

Providing Mobile Sensor Data in a Standardized Way - The SOSmobile Web Service Interface

Christoph Stasch, Arne Bröring, Alexander C. Walkowski

IfGI, Institute for Geoinformatics
Robert-Koch-Str. 26-28, Muenster 48149, Germany
{staschc, arneb, walkowski}@uni-muenster.de

ABSTRACT

The Sensor Observation Service (SOS) of the Open Geospatial Consortium (OGC) defines a web service to distribute sensor data. During the utilization of the SOS in different mobile sensor use cases of the OSIRIS project deficits of the specification were identified. For example, the SOS specification lacks the ability to update information about deployed mobile sensors (e.g. position or status) dynamically. This work presents the concept of an interoperable web service interface (SOSmobile) allowing the exchange of mobile sensor data. The requirement for such a service arises due to the fast growing number of applications which incorporate mobile sensors. To enable the integration of data collected by mobile sensors into multiple applications the use of an appropriate interoperable encoding as well as a well-defined interface to access the data is essential. After introducing the SOS and the related data models the deficits are being described and the new SOSmobile interface based upon the OGC SOS specification is defined.

INTRODUCTION

As mobile sensor platforms are becoming smaller and lower priced while the processing and broadcasting capabilities of these platforms are advancing, the deployment of mobile sensors increases in many different applications (Hellerstein, 2003). These applications range from environmental monitoring over early warning systems up to military applications (Sheperd, 2004). One approach to standardize the discovery, exchange and processing of sensor data, as well as the tasking of sensor systems, is the Open Geospatial Consortium's (OGC) Sensor Web Enablement (SWE) initiative. The SWE initiative defines several models and encodings for describing sensors and sensor observations in combination with several service interfaces which use these models and encodings (Botts, 2006).

We propose a concept that enables the Sensor Observation Service (SOS) (Botts, 2007) for the provision of mobile sensor data. Currently, the integration of mobile sensor data is laborious and relevant information concerning the special aspects of mobile data can not be provided by a SOS. The paper is organized as follows: At first the relevant SWE encodings for sensor data and how this data can be accessed through the SOS are introduced. Afterwards issues of the relevant models and encodings are presented and our modifications to these are described which result in a model for mobile sensor data. Additionally the service interface of the SOS is adjusted as described in the subsequent section. Finally conclusions and questions for future work are presented.

SWE ENCODING AND ACCESS OF SENSOR DATA

The SOS specification leverages the Observations and Measurements (O&M) (Cox, 2006) specification to encode observations and the Sensor Model Language (SensorML) (Botts, 2007)

specification to return sensor descriptions¹. These specifications are explained in the following three subsections.

Observations and Measurements

The O&M specification specifies basic models and encodings for observations and measurements made by sensors (Cox, 2006). An observation could be defined as an act of observing a phenomenon. A measurement is a specialized observation, in which the result is a numerical value.

The basic observation model contains five components (as shown in Figure 1): The *procedure* element should point to the procedure (usually a sensor), which produced the value for the observation. The *observedProperty* element references the phenomenon that was observed. The *featureOfInterest* refers to the real world object to which the observation belongs. The *samplingTime* attribute indicates the time, when the observation was made. The observation value is contained in the *result* element.

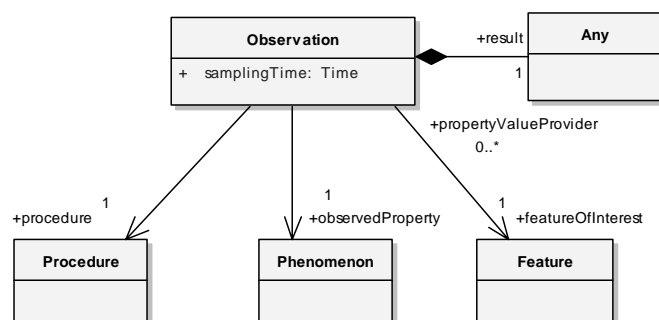


Figure 1: Basic observation model of O&M specification

The observation acts as a *property value provider* for a feature: It provides a value (e.g. 27° Celsius) for a property (e.g. temperature) of the featureOfInterest (e.g. weather station) at a certain timestamp. The location to which the observation belongs is indirectly referenced by the geometry of the featureOfInterest.

Sensor Model Language

The SensorML specification (Botts, 2007) provides models and encodings to describe any kind of process in sensor or post processing systems. Thus, the basic type of all SensorML descriptions is the process type, which contains input and output elements and several additional parameters. Different subtypes of the process type are specified which can be used to depict diverse kinds of detectors, actuators or systems of processes.

¹ Also the Transducer Markup Language (TML) (Havens, 2006) can be used by the SOS to encode sensor descriptions. Due to the reason that currently nearly all SOS implementations use SensorML for the sensor metadata documents, we concentrate on SensorML.

Sensor Observation Service

The SOS provides a standardized web service interface which allows a client to access descriptions of associated sensors and their collected observations. The operations of this interface are divided into four profiles shown in Table 1.

Table 1: Operations of the SOS interface

Entire		
Core	transactional	enhanced
GetCapabilities DescribeSensor GetObservation	RegisterSensor InsertObservation	GetObservationById GetResult GetFeatureOfInterest DescribeFeatureType DescribeObservationType DescribeResultModel

The *core* profile includes the three mandatory operations *GetCapabilities* for requesting a description of the service and the offered sensor data, *DescribeSensor* for retrieving the metadata documents of the sensors and *GetObservation* for querying observations of certain sensors or phenomena using any combination of temporal, spatial and value filters.

The *RegisterSensor* operation of the optional *transactional* profile enables sensor producers to register new sensors. Afterwards the *InsertObservation* operation allows the integration of new observations produced by registered sensors.

The *enhanced* profile offers additional optional operations like the *GetObservationById* operation to request observations by their ID or the *GetResult* operation to retrieve only the results of observations without their metadata. A service implements the *entire* profile if it supports all operations.

A DATA MODEL FOR MOBILE SENSOR DATA

To be able to describe mobile sensor data the current data model defined by the OGC needs to be adjusted. Thereby, the essential extension of the model is to endow an observation with information about the location where it took place as well as with the geometric description of superior features to which it belongs (e.g. the area within a mobile sensor is moving).

Following Probst and Espeter (Probst, 2006), the spatial dimension could be used as a criterion to the classification of phenomena. The phenomena itself are properties of observed features. Thus, the phenomenon's dimension is regarded as the dimension of the feature extent. The following problem might occur during the observation of continuous phenomena belonging to two- or three dimensional features: In general the phenomenon will not be recorded entirely. Instead, discrete measurements will be conducted at several points in space. Figure 3 shows an example scenario for such a situation.

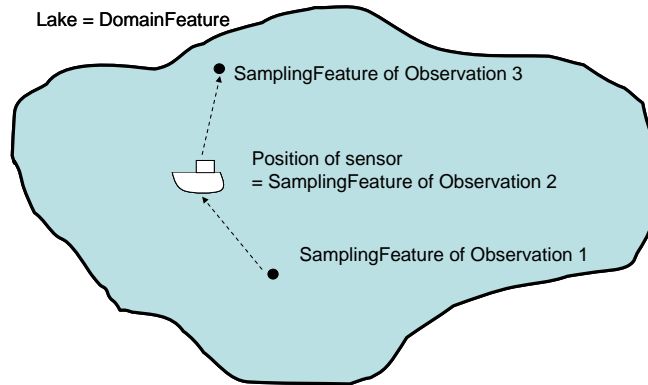


Figure 3: SamplingFeature and DomainFeature

A boat moves across a lake and takes depth measurements at certain locations. The measured value is only valid for the place where the measurement was made. The values usually vary from location to location. However, the attribute “depth” applies to the lake itself which is a three dimensional body and not a point.

Consequently, the depiction of an observation has to contain the location of the measurement as well as the link to the superior feature upon which the observation was made. In the following the location where the measurement took place is called *SamplingFeature* and the superior feature will be named *DomainFeature*. The geometry of the DomainFeature and the SamplingFeature differ in the most cases of mobile sensors. The DomainFeature could be also used for mobile sensors to represent the area in which a mobile sensor is moving.

Without having the information about the link between the depth value and the lake itself the measurement would not be useful for the end-user. Hence the O&M model has to contain information about the exact SamplingFeature as well as the DomainFeature. The proposed integration of these two entities into the observation model shall be called *DomainFeature-Concept*.

Our approach to integrate the notion of this DomainFeature-Concept into the existing O&M model is shown in Figure 4. In this extended model an observation references exactly one SamplingFeature and at least one DomainFeature which represents a superior feature. The SamplingFeature has to be spatially contained in a DomainFeature. In the example of the lake the DomainFeature represents the lake and the SamplingFeature is equivalent to the point on the lake where the measurement took place.

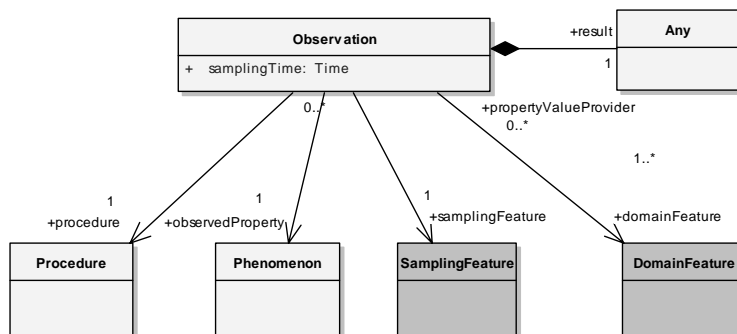


Figure 4: DomainFeature-Concept

Service Metadata Model for the SOSmobile

Structure and content of the service metadata document returned by the GetCapabilities operation have to be adjusted to fulfill the special needs for the distribution of mobile sensor data. A metadata description of the SOSmobile has to list the DomainFeatures observed by the associated sensors. Additionally, this service metadata description has to inform about the relationship between the DomainFeature and its linked sensors.

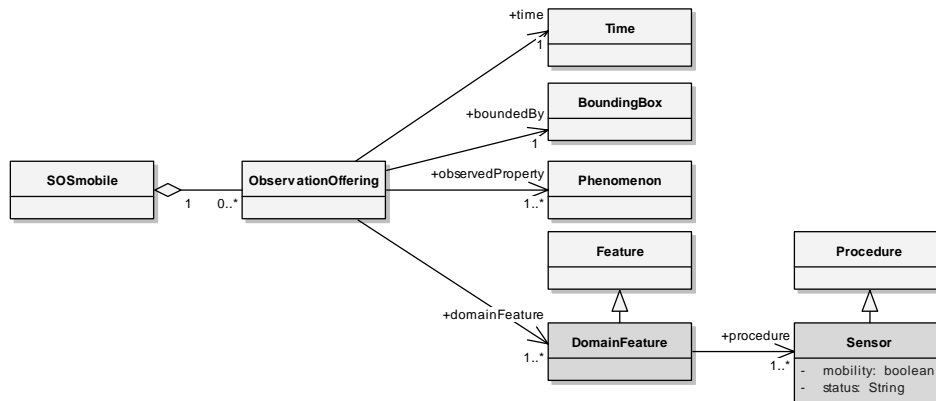


Figure 5: Service metadata model

A part of the adjusted model is shown in Figure 5. Spatially or semantically related observations of a SOSmobile are grouped by ObservationOfferings. The ObservationOffering concept can be compared to the layer concept which is used in common GI technologies. Each ObservationOffering hosts the time for which observations are available, the observedProperties of the observations as well as the boundedBy element representing the spatial extent of all contained observations.

Up to now the service description returned by the GetCapabilities operation contains the URNs² of all featuresOfInterest observed by associated sensors. Due to the movement of a mobile sensor it is possible that the featureOfInterest changes for each observation. If the count of the featuresOfInterest increases to a high number the service description document inflates to a large size and the usage of this document by a client would become increasingly inefficient³. Our approach to solve this problem is to make use of the proposed DomainFeature-Concept and to list the DomainFeatures instead of all SamplingFeatures within the service description. This results in an association of the ObservationOffering with an arbitrary number of DomainFeatures as shown in Figure 5.

Another imperfection of the current SOS specification is the missing declaration of the relationship between features and their observing sensors. Up to now the service description lists all features and sensors separately without explaining which sensor observes which feature. But this

² An Uniform Resource Name unambiguously identifies a web resource. The structure of URNs used in OGC specifications is defined by Reed (2004).

³ The service description of a SOS, which offers fires detected from satellite images in the Advanced Fire Information System (AFIS) (McFerren, 2006) has increased to 250 Megabytes when 120.000 fires have been contained in the SOS. The big size of the service description occurs, because a new featureOfInterest has to be created for every new fire. These featureOfInterests have to be listed in the service description of the SOS.

information is especially of importance, if a SOS offers data gathered by mobile sensors. It is needed to reflect within which area (or DomainFeature) a mobile sensor is moving. Thus, our approach establishes an association between a DomainFeature and its observing sensors (see Figure 5).

Further on, a sensor data consumer demands information about the *status* and the *mobility* of a sensor. Mobile as well as stationary sensors may undergo different states during their lifespan (e.g.: *active, inactive, stand-by, moving, disabled, defect* etc.). It is important for the end-user to be able to retrieve the current status of a sensor and to know whether the sensor currently collects data or not. The sensor status is especially of interest to mark a sensor as “inactive” to indicate that it will not be utilized in future. In this case a complete deletion of the sensor is not meaningful because the information and description of the sensor are still of concern for prior captured observations. One option could be to incorporate the mobility and status information into the SensorML description of the sensor, which can be retrieved by the DescribeSensor operation. But these SensorML descriptions are often shared between multiple SOS instances by the means of common repositories. Due to the fact that the status of a sensor may differ from one SOSmobile instance to another we propose to integrate *status* and *mobility* as attributes of a sensor into the service description as presented in Figure 5. The type of a sensor’s mobility (*mobile* or *stationary*) affects the subsequent behaviour of the client. The same applies for the sensor’s status.

INTEROPERABLE WEB SERVICE INTERFACE FOR THE DISTRIBUTION OF MOBILE SENSOR DATA

The proposed approach of the SOSmobile interface extends and adjusts the existing operations of the SOS to the special needs of mobile sensor data. To ensure interoperability and to enable later the integration of the SOSmobile into existing specifications, the specification of the SOSmobile contains changed operations of the common SOS interface. Table 2 shows the operations of the new SOSmobile.

core profile	transactional profile
GetCapabilities DescribeSensor GetObservation GetFeatureOfInterestTime	RegisterSensor InsertObservation UpdateSensor

Table 2: Operations of SOSmobile

We added the GetFeatureOfInterestTime operation to the Core profile and defined a new operation UpdateSensor in the Transactional profile, which should enable the user to update dynamic parameters of mobile sensors. All operations taken from the SOS specification are modified except the InsertObservation operation. These modifications are described in the following subsections.

Adjustment of DescribeSensor operation

It is necessary that the metadata descriptions of sensors are time-dependently accessible, which is currently not specified by the SOS specification. While the position of a stationary sensor is constant over time the positions of mobile sensors are dynamic and vary over time. In certain cases the end-user is not only interested in the current position of the sensor but also in positions of earlier points in time when an observation was made. The latest SensorML specification allows the integration of multiple positions for one sensor description. But the number of different positions traversed within a sensors lifetime might be huge so that the storage of all positions in one sensor description would become inefficient because it results in very large metadata documents. The DescribeSensor operation of the SOS interface allows the request of sensor descriptions. This operation will be extended by a time parameter to be able to receive time-dependent sensor metadata documents.

Adjustment of GetObservation operation

Another essential requirement is to enable additional filter options for the observation request. Due to the integration of the DomainFeature-Concept the *SOSmobile* needs to support additional filter capabilities to derive benefit from the new information. Thus, it is necessary to be able to specify filters for the SamplingFeature as well as the DomainFeature within the request for observations. Such a mechanism allows for instance the retrieval of all observations which were made upon a specified DomainFeature. But it also allows the client to receive all observations captured at locations which lay within a specified geometry contained in the domain feature's geometry. In the example of the depth measurement, the DomainFeature filter parameter enables users to request all depth measurements for specific lakes of a certain area (e.g. query of depth measurements for all lakes in Germany). On the other hand, if the client is interested in a certain area within a specific lake, he can use the SamplingFeature filter to define a geometry of interest within the lake.

Adjustment of RegisterSensor operation

The SOS offers an operation to register new sensors by the RegisterSensor operation. The request of this operation currently contains two parameters: The SensorML description of the sensor and an observation template indicating the structure of the observations produced by the sensor. The SOS specification does not define to which ObservationOffering the produced observations shall belong and which features are observed by the registered sensor. To solve these issues we extend the operation with the appropriate parameters. The response of the operation contains the ID of the registered sensor. Afterwards one can invoke the InsertObservation operation using this ID to integrate new observations for the registered sensor.

Adjustment of GetFeatureOfInterestTime operation

Further on, it is required to supply a client with the time period within which data is available. At the moment, the SOS specification defines the GetFeatureOfInterestTime operation to allow the request of this time period. But its parameterization is limited. Only a feature ID can be specified. The response of the GetFeatureOfInterestTime operation contains the time periods when the feature was observed. It is not possible to request the time periods in dependency of sensors or certain phenomena. Hence, the *SOSmobile* interface extends the parameterization of the GetFeatureOfInterestTime operation to allow the request of the time when values are available for a specified DomainFeature, phenomenon, sensor or spatial extent.

Definition of UpdateSensor operation

Finally, an indispensable requirement is to endow the client with a mechanism to update and change the attributes of a sensor registered to a SOS. Due to the dynamics of mobile sensors it is required for the *SOSmobile* interface to offer mechanisms to change the attribution of registered sensors. The movement of mobile sensors requires the ability to update the sensor position via the service interface.

Further on it has to be possible to change the state of a sensor. An example pointing this demand out is the power supply of mobile sensors which is still topic of current research. Kar (2006) proposes the usage of redundant sensors: While one sensor is deployed another one recharges. In this case the status information could indicate, whether a sensor collects data ("active") or recharges its battery ("charging").

Especially needed is the ability to set the status of a sensor to "inactive" because a complete removal of registered sensors or their descriptions is not possible. The description of a sensor usually includes essential metadata necessary to interpret associated observations. To sustain the

interpretability of historic observations the SOS can not delete the descriptions of deregistered sensors.

CONCLUSION AND OUTLOOK

The modifications of the O&M data model and the SOS interface, which result in the specification of the SOSmobile, enable the publication of observations acquired by mobile sensors via an interoperable web service. The presented concept of the SOSmobile and its definition is currently implemented in the OSIRIS project (OSIRIS, 2007) as a proof-of-concept under the umbrella of the 52°North (<http://www.52north.org>; Kraak et. al 2005) open source initiative. The developed specification has to show that it can stand the test in the real-world scenarios of the OSIRIS project and in future Sensor Web projects. If the applicability of the mobile-enabled interface and its associated data encoding has been proven the SOSmobile will be contributed to the OGC as a change request or profile definition of the SOS specification.

ACKNOWLEDGEMENT

Parts of the work published in this paper were funded by the project OSIRIS (abbr. *Open architecture for Smart and Interoperable networks in Risk management based on In-situ Sensors*). OSIRIS is an Integrated Project co-funded by the INFOS DG of the European Commission within the RTD activities of the Thematic Priority Information Society Technologies (OSIRIS, 2007).

BIBLIOGRAPHY

- Botts, M., A. Robin, J. Davidson & I. Simonis: OpenGIS Sensor Web Enablement Architecture Document. OpenGIS Discussion Paper. Version 1.0 (OGC 06-021r1). (2006)
- Botts, M. & A. Robin: OpenGIS Sensor Model Language (SensorML) Implementation Specification. OpenGIS Implementation Specification. Version 1.0.0. (OGC 07-000). (2007)
- Cox, S.: Observations and Measurements. OpenGIS Discussion Paper. Version 0.13.0 (OGC 05-087r3). (2006)
- Havens, S.: OpenGIS Transducer Markup Language Implementation Specification. OpenGIS Implementation Specification. Version 1.0.0 (OGC 06-010r2). (2006)
- Hellerstein, J. M., Wei Hong & Samuel R. Madden: The sensor spectrum: technology, trends, and requirements. In: SIGMOD Rec., 32(4): 22-27. (2003)
- Kar, K., A. Krishnamurthy & N. Jaggi: Dynamic node activation in networks of rechargeable sensors. In: IEEE/ACM Trans. Netw., 14(1): 15-26. (2006)
- Kraak, M.-J., A. Sliwinski & A. Wytzisk: What happens at 52N? An Open source approach to education and research. Joint ICA commission seminar 6.-8. July 2005, part of the 22nd ICA conference, ICC 2005: Mapping approaches into a changing world, 2005, 16-20. (2005)
- McFerren, G., S. Roos, A. Terhorst: Fire Alerts On The Geospatial Semantic Web. In: Scharl, A. & Tochtermann, K. [Hg.]: The Geospatial SemanticWeb. Berlin. (2006)
- OSIRIS: Open architecture for Smart and Interoperable networks in Risk management based on In-situ Sensors home. Online at: <http://www.osiris-fp6.eu/> (2007)
- Probst, F. & M. Espeter: Spatial Dimensionality as Classification Criterion for Qualities. International

Conference on Formal Ontology in Information Systems. Baltimore. (2006)

Reed, C.: A URN namespace for the Open Geospatial Consortium (OGC).
IETF Internet-Draft. 1 (OGC 04-013r4). (2004)

Shepherd, D. & S. Kumar: Microsensor Applications. In: Iyengar, S. S., Brooks, R. R. (eds.):
Distributed Sensor Networks Boca Raton (2004)