updating geographic information (OpenStreetMap), or playing games (FourSquare). Considering the increase in mobile internet access through smartphones and the number of (geo)social media platforms, we can expect the amount of VGI to continually grow in the near future. This new wealth of VGI has several advantages over the traditional, authoritative gathering, maintaining and disseminating of geographic information. First, it is more up-to-date, because a larger number of 'surveyors' reports new information or changes to existing information in near real-time. Second, VGI can be very rich in content, providing already pre-processed information instead of raw data. As information

[1] At this stage (August 3rd, 2011), the SWE Common 2.0 data model already became a standard and the second version of Observations and Measurement is close to its official release, while SensorML 2.0 is still under debate. With respect to service specifications, the SWE common model and the Sensor Observation Service (SOS) are available as standards, while discussions on the Sensor Planning Service (SPS) are still ongoing.

Schade S., Ostermann F. and Spinsanti L..
FUNCTIONAL INTEGRATION FOR THE OBSERVATION WEB.
DOI: 10.5220/0003698404980504
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (SSW-2011), pages 498-504
ISBN: 978-989-8425-80-5

portals such as EyeOnEarth[2] successfully illustrate, the provided information often complements the data coming from traditional sensor networks. The EyeOnEarth portal, which is hosted by the European Environmental Agency, provides access to measurement stations in air and water, but also reports on air quality and water quality that have been generated by laymen.

However, there are clearly several problems associated with VGI. The technology-driven development leads to frequent changes in the data structure, since new platforms emerge, old ones disappear, and prevailing ones modify their user and programming interfaces. Further, VGI frequently is rather unstructured in nature, and quality control proves difficult. Even comparatively well-structured and quality-controlled platforms such as OpenStreetMap have to deal with these issues. Therefore, the integration of VGI with existing sensor networks and spatial data infrastructures is a challenging task. This increased diversity of information channels and provided messages makes it even more difficult to establish formal systems for information combination. A conceptual solution of using SWE for integrating VGI with the Sensor Web has been suggested for creating a (more general) Observation Web (DeLongueville et al., 2010), but again semantic aspects have not been considered explicitly.

In this paper, we propose a novel approach to integrate conventional sensor information and VGI. Contrary to common logic-based approaches, we base our developments on another formalization paradigm from software engineering: *algebraic specification* (Ehrich and Mahr, 1985).

The remainder of this paper is organized as follows. We briefly illustrate the proposed solution to sensor integration in the next section, while we argue for the use of algebraic specifications for ontology engineering and present related work in section 3. The formalization of our approach to sensor integration is provided in section 4. This first application of this approach is based on several assumptions and simplifications, which are discussed in section 5. We conclude the paper with a summary of our main findings and an outline of future work. Throughout the paper, the VGI use case of forest fire detection serves as example.

[2] Official Web portal available at http://www.eyeonearth.eu/ (last accessed on August 3rd, 2011)

2 DESIGN OF THE INTEGRATION APPROACH

We consider the application of specific, stepwise processing of a given raw data set as a core principle. The different layers of value-added information can be illustrated as in Figure 1, where the centre represents the initial content and each additional surrounding layer represents the results of one processing step. For example, the raw data might be air temperature measures (for intervals of one minute), and the first processing step might provide daily averages, the next weekly averages, etc. (Figure 1 a). We may also think of data coming from different sources, for example measured by diverse sensor networks, such as air temperature, wind-direction, cloud-cover and humidity values. In this case, the first processing step might be a merger of pieces of information into a complex measure as the fire risk index (Figure 1 b).

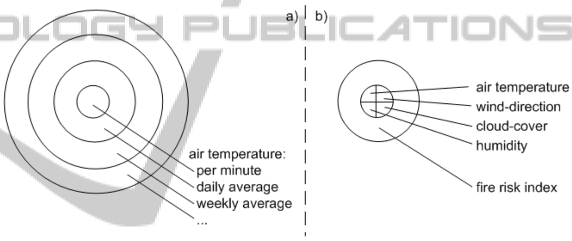

Figure 1: Layers of value-added information, a) averages of air temperature; b) fire risk index.

Alternatively, contents provided by two different sources, e.g. satellite images from the MODIS satellite (a conventional physical sensor) and VGI posts on Flickr (http://flickr.com), may be provided separately. As the information gets processed, resulting layers might overlap. For example, first MODIS images could be analyzed for temperature hotspots, and some of these hotspots might be categorized as forest fires. At the same time, VGI might be analyzed for hotspots as well. In social media, these hotspots could be purely thematic in nature, such as an increase of words like 'fire' in messages, but in the case of sufficiently accurate VGI, the hotspots could correspond to spatio-temporal clusters (Spinsanti and Ostermann, 2011) and subsequently, some of these hotspots might also be categorized as forest fires (Figure 2). Notably, we can arrive at the same kind of (forest fire) event using different information channels.

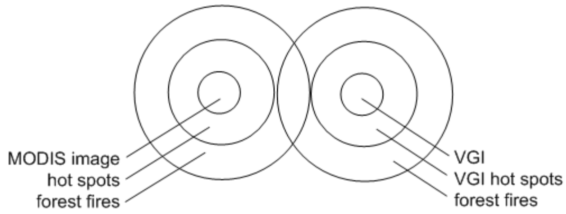

Figure 2: Detecting forest fires using MODIS and VGI.

It is worth noticing the following characteristics of this approach:

1. We can only move from the inner layers to the outside.
2. The information becomes more specialized with distance to the centre, i.e. application specific context is introduced increasingly.
3. Information can only be integrated on a shared layer.

In section 4 we will use these characteristics, together with an ontology for observations, to formalize a system for information integration. Before, i.e. in the next section, we discuss the use of algebra as a tool for (formal) system engineering.

3 ENGINEERING FORMAL SYSTEMS WITH ALGEBRAS

Ontologies have been suggested as the basis for semantic interoperability of information systems (Guarino, 1998). Following Guarino's characterization of an ontology (in the Artificial Intelligence sense), as an "engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words" (Guarino, 1998), assumptions can be stated using any formal theory. We use algebraic specifications of Abstract Data Types (ADTs) (Sommerville, 2007), in which the (algebraic) theory of an ADT describes its abstract behavior, whereas models are given by concrete data types (Ehrich and Mahr, 1985). In other words, we use ADTs together with their (algebraic) specification as ontologies. Additionally, we provide (algebraic) functions to define transitions between ADTs. The resulting formal system will implement the integration approach that has been outlined above.

The decision to use an algebraic approach compared to common approaches, which apply Description Logics or First-Order Logic, for ontology engineering depends on the intended goals. In our case, we face a data integration challenge involving classical sensors and VGI. Encapsulation, as a main feature of ADTs (Sommerville, 2007), provides us with the required abstraction mechanisms. The functional equations that are used in algebraic specification can be directly applied for mapping from sensor- and VGI-specific models to an integrated theory of observations. In order to support clean ontology engineering, the observation theory can even be aligned with an upper-level ontology, as we will see in our examples in section 4. These two possibilities provide clear benefits compared to logic-based approaches, which are for example strong in instance re-classification.

The above mentioned principles relate closely to concepts of No-SQL data bases in information science (Agrawal et al.,2008). Moreover, the history of using algebraic specifications and functional programming for the geospatial sciences dates back at least twenty years. It has been first suggested for user interface design of Geospatial Information Systems (GIS) (Kuhn and Frank, 1991) and soon has been extended to conceptual modeling (Car and Frank, 1995), including the principle of measurement-based GIS (Goodchild, 1999). The explicit use of this approach for (spatio-temporal) ontology engineering dates back at least five years (Schade et al., 2004; Frank, 2007; Kuhn,2007). Recent works direct these ontology developments to sensing (Kuhn, 2009) and data integration (Schade, 2010).

4 FORMALIZATION OF THE INTEGRATION APPROACH

As indicated above, we build our formal system for integrating sensor data using algebraic specifications; in particular, we use the functional programming Haskell (Peyton, 2003) for formalization. As we present the solution directly in an executable language, this corresponds also to the implementation of the desired system.

4.1 Algebraic Specification of the Observation Ontology

Before we can define the transitions between layers (by functions) and introducing data types for intermediate and final integration results, we have to establish an ontology that captures the core principles of observation, i.e. the inner layer of the diagrams presented in the previous section. This should not only cover physical sensing procedures and environmental simulation, but also VGI.

Since a basic observation algebra is already available (Kuhn, 2009), we can re-use ADTs, such as a construct for measurement values, i.e. for raw data that is the result of a measurement (*in Haskell, algebraic data types are introduced by specifying constructor functions. The keyword 'data' is followed by the type name, an equal sign and the constructor function(s). The first element of this function is its name. '|' is used for separating multiple constructer functions for a single ADT*):

```
data Value =  Boolean Bool |
              Count Int |
              Measure Float Unit
              | Category String
```

In addition, we induce an ADT for representing user generated content (UGC), such as a text message, a photo or a audio-video. For simplicity reasons, we include photos and videos by reference, i.e. using http URIs (Coates et al., 2001):

```
data UGC =  Message String |
            Photo URI | Audio URI
            | Video URI
```

An observation consists of either a measurement value or user generated content, each combined with a position and time. We thus extend the basic observation data type with a special construct for VGI:

```
data Observation = Measurement
                   Value Position
                   ClockTime |
                   VGI UGC
                   Position
                   ClockTime
```

For later implementation of our example, we add a construct for MODIS satellite images. Again for simplicity reasons, we define the image as a list of observations, where each observation represents a pixel of the image with its associated values (*we use Haskell type synonyms to give previously defined data types a new name.*):

```
type MODISimage =  [Observation]
```

4.2 Generating Value-Added Information

Now, as the foundations are available, we focus our attention on the transitions from the raw data (innermost layers in figure 1), to any added value, i.e. processed information (outer layers the diagrams). The forest fire example serves for illustrations.

We introduce constructs, which represent the results of a transition (the next outer layer in the diagram), and functions, which represent the actual transition between two layers, such as the creation of hotspots. In analogy to the observations ontology above, the former can be seen as ADTs, which provide unique entry points to each layer, such as hotspots and forest fires. They thereby encapsulate the manifold possibilities in which a single instance might have been produced.

As a first example, we create a type for capturing hotspots as a specific kind of value-added information. Each hotspot represents a set of observations:

```
type Hotspot = [Observation]
```

As an additional benefit of using an executable functional programming language for formalization, all Haskell build-in operations are available for manipulating hotspots (as we will see later). We might define additional operations on hotspots at will, but for the moment, we concentrate only on the function for transitions, i.e. on that function, which generates hotspots from the information that is available on lower layers. We define a constructor function for hotspots in a re-usable manner, i.e. in a way that we can create hotspots out of any observation collection, independent of the nature of the observation. Such reusability is a benefit but is not mandatory for each transition function.

For us, a hotspot is characterized by high spatio-temporal density, and may optionally include a 'filter', such as a specific threshold for a measure or a specific category. We specify the desired minimum density in space or time, and an optional filter as parameters of the hotspot function; we omit the implementing algorithm here, as this is out of the scope of this paper (*Haskell functions can be specified using signature declarations. Here, a function name is followed by '::' and a list of parameters separated by '->'. The last parameter of a signature stands for the output. Parentheses can be used to include functions, such as a filtering function, as parameters.*):

```
hotspots :: Value -> Value ->
            ([Observation] ->
            Observation ->
            [Observation]) ->
            [Hotspot]
```

We follow a similar principle for the next (the forest fire) layer. We use the 'data' construct of Haskell for introducing an ADT, because we want to define special functions on this data type and because we intent to use class instantiation as a tool for ontology alignment later on (see section 4.4):

```
data ForestFire = ForestFire
Hotspot
```

Next, we introduce a function to describe the transition from the lower layers, i.e. from a collection of hotspots to a collection of forest fires. In the example, we categorize all hotspots as forest fires. The implementation also shows the application of common Haskell functions on the 'Hotspot' ADT. We use the 'map' operator, which applies a function (here the Forest Fire constructor function) to all elements in a list:

```
forestFires ::   [Hotspot] ->
[ForestFire]

forestFires hs = map ForestFire
hs
```

e can also use this example to show how specific operations, so called observers can be added to a (newly defined) ADT. The signature of a co-occurrence function for forest fires, which returns a list that contains all collections of forest fires, which overlap in space and time may be defined as:

```
ffCoOccurrence :: [ForestFire] ->
[[ForestFire]]
```

4.3 Common Access to Value-Added Information

The approach outlined above provides a straightforward solution to the integration problem. Information from two separate sources, as for example illustrated for MODIS and VGI in Figure 2 (above), may be merged as soon as the derived information can be provided via a shared ADT. Considering our example, this would be the data type representing forest fires. Co-occurrences of forest fires can now be calculated without required knowledge about the sources that lead to the information about a particular fire.

If we would for example have all MODIS and VGI data available for the 2010 forest fire season in France, then we could find out how many forest fire events were detected by both sources, and how many were detected by only one. Such additional information, which can be derived from using the two sensor sources, MODIS and VGI in this case, can be used to further calibrate the involved sensors and thus to improve measurement quality (Spinsanti, and Ostermann, 2011).

For real-time applications this means that we become able to process (and in particular compare) information about events, such as forest fires, seamlessly, i.e. without any artificial barriers that might have been imposed by diverse use of sensors or user generated content. In this sense, we solved a semantic integration issue in the Observation Web.

It is worth noticing that we remain able to trace back the generation process of a data set. Having co-occurring forest fires identified, we can reveal if a particular instance has been created from a MODIS image or from VGI by uncovering the used constructor function. Among other information, we may get to know if statements of two separate sources confirm or contradict each other.

4.4 Alignment with DOLCE

In order to further specify the semantics of forest fires (as events), we consider an alignment with Foundational Ontology. Based on previous experience, we select Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) (Masolo et al., 2003) as the best candidate. In DOLCE, events are defined as special kinds of perdurants, which extend in time by accumulating different temporal parts. Each perdurant can be associated with a duration in which it occurred.

Again following Kuhn (Kuhn, 2009), DOLCE concepts can be formalized as follows (*the 'class' construct in Haskell introduces type classes for a set of (abstract data) types sharing some behavior. Type classes are named, followed by a parameter for the types belonging to the class. A 'where' clause introduces a block of functions, which capture the shared behaviors of the type class*):

```
class PERDURANTS perdurant where

   getDuration :: perdurant ->
Duration

class PERDURANTS event => EVENTS
event
```

The alignment of the forest fire ADT can be achieved by instantiating the type classes introduced above; we omit the implementation of the duration function:

```
instance PERDURANTS ForestFire
where

instance EVENTS ForestFire
```

5 DISCUSSION OF THE INTEGRATION APPROACH

By using concatenated functions as opposed to (description) logic constructs, the proposed integration approach follows the idea of algebraically specifying GIS and the principle of measurement-based information systems (Goodchild, 1999). The advent of No-SQL data bases (Agrawal et al., 2008) indicates a trend of such solutions even for mainstream IT. On top of these known concepts, we extend the observations ontology with user generated content (VGI in our case) and we apply the approach to the data integration. The encapsulation by abstract data types and the use of functional equations as transformation mechanisms are powerful characteristics of this algebraic approach. It has the potential to solve many of the semantic interoperability problems, which continue to grow with the current trend of user generated content and the ongoing interests for information integration for quality improvement.

Although promising, algebraic solutions are still rare in the Semantic Web and they miss a support in the context of geospatial web applications. This results in some difficulties, such as (i) missing tool support for input and out put data management for the web; (ii) rare examples that illustrate the combined use of logic-based and algebraic specifications for knowledge engineering; and (iii) disconnectivity between the algebraic specification and the Semantic Web community. Research in this direction should be extended in order to avoid focusing solely on logic-based efforts towards a Semantic Sensor Web and running the risk of stagnation.

The presented work lays the function for further investigations. For example others already suggests using Haskell for common error modeling and elegant propagation (Navratil et al., 2008), but further examinations are required in this field, especially exploiting the interplay between the ‘quality’ of VGI and of classical measurements.

Another open issue is the investigation of place. Our approach currently assumes that geospatial components are introduced in form of geographic locations, with precise geometric attributes. VGI however, might merely include vague place names instead of precise locations, introducing uncertainty on the geographical side. The interplay between place and location should be considered as a major aspect to improve the suggested integration approach.

The presented algebra does also not explicitly account for (spatio-temporal) level of detail, i.e. scale. Here lies a major area for improvement, because different information sources are often captured on diverse scales. A functional approach eases scale changes, but still more practical examples, have to be explored.

6 CONCLUSIONS AND OUTLOOK

We have established a formal system of the semantic integration of observation-based information and showed a successful approach to the challenge of integrating them in a forest fire scenario. This illustrates the functionality that we envision for a future Observation Web. The proposed approach still requires maturity. Therefore, we plan to experiment with a more detailed workflow for processing forest fire information, as for example presented in (Ostermann and Spinsanti, 2011). In addition, more cases will be investigated in order to show that our general assumptions hold.

As the current implementation used only parts of a previously developed ontology for observations in Haskell, we are still investigating complete re-use. This should be established next and is a short term goal.

Our intermediate goals are (i) the examination regarding an integrative quality model (including propagation) for classical sensor based information, results of environmental simulations and VGI; and (ii) experimentations on scale transitions.

ACKNOWLEDGEMENTS

This work was partially funded by the exploratory research project "Next Generation Digital Earth: Engaging the citizens in forest fire risk and impact assessment" from the Institute for Environment and Sustainability of the European Commission – Joint Research Centre and under the European Community’s Seventh Framework Programme (FP7/2007-2013), Grant Agreement No. 284898 (ENVIROFI project).

REFERENCES

Schade, S.: Sensors on the Way to Semantic Interoperability. GI-Days 2005: Geo-Sensor Networks

- From Science to Practical Applications, ifgi-Prints Bd. 23. *Münster (2005)*.

Sheth, A. et al.: Semantic Sensor Web. *IEEE Internet Computing (2008)*.

Janowicz, K. et al.: A RESTful Proxy and Data Model for Linked Sensor Data. *International Journal of Digital Earth (2011, in print)*.

Bröring, A. et al.: New Generation Sensor Web Enablement. *Sensors 2011, 11(3) (2011)*.

Maué P. et al.: Geospatial Standards for Web-enabled Environmental Models. *Internal Journal for Spatial Data Infrastructures Research (IJSDIR),* 6 (2011).

Janowicz, K. and Compton, M.: The Stimulus-Sensor-Observation Ontology Design Pattern and its Integration into the Semantic Sensor Network Ontology. *3rd International workshop on Semantic Sensor Networks 2010 (SSN10) (2010)*.

Goodchild, M. F.: Citizens as voluntary sensors: spatial data infrastructure in the world of Web 2.0. *International Journal of Spatial Data Infrastructures Research, 2 (2007)*.

DeLongueville, B. et al.:Digital earth's nervous system for crisis events: real-time sensor web enablement of volunteered geographic information. *International Journal of Digital Earth ,3(3) (2010)*.

Ehrich, H.-D. and Mahr, B.: Fundamentals of Algebraic Specification 1: Equation and Initial Semantics. *Springer (1985)*.

Spinsanti, L. and Ostermann, F. O.: Retrieve Volunteered Geographic Information for Forest Fires. *Proceedings of the 2nd Italian Information Retrieval (IIR) Workshop (2011)*.

Guarino, N.: Formal Ontology in Information Systems. International Conference on Formal Ontology in Information Systems, FOIS'98, Trento, Italy (1998).

Sommerville, I.: *Software Engineering. Pearson Education Limited (2007)*.

Agrawal, R. et al.: The Claremont report on database research. *SIGMOD Record (ACM) 37(3) (2008)*.

Kuhn, W. and Frank, A. U.: A Formalization of Metaphors and Image-Schemas in User Interfaces. In Cognitive and Linguistic Aspects of Geographic Space, NATO ASI Series, Dordrecht, The Netherlands, *Kluwer Academic Publishers (1991)*.

Car, A. and Frank, A. U.: Formalization of Conceptual Models for GIS using Gofer. *Computers, Environment, and Urban Systems*, 19(2) (1995).

Goodchild, M. F.: Measurement-Based GIS. *International Symposium on Spatial Data Quality, Hong Kong, Department of Land Surveying and Geo-Informatics, Hong Kong Polytechnic University (1999)*.

Schade, S. et al.: Comparing Approaches for Semantic Service Description and Matchmaking. *3rd International Conference on Ontologies, DataBases, and Applications of Semantics for Large Scale Information Systems (ODBASE) (2004)*.

Frank , A.: Towards a Mathematical Theory for Snapshot and Temporal Formal Ontologies. *AGILE 2007,* Aalborg, Denmark (2007).

Kuhn, W.: An Image-Schematic Account of Spatial Categories. Spatial Information Theory, 8th International Conference, *COSIT 2007*. Melbourne, Australia (2007).

Kuhn, W.: A Functional Ontology of Observation and Measurement. GeoSpatial Semantics ·*Third International Conference (GeoS)* (2009).

Schade, S.: Ontology-Driven Translation of Geospatial Data. AKA, Heidelberg, Germany (2010).

Peyton Jones, S.: Haskell 98 Language and Libraries - The Revised Report. *Cambridge University Press, Cambridge (2003)*.

Coates, T. et al.: URIs, URLs, and URNs: Clarifications and Recommendations 1.0 (2001).

Spinsanti, L. and Ostermann, F. O.: Validation and Relevance Assessment of Volunteered Geographic Information in the Case of Forest Fires. *2nd International Workshop On Validation Of Geo-Information Products For Crisis Management, Ispra, Italy (2011)*.

Masolo, C. et al.: D18 - Ontology Library (final version). *Deliverable of the WonderWeb Project (2003)*.

Navratil, G. et al.: Lifting Imprecise Values. *11th AGILE International Conference on Geographic Information Science (2008)*.

Ostermann, F. O., Spinsanti, L.: A Conceptual Workflow For Automatically Assessing The Quality Of Volunteered Geographic Information For Crisis Management. *Proceedings of AGILE Conference (2011)*.