Automating the Collection of Semantic Sensor Network Metadata in the Field with Mobile Applications

Laura Kinkead[1], Paulo Pinheiro[1], Deborah L. McGuinness[1]

Rensselaer Polytechnic Institute, Troy NY 12180, USA

Abstract. In the past few decades, the field of ecology has grown from a collection of disparate researchers who collected data on their local phenomenon by hand, to large ecosystems-oriented projects partially fueled by automated sensor networks and a diversity of models and experiments. These modern projects rely on sharing and integrating data to answer questions of increasing scale and complexity. Interpreting and sharing the big data sets generated by these projects relies on information about how the data was collected and what the data is about, typically stored as metadata. Metadata ensures that the data can be interpreted and shared accurately and efficiently. Traditional paper-based metadata collection methods are slow, error-prone, and non-standardized, making data sharing difficult and inefficient. Semantic technologies offer opportunities for better data management in ecology, but also may pose a challenging learning curve to already busy researchers. This paper presents a mobile application for recording semantic metadata about sensor network deployments and experimental settings in real time, in the field, and without expecting prior knowledge of semantics from the users. This application enables more efficient and less error-prone in-situ metadata collection, and generates structured and shareable metadata.

1 Introduction

Over the past few decades, the field of ecology has expanded immensely in scope. The field now tackles scientific questions that address wide ranges of complexity, scale, and time spans [12], [14]. Some of these questions are tackled by large project groups, and others are answered by combining data sources from multiple disparate projects. Despite the essential need for combining data from multiple heterogeneous sources, it is evident that there is no systematic approach for ecologists to share data and data workflows that takes into consideration the ecologist's in-situ knowledge about observations and experiments. More specifically, for projects utilizing sensor networks, there is no systematic approach to capture the ecologist's in-situ knowledge about sensor deployments and configurations such that the knowledge can inform data analysis and management decisions. The lack of contextual knowledge about in-situ scientific activities affects the way data should be further interpreted and analyzed. For example, the data generated by an improperly calibrated instrument or an instrument not correctly placed on its platform should not be used in rigorous statistical analysis. Similarly, configuration settings of sensor network equipment may be changed in the field, which affects the resulting data product and the manner in which it should be analyzed.

We introduce a semantic approach to mediate data usage and sharing by encoding the knowledge about the sensor network that generated the data. In particular, we encode knowledge about the deployments of platforms, instruments, and detectors in support of the data products they generate. With this kind of knowledge, ecologists can better understand the context of their data collection. This information allows scientists unfamiliar with the original collection to make appropriate use of the data [12]. Thus, the availability of semantic annotations encoded as metadata increases the length of time that a data product is useful, as the data no longer relies on the presence of the people who were involved in the data collection process to explain how the data came to be. Whereas a file without metadata may become useless once its originator changes jobs or forgets some details, a file with metadata may be useful for many decades [12], [5]. Extending the longevity of data is especially important as ecological projects look to answer questions spanning long periods of time. In addition, metadata allows a data product to be used in the future to answer unanticipated questions [11]. In this paper, we describe a semantic technology-based mobile application that enables on-site capture of metadata in real time. By embedding the metadata capture system in a mobile device, much of the metadata capture can be completed automatically. The application uses a QR code-based sensor identification system, which automates the identification of sensor equipment and minimizes errors in data entry.

Our work overcomes three major challenges. First, our framework is a solution for overcoming the barrier of entry to semantic technologies for field scientists. This tool makes the collection of standardized and machine readable metadata efficient for the practicing scientist, thus enabling the practicing scientist to benefit from the advantages of semantic technologies without having any prior knowledge of the technology. Second, it describes and implements a framework for a method of using semantic technologies on a mobile platform to record data in real time in the field. This framework makes metadata capture more efficient and less error-prone compared to traditional recording methods. In addition, this method description and implementation has the potential to be broadly reused by a wide range of observational efforts. Finally, this application addresses context-specific challenges to make it more likely for valuable in-situ knowledge to be captured.

2 Challenges of Collecting In-Situ Contextual Knowledge

2.1 Challenges in Collecting Contextual Knowledge

The onset of automation in ecological data collection means that metadata about in-situ contextual knowledge, including the knowledge about the sensor network collecting the ecological data, is more important than ever before. A scientist needs metadata not only for remembering how to use a data set that was collected years ago, but also for simply understanding how to use the data that was collected by an autonomous sensor earlier the same day. Automated sensor equipment collects data at an unprecedented rate, and it is necessary to have a well-structured system for capturing all of the contextual knowledge surrounding each data collection activity so that the data can be accurately used. Metadata is also critical for sharing data sets between research groups who are unfamiliar with the details of each others' work. For these reasons, metadata is widely recognized as a critical component to ecological data management [12], [5], [11].

Semantic technologies can be and are being used to support improved knowledge sharing: these technologies provide a standardized framework for storing and sharing metadata, and they make data interpretation and use more efficient [14]. However, creating semantic metadata is typically a slow and tedious process for a human to do manually. Generating semantic data often requires much technical knowledge, and the final product must be correct and complete if it is to be used effectively by applications. While semantic technologies ease the workload of the data consumer by improving integration and understandability, they are often perceived as a burden by the data generator. These difficulties present a large barrier to entry to many scientists, and many find that the challenges associated with adopting this new technology are not worth the benefits. These problems present a very real barrier for semantic technology adoption in non-computer science fields.

While very valuable, collecting well-structured and comprehensive contextual metadata can be expensive. At one point in time, requiring a scientist to take thirty minutes to create a thorough metadata document was thought to be worth the future value of that document [5]. However, the onset of large, automated data-collection systems that generate data sets by the minute renders such a value judgment unreasonable; such a researcher would be overwhelmed by the rate of data collected by the network [12].

2.2 The Additional Challenges of In-Situ Knowledge Collection

Much of the most valuable contextual knowledge needs to be collected in-situ: while a field scientist is actively collecting samples or making changes to automated equipment. Contextual knowledge such as the GPS points at which a sample was collected, the serial number of instrument was used to collect a sample, or what program was selected for an on-board computer is best written down immediately to ensure the information is accurate and to ensure that the knowledge is recorded at all.

Thus, while metadata recorded in the lab faces an efficiency problem at the level of requiring time of a researcher who is sitting at his desk with competing activities to do, collecting in-situ contextual knowledge has even more challenges. The field is not an ideal place to record data: flat, dry surfaces are difficult to come by, and a field scientist is often in a rush to complete a set of tasks not just before lunch break, but before the sun goes down or before it rains. Traditionally, field scientists have solved this problem by collecting metadata very quickly with pen and paper, without always applying consistency or completeness to the process. This strategy leads to situations such as the one experienced by our group recently in which we received sixteen boxes of field notes containing handwritten metadata for a thirty year observational study. In its handwritten form, this data is challenging to utilize effectively. Thus, we need a compromise between efficiency for the knowledge recorder and usability for the knowledge receiver.

Fortunately, creating shareable, machine readable data is a forte of machines. The emergence of powerful yet affordable handheld devices, namely mobile phones, presents an opportunity for more efficient and less error-prone data collection in-situ. Such tools can validate input to ensure that metadata adheres to the structure of selected metadata standards. In addition, rather than recording one's own metadata in the field on paper and then later trying to fit that data into a standard (and perhaps losing information in the transition), semantic tools help scientists collect the right information in real time and on site. These software tools may also include error-checking mechanisms, such as checking for out-of-bounds values, to ensure that collected metadata is sensible and to highlight mistakes for immediate correction [6].

3 Ontology-driven Contextual Knowledge Capture

This work is performed in the context of the Jefferson Project, a collective effort of Rensselaer Polytechnic Institute, IBM, and The FUND for Lake George. The Jefferson Project studies Lake George in New York state as a model ecosystem and aims to apply the findings to freshwater resource management worldwide [10]. One component of the Jefferson Project is a large network of sensors stationed around the lake and its watershed. These sensors collect data on a variety of attributes, such as the region's weather, the lake's chemistry and currents, and even populations at the lowest levels of the food web. The state and arrangement of these sensors will change many times over the course of the project: instruments will need to be taken back to the lab for calibration, fixed when malfunctioning, upgraded, or may be deployed elsewhere. Throughout all of this activity, the metadata about the position of all of the sensor equipment will need to be recorded so that the data sets the sensors generate can be correctly interpreted. Therefore, we decided to focus our metadata capture application on collecting metadata about sensor deployments because this metadata will be collected many times over the course of the project, thus it will benefit greatly from being automated.

3.1 The Human Aware Sensor Network Ontology

We use the Human Aware Sensor Network Ontology [13], or HASNetO, to encode our metadata. By using a semantic approach, we make the interpretation of the metadata less subject to misleading interpretations, and make it possible for machines to read and leverage the knowledge in the process of managing the data.

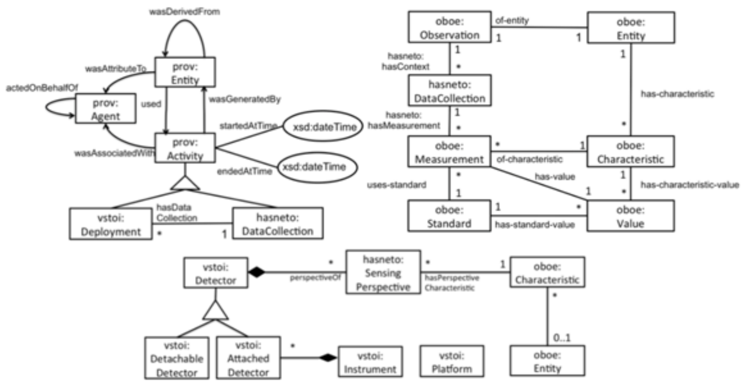

Fig. 1. The Human Aware Sensor Network Ontology (HASNetO) [13]

In related work, concepts from the Extensible Observation Ontology (OBOE) [8], the PROV Ontology (PROV-O) [7], and the Virtual Solar Terrestrial Observatory Ontology (VSTO) [9] were leveraged to build the Human-Aware Sensor Network Ontology, or HASNetO [13]. Notably, we did not incorporate the Semantic Sensor Network Ontology (SSN) into HASNetO because although SSN uses similar concepts, it is not suited to our work because is does not talk about human agents and their involvement in the process of managing sensor networks. HASNetO groups sensor network equipment into three types: detectors, instruments, and platforms. *Detectors* are the objects that do the sensing: they convert the physical signals about the characteristic of interest into a (most often electric) signal that can be read by a computer or human. *Instruments* are the objects that support the detectors. They do not do any sensing themselves, but they provide the framework in which the detector captures signals, and convert the detector's signal into a data point. A *platform* is the object that determines the location of the instrument, whether it be the point of a stationary platform or the path of a mobile one. A platform may also provide overhead services, such as providing the instrument with power, a data connection, and protection against natural and human hazards. In HASNetO, a *deployment* is composed of one platform, one instrument, and one or more detectors. A deployment also has a start time and an end time.

3.2 MOCCASN: Mobile Context Capture for Sensor Networks

Our solution for collecting in-situ contextual knowledge is MOCCASN, an Android application. MOCCASN makes collecting metadata very quick, and it does not require the scientist using the application to have any knowledge of the underlying semantic technology. Using the phone's camera, the MOCCASN identifies sensor network objects via QR codes that are affixed to each object. An object's QR code contains the URI of the object's instance of a HASNetO concept. From the URI, one can retrieve instance properties such as the serial

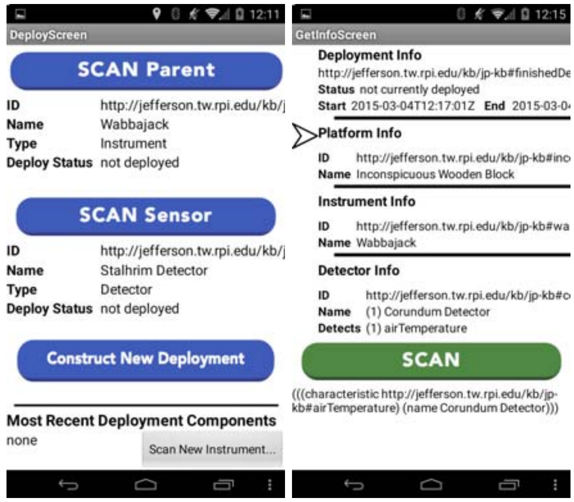

Fig. 2. Constructing a new deployment and getting information about a deployment

number and type of sensing device. The scientist can use MOCCASN to scan an object's QR code and retrieve information about the object, to start a deployment with the object, or to end the object's current deployment. When starting and ending deployments, MOCCASN uses the phone's built-in GPS to automatically assign latitude and longitude coordinates. Time and user log-in information are automatically documented as well. All of this information is automatically recorded in accordance with the HASNetO ontology. The scientist is guided through the process of creating and ending deployments via a dynamic graphical user interface. MOCCASN accomplishes these tasks by communicating with the online knowledge base through a cellular or wifi connection. When the scientist does not have a data connection, records may be saved locally to the phone for later submission. While in communication with the knowledge base, MOCCASN initiates error checks on the scientist's input based on the semantics encoded in the ontology.

The GitHub page for MOCCASN may be found at [1], and a static demo page at [3].

3.3 Software Development Tools

MOCCASN was developed with MIT App Inventor, on online tool with an intuitive interface for developing Android applications quickly and easily [2]. MIT App Inventor made it possible for MOCCASN to be developed in a matter of weeks, and we believe that MIT App Inventor can be used to rapidly create similar specialized applications to match the needs of a variety of observation-based efforts.

4 Results

MOCCASN meets a number of requirements of in-situ metadata collection, including deployment-specific requirements. First, deployments must be able to be assembled in a piecemeal fashion. For example, the user may want to connect the instrument-to-be-deployed with its associated detectors while still on shore, and then connect the instrument on the platform at a later time. We use an existing term in the OBOE ontology, "hasCode", to "tag" partial deployments in the knowledge base as "under construction". With this method, a scientist may connect an instrument and detector in the field and enter this partial deployment in the knowledge base as a "deployment under construction". When that instrument is later brought into the field to be connected to a platform, the tool automatically finds that the instrument is part of a "deployment under construction", and adds the platform to the same deployment. When the user starts the deployment, the "under construction" tag is removed from the deployment. This method ensures that all of the application users and the knowledge base always have the same information.

Second, a data or wifi connection cannot be relied upon. It is common for field sites to be located in areas with no data connection. Therefore, all data must be recorded in a sensible fashion even when the scientist is unable to connect to the knowledge base. MOCCASN meets this requirement by offering a "Save Deployment Data" button, which copies the recorded information to local storage, such as the scanned URIs, the username, and the time. The scientist can view this saved information in the "View Un-Submitted Records" screen, and the scientist may re-submit any of these records when he or she returns to cell service by clicking on the record and clicking "Re-Submit". Upon resubmission, the application will initiate all necessary error-checking routines and attempt to write the data to the knowledge base.

Reasoning in support of error checking is performed when annotated data arrives in the data repository. Reasoning capabilities are limited on the device itself as much of the error checking involves comparing the scanned information against information currently in the data repository. Therefore, when MOCCASN does not have a connection to the knowledge base, the application cannot execute error-checking on the recorded information in real time. Thus, it is possible that locally stored information may contain information that is inconsistent with the information in the knowledge base. These inconsistencies may only be identified when a data connection is regained, and will prevent the data record from being submitted. However, this information is likely still valuable even though it is not completely valid; portions of the data record are likely accurate. To make use of this partially correct information, the scientist is offered the option of emailing the record so that the data record can be corrected and entered manually. This feature ensures that no data that was collected in the field is ever lost.

In addition, by allowing the user to save deployment information, the application is more robust to errors caused by the knowledge base becoming out-of-sync with the true state of the sensor network. For example, consider a situation where a user attempts to end the deployment of an object, but upon scanning the object, the application finds that the object is not currently deployed. This may happen because the scientist who set up the deployment forgot to click the "Start Deployment" button. The current scientist is correct to want to enter the deployment end time information, but the application will not let the scientist do so due to inconsistent information in the knowledge base. The missing deployment start information must be entered into the knowledge base by hand before the app will enter the deployment end information. To accommodate this type of situation, the user may save the end deployment information locally, and re-submit it once the error has been solved manually.

We now introduce a number of use cases for the ways in which MOCASSN has been used to read and write metadata while in the field.

4.1 Use Case: Constructing a Deployment

To construct a deployment, the scientist scans two directly sensor objects that are to be directly attached to each other. For example, a scientist may scan an instrument and one of its detectors, or a platform and its instrument. Every time the scientist scans an object, the application queries the knowledge base to check that the object is in the knowledge base, that it is either a type of instrument, detector, or platform, and that it is not currently deployed (an object must be un-deployed before it may be re-deployed). When a pair of objects is scanned, the application queries to check that the types of the two objects are compatible (an instrument must have a parent of type platform, and an detector must have a parent of type instrument). It also checks that either one of the objects or neither of the objects is part of a deployment under construction.

According to HASNetO, a deployment is comprised of one platform, one instrument, one or more detectors, a start time, an end time, a location, and the deploying scientist. The platform, instrument, and detector objects of the deployment are assembled in a piecemeal fashion. During the time when the deployment is incomplete because all of the pieces of information have not yet been submitted, the deployment is tagged with the code "under construction" to denote that the deployment is not consistent with the HASNetO definition of a deployment. After the remaining deployment information is recorded, the scientist clicks "Start Deployment" to add the end time and remove the "under construction" tag.

4.2 Use Case: Ending a Deployment

To end an ongoing deployment, the scientist scans any of the objects associated with the deployment, and the application adds an end time to the deployment. The application will not allow the scientists to end a deployment that is not currently underway.

4.3 Use Case: Getting Information about a Sensor

The application may also be used to retrieve information about any piece of sensor network equipment. For scientists with no prior knowledge of linked data and the SPARQL language that is used to query linked data, this feature offers a simple solution for viewing deployment data in the knowledge base in a very readable format. This screen presents a number of details about the most recent (or current) deployment of the scanned object. For example, this screen displays the deployment's associated platform, instrument, and detectors, as well as the characteristics that are detected. In addition, the screen will show the deployment's start and end times if applicable.

4.4 Reflections on Field Testing

MOCCASN was field tested with a Jefferson Project field scientist. This scientist has been performing ecological research on Lake George for many years, and is part of the team deploying Jefferson Project sensor network equipment. Testing was performed with an HTC One M7 phone. After reviewing the structure of the application, the scientist started and ended about a dozen deployments. A few mistakes were made at first, such as forgetting to click "Start Deployment" after connecting all pieces of the deployment, and not waiting for a confirmation of data submission before moving on to make the next entry. These problems could be mediated by explicitly warning the scientist of the incomplete data entry when they attempt to move on too quickly. We estimate that it takes about fifteen minutes to introduce how to use the application, and about an hour of practice for the field scientist to get a good understanding of how to use MOCCASN proficiently.

5 Discussion

MOCCASN addresses issues related to barriers to entry, automatic metadata capture, and in-situ context capture. The application removes the barrier-to-entry that researchers often face when presented with semantic technology solutions. With this tool, a researcher can generate machine readable metadata without any prior knowledge of semantics.

Historically, progress on improving the way we model metadata knowledge has come at the cost of increased time spent capturing metadata and increasingly advanced formats. For a long time, the extra time required of the data generator to standardize data was well worth the extra effort because it enabled efficient data sharing. However, with the onset of automated data collection, researchers will simply be overwhelmed by the amount of metadata that needs to be generated. Our work developing MOCCASN counters the trend of increased capture time for the sake of data usability, reducing a researcher's metadata capture to a few QR code scans.

MOCCASN enables in-situ capture of contextual knowledge that would be lost otherwise. Researcher field time is incredibly valuable, and automated tools enable field scientists to quickly record valuable knowledge that would otherwise have been recorded on paper or not recorded at all in an easily usable and shareable format.

By automating metadata generation, we make a number of other advances in metadata capture. The application performs error-checking in real time by communicating with the data repository to help prevent erroneous data from being entered into the knowledge base. In addition, all metadata created with this tool conform to a standard vocabulary and are immediately accessible by anyone else on the team.

5.1 Value of Semantics

By automating the capture of contextual knowledge, we enable field scientists with no technical knowledge of semantic technologies to benefit from the value of semantic technologies. While the structure and capture of contextual knowledge is often standardized within a lab, it is not common for such metadata to follow broad community standards. This makes sharing datasets very difficult and time consuming, as a human must interpret each dataset to determine its usability and compatibility with other datasets. Semantic technologies turn the process of integrating datasets into a machine's task, which can be accomplished automatically, consistently, and thoroughly via the semantic comparison of dataset's contextual knowledge. For instance, machines can verify whether the contents of two datasets are semantically equivalent. Even if they are not equivalent, the machine can identify if any contextual difference is significant enough to enable or not the integration of the datasets.

A lab may feel confident in their current metadata practices for managing hand-collected data. However, for projects involving automated sensor networks, easily accessible and usable metadata is critical to harnessing the power of rapid data collection. Semantic technologies provide a solution for this new era in ecology because semantic metadata is structured and query-able, making it easy to access and use for data management and analysis.

5.2 Share-ability and Re-usability

MIT App Inventor makes it easy to share projects and to allow others to download the application to their phones. For those who wish to re-use this application, the source code file is available to reopen in App Inventor, from which point one can make changes to the interface and logic. Detailed instructions about sharing App Inventor projects are available here [4].

Since this application is based on a public set of ontologies, it may easily be re-used in projects that wish to use the same ontology to capture their deployment metadata. The application could be ready for a new use in just minutes by simply changing the endpoint URLs to a new project's knowledge base. Similarly, it would be relatively easy to make extensions to the application for small extensions required to the ontology. If a research team would like to collect metadata in a similar way, but with a different or modified ontology, it would still be useful to use this application as a starting point. Many of the queries that the app runs would likely need to be modified, but it may reduce development turnaround time to start with this app as a framework.

Our team has already found the need to extend MOCCASN and found the process to be very easy. While demonstrating MOCCASN to our field scientist teammates, we discovered that MOCCASN would be more useful to them if it collected information about samples in addition to information about deployments. After deciding on what terms from existing ontologies to use to represent sample-based knowledge, MOCCASN was extended to capture metadata about samples in just a few hours by adding "collect sample" and "analyze sample" screens.

5.3 Future Work

After having completed initial testing with a small collection of field scientists within our expanded team, we are beginning to deploy MOCCASN for real use in the Jefferson Project. The app will be in use among field scientists as they modify the arrangement of the equipment that autonomously monitors the lake, as well as researchers collecting samples in the field.

In addition, though this work focused on capturing metadata related to deployments, we plan to apply the same framework to rapidly develop additional tools for capturing a wide range of metadata. For example, we plan to build a similar tool to capture equipment calibration metadata in the lab, and for capturing the way sensor's configuration parameters are set. Both of these activities will be performed routinely over the course of the Jefferson Project to maintain a well-functioning sensor network, and the calibration and configuration parameters are important for accurately comparing and combining datasets. In all sorts of human interventions, we are also planning to provide richer provenance knowledge about how sensor calibrations, deployments and configurations are decided.

More broadly, we think it would also be valuable to add to this tool the ability to preview the deployed object's data stream. It is not uncommon to hear about a half of a day's field work lost due to improperly set up equipment. Since many sensor network instruments stream their data back to a central hub, it should be possible for our tool, which is already connected to the knowledge base, to show the user what the instrument is streaming. This would help the researcher to correct mistakes quickly.

6 Conclusions

The advent of automation in data collection poses many opportunities for revolutionizing data analysis in ecology. However, the large volume of datasets will be difficult to use without improved metadata collection strategies. As more diverse data is collected with the aim of integration and analysis, it becomes more critical to thoroughly and accurately capture in-situ information concerning dataset collection. Simultaneously, we do not want to overly burden field researchers with inefficient or error-prone collection methods. We present a mobile application for automating the collection of in-situ metadata in an efficient, standardized, and error-free way.

7 Acknowledgements

We would like to thank Jeremy Farrell and Matt Schuler for their time beta testing and reviewing the application from the point of view of a field scientist. We also thank Katie Chastain for adopting this project after the primary developer moves on. We would also like to thank the Jefferson Project Group at large for their ongoing support and collaboration.

References

1. Github deployment metadata app, https://github.com/lokijuhy/Deployment-Metadata-App
2. Mit app inventor, http://appinventor.mit.edu/ (Date last accessed: March 17, 2015).
3. Moccasn: A metadata collection app, http://tw.rpi.edu/web/project/JeffersonProjectAtLakeGeorge/MetadataApp
4. Sharing and packaging apps, http://appinventor.mit.edu/explore/ai2/share.html (Date last accessed: March 17, 2015)
5. Fegraus, E.H., Andelman, S., Jones, M.B., , Schildhauer, M.: Maximizing the value of ecological data with structured metadata: An introduction to ecological metadata language (eml) and principles for metadata creation. Bulletin of the Ecological Soc. of America 86(3), 158–168 (July 2005)
6. Jones, C., Blanchette, C., Brooke, M., Harris, J., Jones, M., Schildhauer, M.: A metadata-driven framework for generating field data entry interfaces in ecology. Ecological Informatics 2(3), 270–278 (2007)
7. Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Garijo, D., Soiland-Reyes, S., Zednik, S., Zhao, J.: Prov-o: The prov ontology. W3C Recommendation (April 2013), http://www.w3.org/TR/prov-o/ (Date last accessed: March 18, 2005).
8. Madin, J., Bowers, S., Schildhauer, M., Krivov, S., Pennington, D., Villa, F.: An ontology for describing and synthesizing ecological observation data. Ecological Informatics 2(3), 279–296 (Oct 2007)
9. McGuinness, D., Fox, P., Middleton, D., Garcia, J., Cinquni, L., West, P., Darnell, J., Benedict, J.: Solar-terrestrial ontology develop. In: AGU Fall Meeting Abstracts. vol. 1, p. 0324 (2005)
10. McGuinness, D.L., Pinheiro, P., Patton, E.W., Chastain, K.: In21b-3712 semantic escience for ecosystem understanding and monitoring: The jefferson project case study. In: Proceedings of AGU Fall Meeting 2014 (December 15-19 2014, Moscone Center, San Francisco, CA, US) (2014)
11. Michener, W.K.: Meta-inform. concepts for ecological data manage. Ecological Informatics 1(1), 3–7 (Jan 2006)
12. Michener, W.K., Brunt, J.W., Helly, J.J., Kirchner, T.B., Stafford, S.G.: Nongeospatial metadata for the ecological sciences. Ecological Applicat. 7(1), 330–342 (Feb 1997)
13. Pinheiro, P., McGuinness, D.L.: Provenance-enabled integration of sensor network data. In: AGU Fall Meeting 2014. San Francisco, CA, US (Dec 2014)
14. Reichman, O.J., Jones, M.B., Schildhauer, M.P.: Challenges and opportunities of open data in ecology. Sci. 331, 703–705 (2011)