Representing Uncertainties in the Sensor Web

Christoph Stasch [1,*], Richard Jones [2], Dan Cornford[2], Martin Kiesow[1], Matthew Williams[2], and Edzer Pebesma[1]

[1] Institute for Geoinformatics, University of Muenster, Weselerstrasse 253, 48151 Muenster, Germany; E-Mails: staschc@uni-muenster.de (C.S); martin.kiesow@uni-muenster.de (M.K.); edzer.pebesma@uni-muenster.de (E.P.)

[2] Computer Science, Aston University, Birmingham, UK; E-Mails: jonesrm1@aston.ac.uk (R.J.); D.Cornford@aston.ac.uk (D.C.), williamw@aston.ac.uk (M.W.)

***** Author to whom correspondence should be addressed; E-Mail: staschc@uni-muenster.de

Abstract: The Sensor Web provides a Web-based framework for the exchange of sensor data. As sensors indirectly measure values of a process of interest in reality, sensor outputs should be treated as uncertain and are the initial source of uncertainties in environmental modeling workflows. To date, uncertainties in observational data are either ignored or attached to the data in proprietary formats in the Sensor Web. In this paper, we present an approach for integrating uncertainties in the Sensor Web. First, we present an exchange format for uncertain observations. Secondly, we define an uncertainty-enabled Sensor Observation Service.

Keywords: Sensor Web, Uncertainty, Model Web, Sensor Observation Service

1. Introduction

The idea of the Model Web envisions an infrastructure for accessing and coupling (environmental) models in the Web [1]. The European research project UncertWeb[1] aims to uncertainty-enable the Model Web. To propagate uncertainties in model chains that consume observations the uncertainties need to be communicated in a standardized way. In this paper, we present an approach for uncertainty-enabling observation data in the Sensor Web. First, we provide some background on the Sensor Web and on uncertainty representations. We then present how to integrate uncertainties in the Observation & Measurements (O&M) format [2] and how to uncertainty-enable the Sensor Observation Service [3] followed by a description of our prototypical implementation. Finally, we discuss first results and the next steps needed to uncertainty-enable the Sensor Web.

[1] http://www.uncertweb.org

2. Background

The Sensor Web aims to make sensors discoverable, accessible and taskable in the internet [4]. The main model and encoding for sensor observations in the Sensor Web is defined by the O&M specification [2]. The Sensor Observation Service [3] provides a standardized service interface to manage observations in the Sensor Web. The Model Web goes one step beyond the Sensor Web and envisions an infrastructure for coupling (environmental) models in the Web [1]. As sensor observations are essential inputs to environmental models, the Sensor Web can be seen as part of the Model Web or, at least, as an essential component to realize the Model Web [5]. Assessing the observational and other uncertainties and propagating them through model workflows is crucial to support sound decision making in the Model Web.

While uncertainty representations for sensors in databases have been partly solved, the communication of uncertainties in the Sensor Web still remains a challenge [6]. The Uncertainty Markup Language (UncertML) was developed in the INTAMAP project[2] [7] and developed to a new version in the UncertWeb project[3]. UncertML allows the communication of probabilistic uncertainty. A first approach to combining UncertML with O&M was presented at the EGU meeting in 2011 [8].

In this paper, we describe how we extend the previous work by showing how to integrate UncertML with O&M and how to query observations with uncertainty information from a SOS. Furthermore, we present a prototypical implementation of the approach.

3. Uncertainty Representation in the Sensor Web

In this section, we describe the two building blocks for representing uncertainties in the Sensor Web. First, the uncertainty-enabled O&M format (U-O&M) is presented. Afterwards, we introduce our concept for an uncertainty-enabled SOS.

2.1. Uncertainty-enabled Observation & Measurements (U-O&M)

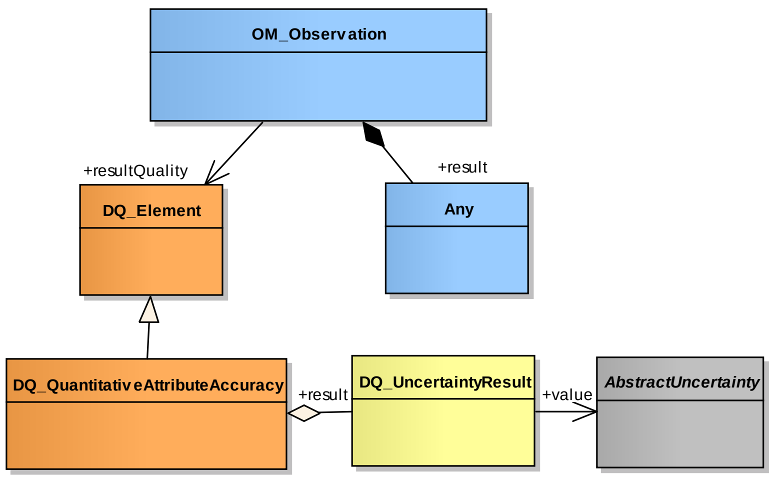
The U-O&M is a profile of the O&M specification [2] that allows the user to add uncertainties to observations. In addition, the profile defines restrictions on spatial and temporal geometries and result types to enable the development of software to support our U-O&M specification. Thus, it also defines a lightweight O&M profile described in detail in [9].

There are two ways to provide uncertainty information in sensor observations[4]: first, the uncertainty information can be considered as additional metadata to the measured valued. Secondly, the measured value itself can be considered as uncertain and can be represented as, for example, a probability distribution. In Figure 1 the extension introduced in the first approach is shown. Blue boxes show

[2] http://www.intamap.org
[3] http://www.uncertml.org
[4] In general, three sources of uncertainty can be distinguished for sensor observations: spatial location uncertainty, temporal uncertainty and uncertainty of the measurement value. In this approach, we focus on the uncertainty of the measurement value.

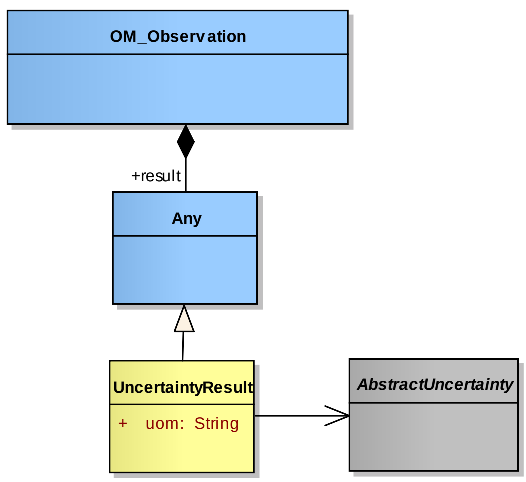
classes from the O&M model, orange boxes classes from the ISO standard Geographic – Metadata [10], the yellow box shows the extension defined in the U-O&M model, and the grey box the uncertainty class from UncertML. For additional metadata about the quality of an observation result, the `resultQuality` element is defined in the O&M model. As the `resultQuality` element is of type `DQ_Element` from [10], we introduce the `DQ_UncertaintyResult` that allows the integration of uncertainties defined in UncertML in the `resultQuality` element. In this approach, the result value of an observation remains a scalar value, with implicit uncertainty in the result quality metadata.

Figure 1. U-O&M model for observations that contain uncertainty information as additional metadata.



Instead of providing the uncertainty information as additional metadata, the observation result might more naturally itself be assumed uncertain. This approach is shown in Figure 2. The `UncertaintyResult` type is introduced that allows encoding UncertML uncertainties as result values of observations. In addition to the uncertainty information, the `uom` attribute allows the provision of information about the unit of measurement.

Figure 2. U-O&M model for observations that contain uncertain values as results.



To identify the U-O&M format in the SensorWeb, the MimeType identifier `application/x-om-u` has been defined. A JSON, as well as an XML, encoding has been defined. The encoding is appended to the basic MimeType identifier (e.g. `application/x-om-u+xml`).

3.2. Uncertainty-enabled Sensor Observation Service (U-SOS)

To store and retrieve observations in the Sensor Web, we define the uncertainty-enabled SOS (U-SOS) as a profile of the SOS. The profile utilizes the U-O&M format as default response format. The response format is indicated in the capabilities of the service by the MimeType identifiers as introduced in the previous section. In addition, a concept for filtering on uncertainties is defined enabling queries such as "Select all observations where the standard deviation of the measurement error is lower than 2". We restrict the `extension` element of a `GetObservation` request to be an `UncertaintyFilter`. The `UncertaintyFilter` contains a `ComparisonFilter` as defined in the Filter Encoding specification [11]. Each comparison filter has to contain a `valueReference` that uses the URLs for uncertainty concepts defined in the UncertML dictionary[5]. Listing 1 shows an excerpt of a GetObservation request that contains a filter that filters observations with normal distributed values and a variance lower than 0.3. The variance is identified in the `valueReference` element.

Listing 1. Uncertainty filter in an GetObservation operation request.

```
<?xml version="1.0" encoding="UTF-8"?>
<sos:GetObservation service="SOS" version="2.0.0" xsi:schemaLocation="http://www.opengis.net/sos/2.0 ../sosUncertaintyFilter.xsd" xmlns:sos="
http://www.opengis.net/sos/2.0" xmlns:wsa="http://www.w3.org/2005/08/addressing" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:swe="http://www.opengis.net/swe/2.0" xmlns:swes="http://www.opengis.net/swes/2.0" xmlns:fes="http://www.opengis.net/fes/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:ogc="http://www.opengis.net/ogc" xmlns:om="http://www.opengis.net/om/1.0">
  <!--uncertainty filter; filters observations whose standard deviation is lower than 0.3-->
  <swes:extension>
    <sos:UncertaintyFilter>
      <fes:PropertyIsLessThan>
        <fes:ValueReference>http://www.uncertml.org/distributions/normal#variance</fes:ValueReference>
        <fes:Literal>0.3</fes:Literal>
      </fes:PropertyIsLessThan>
    </sos:UncertaintyFilter>
  </swes:extension>
  <!--identifier of an offering-->
  <sos:offering>http://www.my_namespace.org/water_gage_1_observations</sos:offering>
```

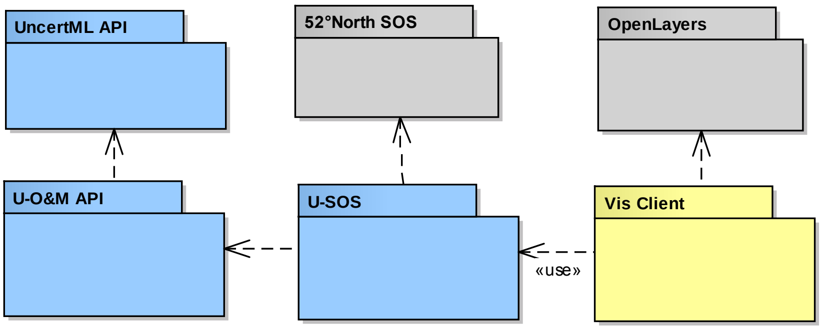
4. Implementation

There are four main components that have been developed in the implementation of the U-SOS. These are shown in Figure 3. The *UncertML API* provides a Java implementation of the UncertML conceptual model[6]. It contains parsers and encoders for XML and JSON encodings. The *U-O&M API* implements the U-O&M profile as defined in Section 3.1[7]. The *U-SOS* implementation is based on the

[5] http://www.uncertml.org/dictionary
[6] The API is available as Open Source under GPL 2.0 at https://svn.52north.org/svn/geostatistics/main/uncertweb/uncertml-api.
[7] The API is available as Open Source under GPL 2.0 at https://svn.52north.org/svn/geostatistics/main/uncertweb/om-api.

52°North SOS implementation[8]. In order to visualize the uncertain observations, a client has been developed that is written in JavaScript and uses the OpenLayers API[9].

Figure 3. Components of the U-SOS implementation. The blue boxes are Java components developed for the U-SOS, the yellow box indicates a JavaScript component and the grey boxes are APIs that are used in the implementation.



For the uncertainty extension, the database model of the 52°North SOS has been extended adding an `uncertainty` relation that is linked to the `observation` table that stores the observations. Furthermore, encoders and decoders have been implemented using the U-O&M API. The current implementation allows the insertion of uncertain observations using the transactional interface of the SOS and the retrieval of uncertain observations using the GetObservation operation. The uncertainty filters are not yet supported.

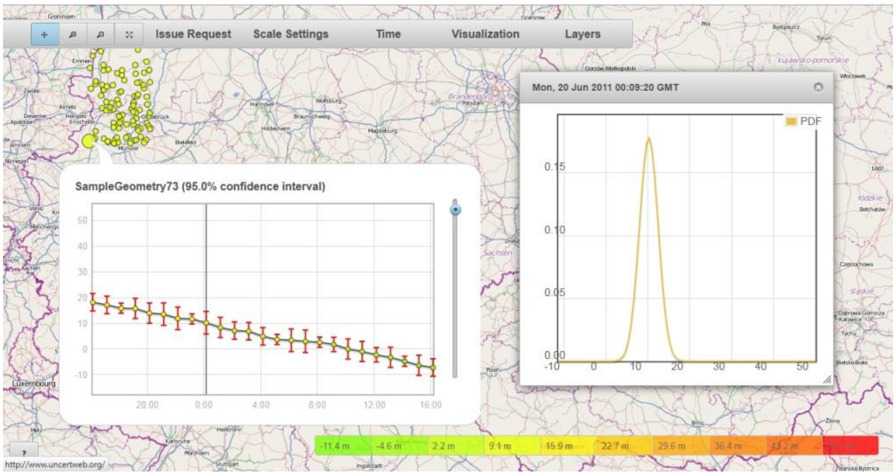
The visualization client allows visualization of the uncertain observations in different ways. Figure 4 shows the visualization of observations that contain normal distributions as result values. The observations are visualized showing a time series of the mean values and the 95% confidence intervals. The confidence level can be defined by the user. Furthermore, clicking on the time series produces a new chart in a pop-up window that shows the probability distribution of a particular observation. The client is also able to convert the normal distribution to other representations, e.g. exceedance probabilities.

---

[8] The U-SOS implementation is available as Open Source under GPL 2.0 at https://svn.52north.org/svn/geostatistics/main/uncertweb/u-sos

[9] http://www.openlayers.org

Figure 4. Open Layers based client for the visualization of uncertain observations.



5. Conclusions/Outlook

This paper provides the definition of an uncertainty-enabled Sensor Observation (U-SOS). Profiles of the O&M and Sensor Observation Service that allow for uncertain information are defined. The feasibility of the approach is shown in a prototypical implementation that comprises Java APIs for the encodings as well as the service interface. Furthermore, a client has been developed that allows the visualization of the uncertain observations in different ways.

The current approach allows the integration of quantitative uncertainties in sensor observations based upon probability theory. While we have demonstrated how to integrate UncertML in the O&M format, further work is needed to explore how to express uncertainties in the Sensor Model Language (SensorML) [12], which would open up the opportunity to describe model uncertainties.

We consider the current work presented here as a step towards uncertainty-enabling the Sensor Web and, in a next step, to also uncertainty-enable applications and models consuming the sensor observations. While our approach allows for exchanging uncertain observations in a standardized format, the issue remains how to motivate sensor manufacturers and sensor data providers to provide and communicate the uncertainty in the sensor observations. The calculation of per observation uncertainty, and in particular the representativeness errors, remains a challenging statistical and operational problem. Thus, further case studies need to be undertaken to demonstrate the benefit of calculating and communicating the uncertainty, and tools need to be developed to support the estimation or inference of observational uncertainty. Our vision is of a future Sensor Web that conveys information, not just observations, and where the information can be used to address a wide range of potentially unforeseen applications within a formal decision making framework. Without a quantitative measure of uncertainty associated with each observation we believe this vision cannot be fully realized.

Acknowledgments

The presented work has been funded by the European research project UncertWeb (FP7-248488, see http://www.uncertweb.org).

References and Notes

1. Geller, G.; Melton, F. Looking Forward: Applying an Ecological Model Web to assess the impacts of climate change. *Biodiversity* 2008, 9 (3&4), 79-83.
2. ISO 2011:19156 Geographic information - Observations and Measurements. International Standardization Organisation (ISO), 2011.
3. Bröring, A.; Stasch, C.; Echterhoff, J. OGC Interface Standard 12-006: SOS 2.0 Interface Standard. Candidate Standard. Open Geospatial Consortium, Wayland, MA, USA, 2012.
4. Botts, M.; Percivall, G.; Reed, C.; Davidson, J. OGC Sensor Web Enablement: Overview And High Level Architecture. Open Geospatial Consortium, Wayland, MA, USA, 2007.
5. Maué P.; Stasch, C.; Athanasopoulos, G.; Gerharz, L. Geospatial Standards for Web-enabled Environmental Models. *International Journal of Spatial Data Infrastructures Research (IJSDIR)*. 2011, 6. Available online: http://ijsdir.jrc.ec.europa.eu/index.php/ijsdir/article/view/198.
6. Balazinska, M.; Deshpande, A.; Franklin, M.; Gibbons, P.; Gray, J.; Nath, S.; Hansen, M.; Liebhold, M.; Szalay, A.; Tao, V. Data Management in the Worldwide Sensor Web. *Pervasive Computing*, IEEE, 2007, 6, 30 -40.
7. Williams, M.; Conford, D.; Bastin, L.; Pebesma, E. Uncertainty Markup Language (UncertML). OGC Discussion Paper 08-122r2. Open Geospatial Consortium, Wayland, MA, USA, 2009.
8. Cornford, D.; Williams, M.; Bastin, L. Uncertainty enabled Sensor Observation Services. European Geosciences Union General Assembly 2010, Vienna, Austria, 02-07 May 2010.
9. Jirka, S.; Stasch, C.; Bröring, A. Lightweight SOS Profile for Stationary In-Situ Sensors. OGC Discussion Paper 11-169. Open Geospatial Consortium, Wayland, MA, USA, 2011.
10. ISO 2007:19115 Geographic Information – Metadata. International Standardization Organisation (ISO), 2007.
11. ISO 2010:19143 Geographic information – Filter encoding. International Standardization Organisation (ISO), 2010.
12. Botts, M.; Robin, A. Sensor Model Language (SensorML). OpenGIS Implementation Specification 07-000. Open Geospatial Consortium, Wayland, MA, USA, 2007.