Fourth IEEE International Conference on eScience

Sensor Metadata Management and its Application in Collaborative Environmental Research

Nicholas Dawes*, K. Ashwin Kumar#, Sebastian Michel#, Karl Aberer# and Michael Lehning*

*WSL, Swiss Federal Institute for Snow and Avalanche Research (SLF), Davos, Switzerland
dawes@slf.ch, lehning@slf.ch
#Ecole Polytechnique Fédérale de Lausanne, Switzerland
ashwinkumar.kayyoor@epfl.ch, sebastian.michel@epfl.ch, karl.aberer@epfl.ch

***Abstract*— This paper[1] considers metadata generation and tracking in a collaborative environment where users publish raw sensor data in the form of virtual sensors and post-process data by means of filtering, modeling, or query processing techniques. In the metadata system described, data from different sources with different provenance will be enriched with further metadata at each processing step to describe the processing implemented and/or observations which may explain anomalies in the data. The management of this data is the subject of this paper. In the context of sensor data processing, in particular in the environmental sciences, there is still a large gap between data acquisition and metadata gathering, further complicated by the problem of combining both. In this paper, an attempt is made to bridge the gap between data management and semantic annotation. This paper describes a user friendly, easily deployable system for gathering sensor metadata and capturing semantics behind higher level data processing steps. These semantics are particularly useful in understanding data processing workflows. Furthermore, different methods of querying, exporting and importing gathered data from and to higher level applications are examined.**

I. Introduction

In recent years, sensor networks have gained wide popularity in a variety of application scenarios, ranging from monitoring applications in production chains to more sophisticated sensor deployments in the environmental sciences.

In science, the requirement to develop a bespoke acquisition, databasing and querying infrastructure for each sensor or sensor network application adds a layer of expense and a requirement for skills which may not be present within the team. A generic infrastructure which addresses all of these issues whilst remaining open and flexible enough to allow the scientist to deploy any sensor and carry out any data processing required, allows costs to be reduced and allows more "science" to take place. Such a generic infrastructure to support environmental research exists within the Swiss Experiment (SwissEx)[2]. This paper considers the development of a metadata infrastructure to support the generic data infrastructure and track the provenance of the data.

SwissEx is a collaboration of environmental science and technology research projects. These projects cover a range of environmental hazards from sustainable land use, to earthquakes and avalanches. In the details of these projects, there is however a large range of overlap where they may benefit from sharing data, particularly if experiments can be arranged to take place on common sites. Measurements such as meteorological parameters, soil temperature/conductivity/humidity and hydrological parameters are common across many projects and some projects even have synergies on much larger scales.

In the past, many scientific projects have been very isolated, data has seldom been reused within departments, opportunities for data sharing within institutions has been missed and collaboration across institutions has generally only taken place when the expertise did not exist in-house. E-science is changing this and SwissEx is one such e-science project. The SwissEx collaboration will encourage data sharing and preservation of knowledge across projects and institutions through the use of a common, state-of-the-art databasing and data processing infrastructure. The sharing of data within such an e-science community is purposeless without good metadata management both at the source and throughout the processing chain. Without metadata provenance tracking and a method of providing the relevant metadata at the data interface, data may be misinterpreted by a 3rd party. It is for this reason that scientists have traditionally preferred to collect their own data, where the provenance is known.

This paper introduces the Sensor Metadata Repository (SMR) and tools to propagate this metadata to the data interface and aims to provide an infrastructure to allow sharing of data and metadata simultaneously in a common interface.

A. Contents of the paper

This paper mainly concentrates on metadata, its management and its applications in collaborative environmental research projects.

The paper is organized as follows: Section II presents the related work. Section III provides some of the fundamentals

[1]The main funding sources for this paper were the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322 and the Competence Center Environment and Sustainability of the ETH Domain (CCES).

[2]www.swiss-experiment.ch

978-0-7695-3535-7/08 $25.00 © 2008 IEEE
DOI 10.1109/eScience.2008.27

of data and metadata in environmental research. Section IV contains a discussion of the workflow and the implementation of the Sensor Metadata Repository. Section VI deals with the integration of the Sensor Metadata Repository with SensorMap (Microsoft's Visualization tool). Section VII gives an outlook on ongoing and future work. Section VIII presents the conclusion.

II. Related Work

Data provenance [1] [2] has been a hot research topic recently and is finding its application in variety of research problems [3] [4] [5] [6], especially in e-science [7] [8] [9] [10] [9]. Simmhan et al. [11] discuss the taxonomy of data provenance characteristics and apply it to current research efforts in e-science. Data annotation is yet another buzz word which is very often used in relation to data provenance. Usually, researchers not only produce and consume data, but they also comment on it and refer to it, as well as referring to the results of queries upon it. Annotation is therefore a significant aspect of scientific communication. One researcher may wish to highlight a point in data space for another to investigate further. They may wish to annotate the result of a query such that similar queries show the annotation.

J. Zhao et al. [12] carried out research into annotating, linking and browsing provenance logs for e-science using a conceptual open hypermedia system to build a dynamically generated hypertext of web of provenance documents. Their work does not, however deal with the acquisition of annotations and metdata storage which is given more weight in the SwissEx application. Fox [13] discusses different sources of metadata and approaches to metadata. However the term "Metadata" is loosely defined and has been used in variety of contexts.

Much research has been carried out in the area of data annotation, but less effort has been put into documentation of the metadata and its storage in the e-science context. This paper concentrates on these areas, discussing the issues, ideas, implementation and future scope of the same.

III. Data and Metadata – Basics

Metadata is "data about data" and can be thought of as descriptions of other data. Once data has been acquired, the metadata becomes equally important. In general, four types of metadata may be considered.

1. **Experiment centric static information** - background information on the experiment, the overall structure and the phenomena which the sensor network is there to measure, which must be stored once.
2. **Sensor centric static information** - background information such as the type and location of each sensor, which must be stored once.
3. **Dynamic sensor information** - information on the servicability of the sensor i.e. whether it is deployed, stored, broken etc.
4. **Dynamic data quality information** - a measure of the quality of the data which requires a continuous variable.

Without this metadata, many anomalies in the data are hard to interpret. If this metadata is available for comparison, it makes the data significantly easier to interpret, although the ultimate aim is to annotate the data with metadata to provide the analyst with timely information on the reliability of the data without having to search.

A real world scenario which throws light on the importance of metadata documentation is considered below:

A scientist, while carrying out data analysis realizes that the recently acquired data contained errors. He decides to locate the faulty sensors responsible for the corrupted data, so that he can get them repaired, but there are several sensors of the same type located at the same position. His predecessors deployed the sensors, but kept paper records of the locations of the sensors in their notebooks which were destroyed when they left, hence this information is not available. To locate the faulty sensor, he needs an archive of sensor serial numbers and their related database parameter names as well as other supporting metadata which may help him locate the correct sensors. It would be highly time consuming for him to investigate this without the metadata.

The scenario provided emphasizes that if records are not kept, kept in paper records, or kept in a file structure, known only to the scientist who deployed the sensor, the knowledge required to do simple tasks in future may be lost when the scientist leaves or forgets where he recorded it 5 years on. If a scientist periodically enters the details of sensor and its deployment into a semantic repository, the metadata can later be merged with the data to find out its origin.

This raises the requirement and provides the motivation for the research and development of the Sensor Metadata Repository (SMR).

IV. Metadata Management

Several e-science projects [14] [15] [16] make use of Web 2.0 portals in the form of wikis or custom made environments to foster collaboration among scientists. For instance, Kepler and our own platform, Swiss Experiment [3], are based on the media wiki platform and employ several extensions to ease its use. The main application behind myExperiment is implemented using ruby on rails but it also makes use of a wiki to provide additional information to the users about the particular experiments. Swiss Experiment aims however to provide support for all types of metadata, whereas the other projects concentrate only on workflows.

A. Standard Wiki for Experiment Centric Static Metadata

A wiki is a collaborative software platform which allows users to create, enter and modify pages using a simplified markup language in a browser based editor. Wikipedia is a very popular example of a wiki. The important characteristics of wiki technology which make it ideal for use in collaborative projects are as follows:

- Ease with which pages can be created and updated

[3] www.swiss-experiment.ch

- Hooks for extensions provided in the wiki engine make it a flexible software as any functionality can be added simply by writing extensions

The basic wiki is ideal for scientific notation of the experiment centric static information, i.e. free text information which is not suited to a database structure and which would not be useful in annotating the data. It allows scientists easy collaborative access to a central store of background information.

B. Semantic Wiki for Sensor Centric Static and Dynamic Metadata - the Sensor Metadata Repository

In addition to the basic wiki functionality, the Semantic Wiki [17] offers an easy to use technique of annotating wiki pages with semantics in the form of (attribute, value)-pairs. For example, semantic MediaWiki allows annotations in the form of [[<predicate>:=<object>]]. Semantic annotation provides information about which entities (or more generally, semantic features) appear in a text and where they appear. More formally, semantic annotations represent a specific sort of metadata, which provides references to entities in the form of URIs or other types of unique identifiers. Semantic annotations can model any process by meaningfully annotating the entities, connecting them semantically to each other. Transparently to the users, these annotations can be automatically generated based on semantic templates or (more advanced) through the use of semantic forms. We will briefly discuss the usage of these tools to build up a sensor directory. The sensor directory serves as a fundamental concept for gathering both basic and advanced sensor metadata.

The Semantic MediaWiki stores information in a dynamic database structure. This kind of structure is ideal for storing dynamic and/or static data on the sensor, but not provided by the sensor, i.e. a user interface for a metadata database. The storage of this data in a database format means that it may be later queried for annotating the data.

Based on the user's semantic annotations of articles, the semantic MediaWiki generates machine-readable documents in OWL/RDF format. The Resource Description Framework or RDF [18] is an emerging interoperable standard for metadata on the web. The main advantage of RDF is that it converts resources on the web from a machine-readable to a machine-understandable format. This provides the interoperability between applications that exchange machine-understandable information on the web. It is therefore ideal to create a sensor metadata repository which can be exportable in RDF format, so that the collected metadata can be easily integrated with external data sources.

C. Dynamic Data Quality Metadata

Data quality information can take many forms, but ideally a measure of quality for every data point is required. This quality data is therefore best stored in the same database as the data. A measure of data quality inherently infers a prior knowledge of parameters, e.g. statistical parameters, associated with good data on a particular sensor. In SwissEx, the statistics/fundamental components of the data will be monitored as well as checking simple data ranges.

The component of the data monitored by the data quality monitoring algorithms is dependent on the sensor type and measurement scenario. Different measurements will have widely varying components which best describe the quality of the data. For instance, the overall correct functioning of a solar radiation sensor may be monitored by detecting the daily fluctuation of overall power. This will monitor that the sensor is functioning, but does not provide a measure of the amplitude of the fluctuation or the amount of noise on the measurement, hence this algorithm must be combined with further detection/threshold algorithms to check these components. As a minimum, each sensor should have a range checking quality measurement, checking that the recorded values are not outside the possible range of the sensor/virtual sensor. More advanced automatic control of measured data is possible [19] but most likely not suited for all possible applications.

D. Integration of the three systems

This approach to metadata recording provides two databases, a static/periodically sampled metadata database containing the sensor parameters and observations of the sensor, and a continuously sampled database containing the data and quality metadata (the standard wiki can act as a data resource, but does not store data structurally). A subject for later in this paper is the integration of these two databases, which allows the metadata to be propagated into the data visualisation/provenance system.. The data quality variable will then be available as a measurement parameter alongside the raw data, annotated with the static metadata. It is envisaged that the data quality could be overridden by the static metadata during periods between the sensor being observed as damaged/repaired, or removed/deployed, providing a single data quality variable. The data plots may also be annotated with the metadata to provide the analyst with timely information on the data displayed.

- **Example 1:** consider the case where a data quality algorithm highlights data outside a specific range, which may have been determined on the fly through statistical methods as anomalous data. When a sensor fails and provides a constant output, a simple data quality algorithm (checking only data ranges) would still calculate the data as within the limits. A manual record of when this sensor failed and when it was repaired and reinstalled, could be used to override the algorithm calculated quality values and highlight the data as unusable. Annotation of the data/plots would tell the analyst why this data was unusable.
- **Example 2:** consider the case where snowfall or strong wind periodically corrupt the snow height measurement, which is based on an ultra-sonic distance measurement. Here, the manual record of sensor failures cannot be used to annotate the data as the sensor is working perfectly. In this case, the data series exhibits typical spikes ([19],

which can be detected by an automatic data control. The algorithm calculated data quality value could then be used to highlight the anomalous values and even provide a suggestion for corrected data.

- **Example 3:** consider the case where changes in external factors such as surface cover corrupt the data with a constant (or varying) offset. An example is the calibration that is needed for IR surface temperature measurements. These measurements are increasingly used and have augmented significantly our knowledge on atmosphere - surface exchange. However, the IR sensor needs to know the emissivity of the surface, which is significantly different for snow and other surfaces such as soil or vegetation. If this effect has not been identified as a problem with the sensor, it is unlikely to be detected by the quality algorithm. Once the scientist has identified this phenomena in the data, he/she may use the Semantic MediaWiki to note that during this period, the values were subject to an offset and the reason for the offset, which could be used to annotate the data and/or adjust the data quality variable. An even better approach is to try to re-calculate surface temperature based on depth measurements (see Example 1), which help to detect presence of snow cover or growth of vegetation.

V. Sensor Metadata Repository - SMR

The Sensor Metadata Repository (SMR) was introduced in Section IV as the repository for static and dynamic sensor centric metadata. A Wiki based SMR tool was developed for Swiss Experiment. This tool provides scientists with an easy to use interface to enter the sensor metadata by creating a set of interlinked pages capturing the semantics of a sensor and its deployment. The following subsections detail key points encountered in the development of the SMR

A. Structure

The structure of the repository must be as general and flexible as possible. The structure may be deemed general if the model structure fits the requirements of diverse groups of environmental scientists whilst still capturing all the semantics of the sensor and its deployments. The structure is deemed flexible if the entities can be added to and removed from the model easily with minimal changes being made to the entire model. Figure 1 shows one such sensor directory model (as implemented for SwissEx) which is both general and flexible. It shows a set of sensors with unique serial numbers (C) connected to sensor stations (A) with a specific action (deployment, repair, observation, removal) (B). A station is defined as one or more sensor attached to a common logger or data communications platform. Figure also shows a group of sensors attached to the model (D) to which they belong to. Multiple database parameter names (E) are attached to sensor from which they derive their data.

B. Use of Semantics

Choosing the attributes to use as semantic annotations is very important. A model having no relevant semantic

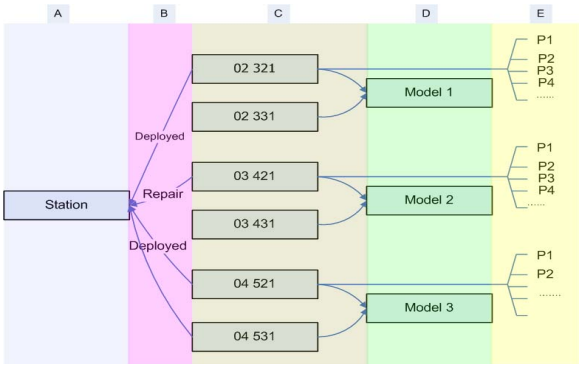


Fig. 1. Sensor directory model

annotations cannot capture the semantics of the sensor and its deployments, leading to poor metadata management and data provenance. The sensor attributes as semantic annotations are best defined by the domain experts, e.g. environmental scientists in the context of SwissEx. The domain experts deal with the data and know what kind of annotations would help them to document the sensor metadata and how the data provenance should be defined.

Table I shows the semantic annotations chosen for each of the model entities in SwissEx.

TABLE I
Semantic Annotations for Model Entities

Label	Model entities	Semantic Annotations
A	Sensor Station	Station name, General information, Geographical Coordinate
B	Sensor-station action	Station name, Sensor serial no, Sampling rate, Action (deployment, repair, removal, observation), Action date
C	Sensor	Sensor Serial no, Model no
D	Sensor Model	Model no, Manufacturer, Measured quantities, Units
E	Virtual Sensor	Paramater name, Sensor serial no, Measured quantity, Units, Sampling frequency, Measurement accuracy

Five types of semantic templates were created. Templates define the formatting of the page with semantic attributes embedded in them. Each one represents a model entity as shown in Figure 1. Semantic forms were provided as an user interface which uses semantic templates to create semantically annotated pages. The pages created contain the formatting and attributes specified in the templates. Multiple semantically identical instances of an entity can be created using this mechanism, for example: multiple instances of a sensor entity can be created, each instance is annotated with annotations provided in Table I. The implementation of the semantic entity creation is shown in Figure 2. Label (B) in the figure is the semantic template which contains HTML tags and CSS for the formatting. Importantly, it contains semantic (attribute, value) pairs,

e.g. [[Serial No:={{{sno}}}]], where ”Serial No” is the sensor semantic attribute and {{{sno}}} is the variable which

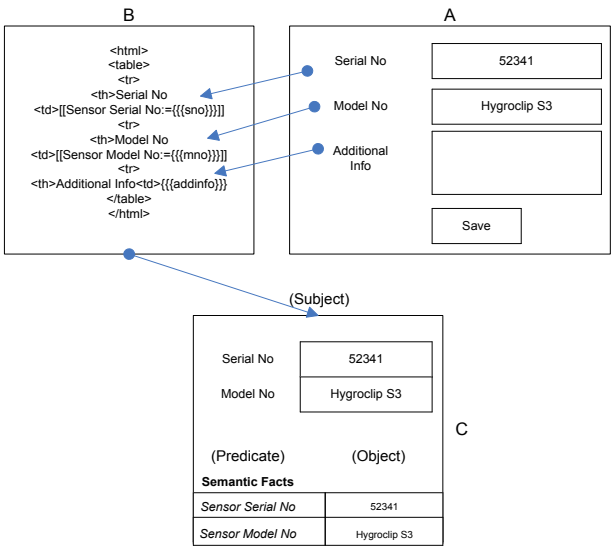
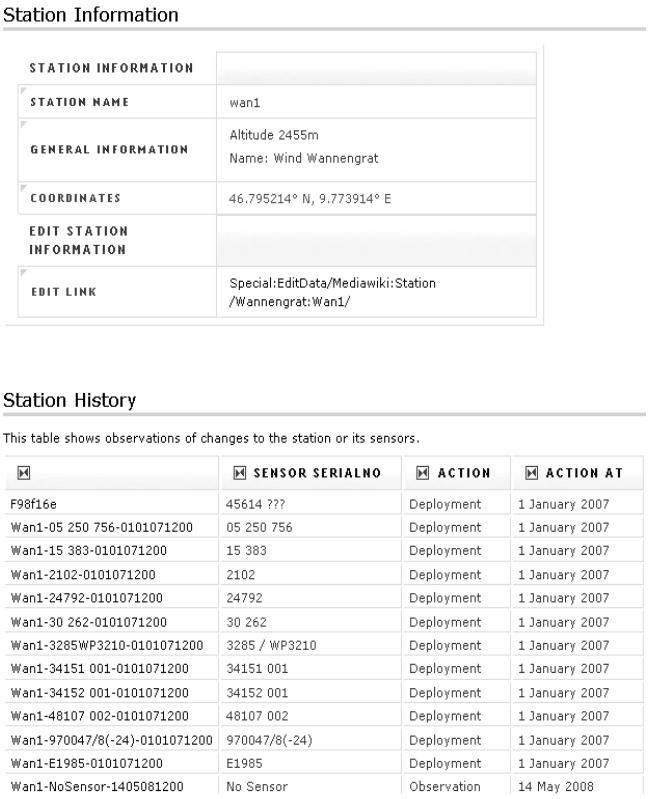


Fig. 2. A) Semantic Template B) Semantic Form C) Semantic Annotated Entity (Wiki page)

contains the value. When the user completes the form (A) and submits it, corresponding variables in the semantic template (such as {{sno}}} and {{{mno}}}) get the values from the form fields together with the defined formatting and create the semantic entity (Wiki page). The Wiki page represents the subject, where as the semantic attributes represent the predicates and their values represent the objects as shown in Figure 2 with label (C). Common semantic attributes between the entities make them semantically related to each other and using these attributes, information from the entities can be easily queried. These queries can be embedded in the semantic templates, so that each entity created using the templates shows the information returned by the query. The information returned by the embedded query is automatically updated whenever there is a change in information (addition, deletion, modification) in the repository. In this way the log of sensor deployments can be maintained with the deployment date and the action to which scientists can refer to, while carrying out their experiments. In the following, we give a sample query embedded in the semantic template of a station, which retrieves (satisfying the given condition) the list of sensor "Serial No", "Action" performed and the date "Action at" on which the action was performed.

```
{{#ask:[[station name::{{{stationname}}}]]
[[Sensor serialno::+]]|?Sensor serialno|
?Action|?Action at|format=table}}
```

Figure 3 shows the embedded semantic query results at the bottom of sensor station page (semantic entity). This information in the station page shows the history of the station. If 'No Sensor' is included as the serial number (see bottom of list), the action refers to the station in general.



Station Information

STATION INFORMATION	
STATION NAME	wan1
GENERAL INFORMATION	Altitude 2455m Name: Wind Wannengrat
COORDINATES	46.795214° N, 9.773914° E
EDIT STATION INFORMATION	
EDIT LINK	Special:EditData/Mediawiki:Station /Wannengrat:Wan1/

Station History

This table shows observations of changes to the station or its sensors.

	SENSOR SERIALNO	ACTION	ACTION AT
F98f16e	45614 ???	Deployment	1 January 2007
Wan1-05 250 756-0101071200	05 250 756	Deployment	1 January 2007
Wan1-15 383-0101071200	15 383	Deployment	1 January 2007
Wan1-2102-0101071200	2102	Deployment	1 January 2007
Wan1-24792-0101071200	24792	Deployment	1 January 2007
Wan1-30 262-0101071200	30 262	Deployment	1 January 2007
Wan1-3285WP3210-0101071200	3285 / WP3210	Deployment	1 January 2007
Wan1-34151 001-0101071200	34151 001	Deployment	1 January 2007
Wan1-34152 001-0101071200	34152 001	Deployment	1 January 2007
Wan1-48107 002-0101071200	48107 002	Deployment	1 January 2007
Wan1-970047/8(-24)-0101071200	970047/8(-24)	Deployment	1 January 2007
Wan1-E1985-0101071200	E1985	Deployment	1 January 2007
Wan1-NoSensor-1405081200	No Sensor	Observation	14 May 2008

Fig. 3. Example Wiki page

C. Import/Export

Sensor metadata collected through the SwissEx SMR is to be integrated with the following data sources:

- Distributed GSN data sources (data provenance)
- Microsoft SensorMap (visualization, data tagging)

The semantic Wiki stores all of the data in the form of RDF triples, with the Wiki page as a subject, semantic attributes as predicates and their values as objects. All this information is usually distributed over numerous database tables. Because of the huge number of parameters and their distribution among a number of tables, it is very difficult to write queries to extract the metadata by joining the tables to satisfy a large number of conditions. Environmental scientists should not have to learn the query syntax or have to deal with such complex queries. Moreover, running queries directly into the main database is not recommended because of security and safety issues. This issue is addressed by providing a Netapi[4] SPARQL endpoint to the Wiki. SPARQL is a query language for RDF, more precisely it is a query language and protocol for RDF. As the semantic Wiki has the capability of storing the data in the form of RDF graphs, SPARQL is the ideal language to query it. In the following, we provide a sample SPARQL query which retrieves all of the distinct sensor models in the repository:
Query:

[4]http://www.w3.org/Submission/2003/SUBM-rdf-netapi-20031002/

```
PREFIX a:
<http://www.swiss-experiment.ch/index.php/
Special:URIResolver/>

SELECT DISTINCT ?smodel
WHERE {?page a:Property:Sensor_model
?smodel}
```

Using SPARQL, one can query multiple sources at once in a single query and get the returned data exported through the web endpoint. Consider a situation where two different groups of environmental scientists are using two different Wikis with the SMR implemented in them, each capturing the semantics of different processes. Suppose their repositories contain metadata which compliment each others processes and they want to merge them and integrate the result with another external data source. In this case a simple SPARQL query is needed which would consider both the Wikis as two different (meta)data sources, retrieves both sets of metadata at once and merges them to satisfy a given condition. The external data source would then be able to receive this SPARQL returned metadata through the web endpoint and integrate it with its data. The sample query to retrieve the combined list of sensor models from projects experiment1 and experiment2 is provided below:

```
PREFIX a:
<http://www.experiment1.ch/index.php/
Special:URIResolver/>
PREFIX b:
<http://www.experiment2.ch/index.php/
Special:URIResolver/>

SELECT ?smodel
FROM NAMED <wiki1.rdf>
FROM NAMED <wiki2.rdf>
WHERE {
  GRAPH <wiki1.rdf> {
    ?x a:Property:Sensor_model ?smodel .
  } .
  GRAPH <wiki2.rdf> {
    ?y b:Property:Sensor_model ?smodel .
  } .
}
```

D. Database Migration

It has been found that scientists who are already using traditional relational data models for storing sensor metadata would like to migrate to the idea of semantic SMRs for ease of data entry (and hence to increase the amount of metadata entered). Limitations in doing so would discourage them to migrate towards better technology. There should therefore be enough flexibility in the tool to enable this migration. Consider a case where scientists have large amount of metadata in their relational database and many external applications accessing and using it. This poses problems in completely migrating to a different platform like the semantic SMR. In this case they would need a mechanism which would:

- Map the relational data model schema to the SMR model.
- Update the semantic SMR whenever there is change in the relational database.
- Update relational data model whenever there is change in the semantic SMR, so that external applications connected to the relational data model can receive the updated data without getting disrupted.

The issue of migration from relation DB based repositories (data models) to more advanced semantic SMRs is addressed below by discussing two solutions:

R2D2 - RDF to Database too [20] is a mechanism which provides mapping of the RDF query to a relational DB without replicating the data. It comes with the D2RQ Mapping Language - a declarative mapping language using which one can specify the relation between an ontology and a relational data model. Using this, one can write a mapper file which would map the relational data model to the ontology. This way one can query the metadata in a relational data model using SPARQL queries which will automatically be translated into an SQL query. This solution is ideal in the case where data in the relational data model is of manageable size and can be completely migrated at once to semantic SMRs. This is not ideal in the cases where one may want to update the relational data model to the changes in semantic SMR using RDF queries.

Two Way Synchronization: Current generation Wikis provide the mechanism of exporting the pages in different formats, e.g. RDF, and importing pages from same file formats. Many of them also provide wiki maintenance scripts which include scripts for importing external files to create Wiki pages programmatically. The semantic Mediawiki provides similar mechanisms and maintenance scripts, e.g. “importDump.php” which imports RDF files to create Wiki pages. These mechanisms may be used to provide a simple and effective two way transformation from semantic SMR model to relational data models and vice versa.

Relational data model to semantic SMR: the following points detail the steps required to import and update the metadata from the relational data model into the semantic SMR:

1) A semantic SMR model should be designed, which may be mapped to the relational data model schema.
2) Suppose we have four different semantic entities in the SMR, e.g. Station, Sensor, Model and Action. Using the export utility in the wiki, RDF files should be exported for each of the different page types.
3) A script should be written, which carries out the following at regular intervals:
   - Uses the exported RDF files (step 2) as templates.
   - Updates these templates by writing the data from relational DB tables to the data fields in the template RDF file.
   - Imports the updated RDF files into the wiki using “importDump.php”.

Semantic SMR to relational data model: the relational data model can be synchronized to the changes in the semantic SMR by writing a wrapper script which would do the following:

- Execute the SPARQL query on the semantic SMR to retrieve all the recent updations.
- Collect the SPARQL returned data, wrap it into SQL queries and update the relational data model.

E. User Interaction

After a day in the field, the scientist does not wish to spend hours entering metadata and so if the process is inefficient, the metadata will not get entered. This is addressed in the SMR by using the semantic form extensions combined with automated page creation links. When the user selects the entity they want to create, a wiki page is automatically created containing the correct form type. The user must then fill in the form and select save. Entering metadata could not be simpler or less thought intensive. For viewing the metadata, the SMR has 5 interfaces, corresponding to the different entities: stations, models, serial numbers, database parameters and observations/actions. Figure 1 shows five regions (A, B, C, D and E) which represent the different types of interfaces.

VI. Integration of SMR with Microsoft SensorMap

SensorMap is a platform for publishing, browsing and searching for streaming sensor data based on a geo-spatial interface using the Microsoft Virtual-Earth engine. It is also capable of generating visualizations based on a grid interpolation between sensors and overlaying them on top of a topographical model. "Time traveller" is a feature of SensorMap which allows users to select any time point and see the spatial distribution and sensor data. This helps scientists to analyze the spatial and temporal variations in the data at once which can help to increase awareness of the processes which are occurring. This data does not however, contain any additional information supporting it, for an instance information such as "what was the condition of the sensor responsible for that data at that particular time point". The scientist looking at this data with no previous knowledge of the sensors could draw wrong conclusions if it was assumed that all of the sensors were behaving in the same way. Data tagging is therefore required to help scientists to get more information about the data indicating its origin and external conditions influencing it. The metadata collected through the SMR is integrated with SensorMap. This allows users to visualize the metadata along with the data at the particular time point. The purpose of the integration is to provide all of the information on the sensor, for example: type, accuracy etc, along with the observations (repairs, problems etc.). The user can then see what has caused anomalies in the data. Figure 4 shows the overall workflow of the system.

VII. Future Scope

With the powerful semantic SMR implemented, the following tasks are envisaged as further development of the system:

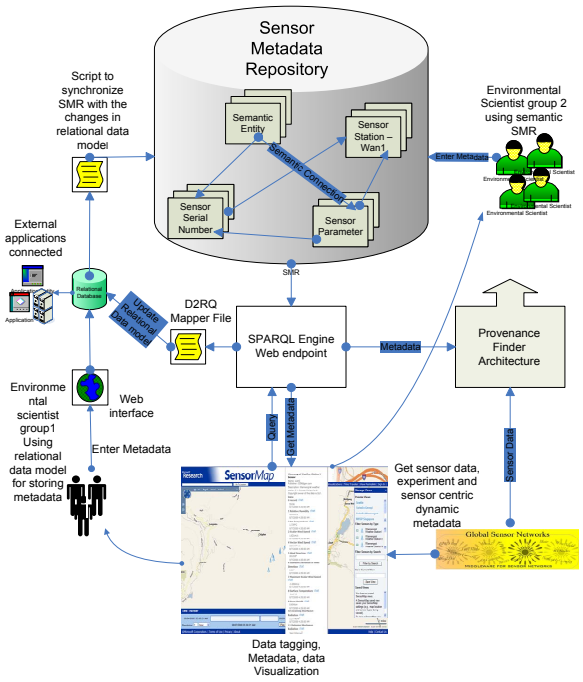


Fig. 4. Workflow Diagram

- **Integration with GSN middleware data databases:** In short, GSN (Global Sensor Networks) [21] is a software middleware designed to facilitate the deployment and programming of sensor networks. At present diverse groups of environmental scientists within SwissEx are using it for acquisition of data, there are therefore distributed GSN data databases storing large amounts of real-time sensor data. This metadata will be integrated into the SwissEx semantic SMR with the distributed GSN data databases for data provenance.
- **Automatic generation of GSN virtual sensor descriptors:** GSNs concept of virtual sensor abstraction enables the user to declaratively specify XML-based deployment descriptors. As there is a huge amount of sensor specific metadata within SwissEx, we plan to use this metadata coupled with a mechanism to dynamically generate the XML-based virtual sensor deployment descriptors. This will be more efficient, providing the user with a predefined skeleton of the deployment descriptor containing much of the deployment specific information in it.

VIII. Conclusion

This paper has considered the generation, storage and use of metadata of different provenance and has provided a solution which is both flexible and general enough for use throughout the environmental science domain.

A tiered metadata recording system was presented: a non-semantic wiki was used for the background experimental metadata; a semantic wiki was used for the metadata which could

be directly related to a single sensor; a relational database containing an algorithm based continuously sampled dataset was considered for metadata which could be directly related to a single data point (the quality data). This system provides the ideal storage method for each of the data types.

Semantics are a good way of recording textual information in a structural format, but on its own, a semantic wiki is somewhat 'freeform'. When semantic forms are used in the SMR, an elegant solution for recording metadata is produced. These forms can be configured to user requirements so that metadata may be entered in pre-defined fields (which can be mapped to a fixed field relational database if required), hence the metadata recorded is common over all sensors. When the metadata fields are common, querying the data becomes simple. Queries may be carried out using SPARQL in order to insert the relevant metadata alongside the data in the data analysis tools. Methods for doing this were presented.

The SMR using semantic forms, together with the automated creation of the Wiki pages is both simple and efficient, making it attractive to the scientist to use. Training is however required educate scientists into this electronic way of working.

Solutions for the propagation of the metadata into external tools and database structures were also discussed. Methods presented showed that the method of recording metadata using semantics can be used as a GUI for existing relational fixed field databases (both for displaying existing data and for entering new data) and the metadata can also be utilised in data plotting tools, such that the relevant metadata is displayed to the data analyst when the data is queried. If the system is used effectively (metadata is entered when observations affecting the data are made or when metadata on the overall sensor deployment will later be important in the correct interpretation of the data) then this will enable users outside the original project team to use the data without prior knowledge (other than the metadata provided) of the sensor system and will eliminate false interpretation of the data.

This paper may be concluded with an example of the overall SMR use:

A database of hundreds of meteorological stations from across Switzerland exists as an Oracle database. Contained within this database is a metadata schema. The stations are maintained by local personnel, hence the database team wishes to have a web based interface for entering metadata (observations), but wish to maintain their current metadata schema which is used by various models and automated reporting tools.

Using the SMR, the database team can configure the semantic forms to suit their database structure and then import the data from the database to the SMR using R2D2. The station maintainers then have a fully populated SMR, to which they can easily add their observations by clicking on the relevant link, filling out the fields provided and saving them. Whenever a change is made to the SMR, R2D2 automatically exports the data entered to the oracle database.

The database of stations is read by a sensor middleware (GSN) and imported into SensorMap. This provides a geospatial interface for accessing and querying the data. If a user queries a station, SensorMap also queries the SMR to obtain the corresponding metadata for that station and sensor over the specified time period. SensorMap also provides a link to the relevant SMR page URL should the user want to access metadata from any other time period. In addition to plotting the data in SensorMap, the user can also plot the quality data alongside, hence in a single station query, the user gets a plot of both the data and its associated data quality annotated with observations and static metadata such as the measurement error. As this data is used by institutions from across Switzerland (or even further afield), this metadata is crucial to the correct interpretation of the data.

REFERENCES

[1] R. S. Barga and L. A. Digiampietri, "Automatic generation of workflow provenance," in *IPAW*, 2006, pp. 1–9.

[2] B. Glavic and K. R. Dittrich, "Data provenance: A categorization of existing approaches," in *BTW*, 2007, pp. 227–241.

[3] P. Buneman and W.-C. Tan, "Provenance in databases," in *SIGMOD '07: Proceedings of the 2007 ACM SIGMOD international conference on Management of data*. New York, NY, USA: ACM, 2007, pp. 1171–1173.

[4] S. Vansummeren and J. Cheney, "Recording provenance for sql queries and updates," *IEEE Data Eng. Bull.*, vol. 30, no. 4, pp. 29–37, 2007.

[5] S. Miles, S. Munroe, M. Luck, and L. Moreau, "Modelling the provenance of data in autonomous systems," in *AAMAS*, 2007, p. 50.

[6] S. Bowers, T. M. McPhillips, M. Wu, and B. Ludäscher, "Project histories: Managing data provenance across collection-oriented scientific workflow runs," in *DILS*, 2007, pp. 122–138.

[7] S. Miles, S. C. Wong, W. Fang, P. Groth, K.-P. Zauner, and L. Moreau, "Provenance-based validation of e-science experiments," *Web Semant.*, vol. 5, no. 1, pp. 28–38, 2007.

[8] S. B. Davidson, S. C. Boulakia, A. Eyal, B. Ludäscher, T. M. McPhillips, S. Bowers, M. K. Anand, and J. Freire, "Provenance in scientific workflow systems," *IEEE Data Eng. Bull.*, vol. 30, no. 4, pp. 44–50, 2007.

[9] S. Miles, S. C. Wong, W. Fang, P. T. Groth, K.-P. Zauner, and L. Moreau, "Provenance-based validation of e-science experiments," *J. Web Sem.*, vol. 5, no. 1, pp. 28–38, 2007.

[10] S. Miles, P. T. Groth, M. Branco, and L. Moreau, "The requirements of using provenance in e-science experiments," *J. Grid Comput.*, vol. 5, no. 1, pp. 1–25, 2007.

[11] Y. L. Simmhan, B. Plale, and D. Gannon, "A survey of data provenance in e-science," *SIGMOD Rec.*, vol. 34, no. 3, pp. 31–36, 2005.

[12] J. Zhao, C. A. Goble, R. Stevens, and S. Bechhofer, "Semantically linking and browsing provenance logs for e-science," in *ICSNW*, 2004, pp. 158–176.

[13] G. Fox, "Data and metadata on the semantic grid," *Computing in Science and Engineering*, vol. 5, no. 5, pp. 76–78, 2003.

[14] http://datafedwiki.wustl.edu/index.php/DataFed_Wiki.

[15] http://wiki.myexperiment.org/index.php/Main_Page.

[16] http://kepler-project.org/.

[17] M. Krötzsch, D. Vrandecic, M. Völkel, H. Haller, and R. Studer, "Semantic wikipedia," *J. Web Sem.*, vol. 5, no. 4, pp. 251–261, 2007.

[18] K. S. Candan, H. Liu, and R. Suvarna, "Resource description framework: metadata and its applications," *SIGKDD Explor. Newsl.*, vol. 3, no. 1, pp. 6–19, 2001.

[19] M. Lehning, P. Bartelt, R. Brown, C. Fierz, and P. Satyawali, "A physical snowpack model for the swiss avalanche warning services. part iii: Meteorological boundary conditions, thin layer formation and evaluation," *Cold Reg. Sci. Technol.*, vol. 35, no. 3, pp. 169–184, 2002.

[20] http://aksw.informatik.uni-leipzig.de/Projects/R2D2.

[21] K. Aberer, M. Hauswirth, and A. Salehi, "Infrastructure for data processing in large-scale interconnected sensor networks," in *MDM*, 2007, pp. 198–205.