The Gravity Data Ontology: Laying the Foundation for Workflow-Driven Ontologies

Ann Q. Gates[1], G. Randy Keller[2], Leonardo Salayandia[1], Paulo Pinheiro da Silva[1], and Flor Salcedo[3]

[1] University of Texas at El Paso, El Paso TX 79902, USA
[2] University of Oklahoma, Norman OK 73019, USA
[3] Rockwell Collins, Cedar Rapids, IA 52498, USA

Abstract. Ontologies can be tailored in ways that can facilitate the description of workflows by specifying how concepts representing services are used to access and create concepts representing data and products. Early work on the development of such ontologies, and reported in this paper, has resulted in the construction of a gravity data ontology. The relationships that are defined in the ontology capture inputs and outputs of methods, e.g., derived data and products, as well as other associations that are related to workflow computation. This paper presents the basis for a computation-driven ontology that evolved into the workflow-driven ontology approach. In addition, the paper describes the process used to construct an ontology for gravity data using the computation-driven approach, and it presents a gravity ontology that documents the processes and methods associated with gravity data and related products.

1 Introduction

Numerous institutions and organizations around the country have collected geospatial data, algorithms, and processes for manipulating and integrating these data with other diverse data sets, generating results that are useable by them, other scientists, or the general public. The goal of the work presented in this paper is to move from an environment in which a scientist relies on a professional network and manual processes to complete their work to one in which a scientist uses an automated system to accomplish tasks. An approach for realizing this goal is to capture knowledge through an ontology and then leverage the knowledge to support the design and execution of scientific workflows that compose software services to compute a particular result or generate a product.

There are several challenges that scientists face when creating any ontology: defining the scope of knowledge capture, determining the level of abstraction used to describe concepts and relationships, and identifying useful concepts and relationships. Clearly, creation of an ontology should be a continuing process that requires revision and refinement.

This paper presents an overview of a computation-driven ontology. The main contributions of the paper are to provide the rationale for establishing the key concepts in the computation-driven ontology and to document the process used to create a computation-driven ontology for gravity data. The paper also presents

an overview of the effort to develop tools that assist a scientist during the process of creating and validating an ontology and generating abstract workflows. These workflows denote how a result is achieved by presenting the composition of methods (software services or algorithms) including the flow of data and control among the methods.

2 Basis for Computation-Driven Ontologies

The basis for the concept of a computation-driven ontology was inspired by a February 2004 Seismology Ontology workshop held at Scripps Institution in San Diego. The attendees of the workshop included experts in the areas of seismology and information technology.

While the initial focus of the workshop was on creating a discipline-based ontology, i.e., an ontology focused on capturing knowledge about a particular discipline, it ended with a categorization and a set of relationships that were based on a general workflow that describes a common task performed by seismologists. After struggling with identifying the concepts that should be captured in a seismology ontology and motivated by a desire to identify concepts and relationships that would be useful to the community, the workshop participants defined concepts of interest by constructing the workflow shown in Figure 1. For the scientists, the workflow captured the steps for completing the task of creating a P-wave velocity model and the necessary concepts that are involved in completing such a task. After completing the workflow, the seismologists next partitioned the diagram into three categories: "Data," "Method," and "Product," where *Data* denotes input to or output from a *Method*, *Method* is a software service or algorithm, and a *Product* is an artifact. A summary of observations

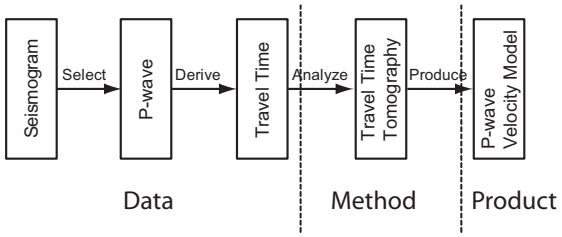

Fig. 1. A workflow created at the 2004 Seismology Ontology Workshop.

from the workshop includes the following:

1. *The benefits of using a workflow to drive creation of a specialized ontology*- If one considers how a desired product or result is generated, a discipline expert can identify the data, derivation algorithms, transformation algorithms, and other data processing algorithms involved as well as the relationships between them.

2. *The benefits of using a workflow to determine missing concepts or relationships*- It's important to note that the workflow given in Figure 1 is not complete. The step from *P-Wave* to *Travel Time* requires a transformation method that is not depicted in the diagram. The ability to view a workflow based on concepts captured in an ontology can assist in the iterative process of refining an ontology.
3. *The importance of using abstraction in the ontology-construction process*- Related to the second observation, this promotes the need to focus on a particular product or result at a high-level while neglecting other aspects. Moving from a high-level abstraction to detail allows one to manage the complexity in defining an ontology. For example, one can specify that *P-Wave derives Travel Time* and in subsequent iterations specify the method by which this is done.
4. *The importance of having ontologies that are created by scientists and for scientists*- While technology is critical for the development of cyberinfrastructure, the tools that scientists use to define and manage ontologies and workflows must be scientist-friendly and relevant to them.

3 Overview of the Computation-Driven Ontology

The observations that were made at the 2004 Seismology Workshop led to the definition of a specialized ontology called a *computation-driven ontology*, an ontology that encodes discipline-specific knowledge in the form of concepts and relationships supporting visualizations that depict how data is derived or results are obtained, e.g., in the form of a workflow. It is important to note that a computation-driven ontology casts concepts from a discipline-specific ontology into pre-defined concepts and relationships.

As a proof-of-concept, Salcedo and Keller [10] applied the approach to develop a gravity-data ontology. The top-level categories of the ontology are described as they apply to the gravity domain:

- *Data* define three types of concepts: (1) Field Observations, the purest form of gravity data; (2) Principal Facts, i.e., latitude, longitude, elevation and observed gravity values; and (3) Derived (Reduced) Data, i.e., values that are perceived and sought as data by the user community. All three types are values associated with a point.
- *Methods* are algorithms that are applied to the various forms of data to produce results that are interpretable from a geologic point of view. Results from methods yield derived data or products.
- *Products* are artifacts that result from application of a method. These artifacts are not perceived and sought as data by the user community. Examples include maps, models, or images.

Table 1 summarizes the main relationships that are defined for a computation-driven ontology. The table gives the inverse relationships and indicates whether

Table 1. A summary of relationships for a computation-driven ontology.

Tuple	Inverse	Trans.	Description
$[c1, isInputTo, c2]$	getsInputFrom	No	c1 is a Data or Product concept with raw numerical values; c1 is input into Method c2
$[c1, isOutputOf, c2]$	outputs	No	c1 is a Data or Product concept; c2 is a Method concept
$[c1, isDerivedFrom, c2]$	isConvertedTo	Yes	c1 is a Data or Product concept; c2 is a Data or Product concept; c1 has been created through a transformation of c2; c1's existence depends upon the existence of c2
$[c1, includes, c2]$	isIncludedIn	Yes	Method c1 includes Method c2 as a helper Method
$[c1, uses, c2]$	isUsedFor	Yes	c1 is a Method concept; c2 is a Data or Product concept; a Method uses a Product or Data when neither one is direct input into the Method

the relationship supports transitivity, i.e., if a is related to b and b is related to c, then a is related to c.

Consider the following statement: the adjusted gravity reading in milligals is derived from the raw gravity reading via the equation:

$$AGR = (RGR * CC) + DC + TC$$

where AGR is the adjusted gravity reading, RGR is raw gravity reading, CC is calibration constant for the gravity meter, DC is drift correction, and TC is tidal correction. From this text, we identify a method MAGR that computes AGR, and we identify the following relationships:

- $[RGR, isInputTo, MAGR]$
- $[CC, isInputTo, MAGR]$
- $[DC, isInputTo, MAGR]$
- $[TC, isInputTo, MAGR]$
- $[AGR, isOutputOf, MAGR]$

In the initial iteration of the ontology, one could state: $[AGR, isDerivedFrom, RGR]$, if the equation was not available or not considered because that level of detail was being abstracted. The next example shows the application of the include relationship, and makes an argument for incorporating it in a computation-driven ontology. Consider the text: Gridding methods include interpolation methods. This could be denoted as: $[M_{Grid}, includes, M_{Inter}]$. There are a number of interpolation algorithms that could be used with a gridding algorithm, and the includes relationship is used to capture this notion. To illustrate the uses relationship, consider the following statement: a Regional Gravity Map (RGM)

is used to determine whether to use a Directional Filter Method because the user must visualize the anomaly values to decide whether to use this filter. This denotes a manual process and should be considered when deriving a workflow description. The relationship would be expressed as: $[M_{Filter}, uses, RGM]$.

4 Constructing a Computation-Driven Ontology

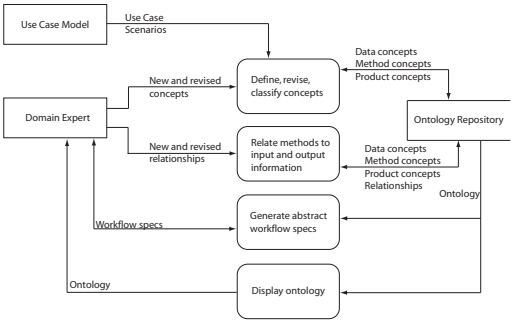

Fig. 2. Flow of information when constructing a computation-based ontology.

Ontology 101: A Guide to Creating Your First Ontology [6] presents guidelines for creating an ontology, which are applicable to a computation-driven ontology. In particular, use case modeling is an effective approach for driving the creation of any ontology.

The computation-driven approach places the primary focus on methods and data that generate results of interest to the scientist as well as on workflow-based relationships. Figure 2 presents a data flow diagram that depicts the processes or steps for defining a computation-driven ontology. The square in the diagram represents a source or sink, the rounded boxes depict transformation of information, and the open rectangle a store. As depicted in the figure, creation of an ontology is a continuing process, and it includes the use of an abstract workflow (as depicted in Figure 3). The processes are described next.

Identify concepts. Use cases allow one to scope the knowledge capture and identify useful concepts. In use-case modeling, the scientist identifies the primary uses of the ontology. Identifying use cases is complementary to developing workflows as an initial approach for specifying appropriate concepts. The discipline expert should consider the following questions: What types of data are available or can be derived? What existing algorithms, tools, or steps are used to generate data? What results are important to me or the community?

To illustrate the benefit of use cases, consider the following use cases in the gravity domain: "determine the Complete Bouguer Anomaly for points in a gravity data set," and "create a free-air anomaly map." Given the use case as

a starting point, the scientist would identify related algorithms for generating the desired data or product. For example, starting with the concept *Complete Bouguer Anomaly* and knowing that "Variations in Simple or Complete Bouguer Anomaly values are the major input into interpretations of the geological features present in the area of a geophysical study" would lead to the following concepts (types in parenthesis): *Simple Bouguer Anomaly* (*Derived Data*), *Complete Bouguer Anomaly* (*Derived Data*), *Interpretation Method* (*Method*). The following statement, "Calculation of the Complete Bouguer Anomaly uses the Free Air Correction value," leads to the following concepts: *Calculate Complete Bouguer Anomaly* (*Method*) and *Free Air Correction Value* (*Derived Data*). The following statement, "Observed Gravity Data is input to the Calculate Free Air Anomaly method where it has modifications performed on it and this produces a Free Air Anomaly," leads to the following concepts: *Observed Gravity Data* (*Processed Data*), *Calculate Free Air Anomaly* (*Method*), and *Free Air Anomaly* (*Processed Data*).

To elucidate the process of using a workflow to drive elicitation of concepts, consider that a discipline expert identifies *Anomaly Map* as an important result. Geospatial-mapping software, such as GMT (Generic Mapping Tools) [13] and denoted in the figure as *Mapping*, takes *Anomaly Values*, grids them, and contours them to generate an *Anomaly Map*. *Anomaly Values* are the result of raw gravity data reduction (e.g., [3]), which can be obtained through a series of steps programmed in Excel (e.g., [4]). In this example, *Anomaly Map* would be classified as *Product* and *Anomaly Values* would be classified as *Derived Data*. *Mapping* and *Excel Reduction* are classified as *Method*. Figure 3 presents two views for specifying this workflow. In the first depiction, methods are shown on the right side of the diagram, data and products are shown on the left. The relationships are marked above the arrows. In the second, the text in bold denotes the desired output. Questions regarding "how the output is generated" results in the specification of the next step. This continues until the base or initial concept is reached, i.e., *Raw Gravity Data*. The darkened arrows denote the outputs from methods and the text within parenthesis denote the inputs to the methods. Defining a simple workflow as shown in Figure 3 can be useful

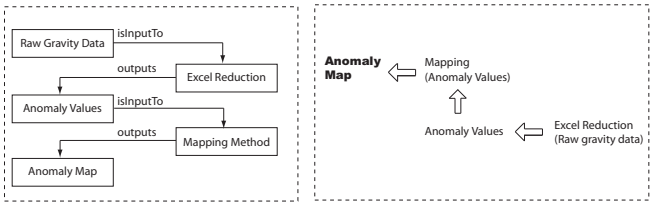

Fig. 3. Two views for illustrating the steps towards generating an abstract workflow specification for an Anomaly Map.

for defining concepts as well as refining concepts. For example, if the discipline expert had not included the *Excel Reduction* method and instead used the relationship *Anomaly Values isDerivedFrom RawGravityData* in the first diagram of Figure 3, then the expert would recognize that the ontology is underspecified; he or she would specify the method *Excel Reduction* during refinement.

Identify relationships. The discipline expert also identifies the relationships between concepts. All *Derived Data* and *Product* concepts should be associated with at least one *Method* class, and all *Method* classes should have input and output relationships.

The gravity data ontology is represented in the Ontology Web Language (OWL) [12], and the concepts described in this paper are referenced as classes in OWL. As a result, the class hierarchies are grounded in the OWL class *Thing*. During construction of the gravity data ontology, super class *Product* was divided into subclasses *Gravity map* and *Gravity model*, and subclasses *Anomaly Map* and *Contour Map* were defined under *Gravity Map*.

As described earlier, creation of an ontology is a continuing process that requires revision and refinement. For example, refinement of the ontology resulted in refining the *Interpretation* concept to include subclasses *Modeling* and *Mapmaking*. A similar refinement process occurred in which concepts *Complete Bouguer Anomaly* and *Free Air Anomaly* were classified as *Corrected Gravity Data*. Figure 4 shows a portion of the gravity data ontology that was created with

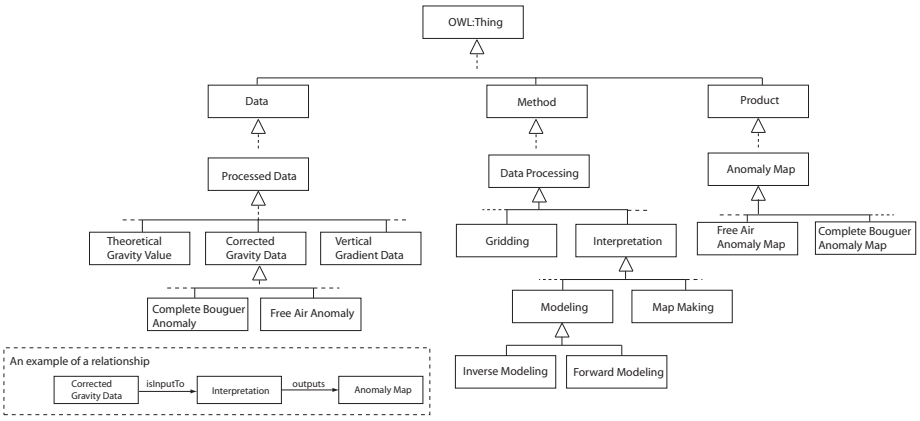

Fig. 4. A portion of the Gravity Data Computation-Driven Ontology.

experts in the field of geophysics using Protégé Ontology Editor and Knowledge-Base Framework tool, Version 3.1 Beta Full. Because of space constraints, the graphical depiction does not show relationships or annotations associated with each concept. See http://trust.utep.edu/ciminer/collaborations/ for documentation of the ontology.

5 Tool Development Efforts

The experience of creating a workflow-driven ontology for gravity data provided a number of insights. The scientist involved in defining the gravity data ontology found it more amenable to work on an Excel worksheet to initially store the concepts and relationships prior to specifying them in a formal ontology language such as OWL [12] and with the aid of an ontology editor such as Protégé. Moving toward a scientist-friendly approach to specification of ontologies has become a focus of the research. Indeed, the computation-driven ontology has evolved in the workflow-driven ontology (WDO) approach [8, 9].

The WDO-It! tool provides a graphical-user interface that is consistent with the concept classification requirements of workflow-driven ontologies and that can guide the scientist to elicit concepts and relationships from which abstract workflows can be generated. In addition, the WDO-It! tool provides workflow-generation functionality that allows the scientist to select a target data concept, and to generate graphical representations of abstract workflows that derive the selected data concept. The workflow generation functionality of WDO-It! is based on the Jena Ontology API [14] that supports inference engines that can interpret and reason about ontologies specified in OWL. The graphical representation of abstract workflows generated by WDO-It! serve as scientist-friendly devices that can be used towards the refinement and validation of the ontology, and that can be leveraged by scientists and technologists towards the development of executable workflow specifications. The authors are in the process of validating the usability of WDO-It!.

In addition, the capture of provenance information [7] provides the scientist with the ability to annotate data and method concepts with source metadata. For example, metadata regarding *Raw Gravity Data* could include information about the instrument used to collect the data, accuracy estimates, and the individual or entity that recorded the readings, while metadata regarding *Gravity Data Reduction Method* could include information about the specific implementation of the method and its constraints. As a result, once workflows are constructed from these data and method concepts, more complex data concepts or products could be automatically annotated with provenance information that includes the source data, methods, and workflow process used to generate them. Probe-It! is a prototype tool that provides the visualization of provenance data of an executing workflow. Assuming that the executable workflow is composed from provenance-annotated concepts, Probe-It! traces the provenance and constructs provenance proofs on the fly as a workflow is executed.

6 Related Work

There are numerous published ontologies. This section summarizes three: the Gene Ontology (GO), the Transparent Access to Multiple Biological Information Sources (TAMBIS), and the Semantic Web for Earth and Environmental Terminology (SWEET) ontologies.

GO [11] provides a controlled vocabulary to capture gene information. In the GO ontology, a function describes methods, and the process ontology describes a series of steps similar to a workflow. TAMBIS [2] is a bioinformatics ontology whose design is based on description logics in order to allow dynamic creation and reasoning about the concepts. TAMBIS is organized into multi layer divisions. For example, a structure can be separated into its physical and abstract representations. The ontologyalso has separate concept divisions for biological processes and biological functions. Similar to TAMBIS, the computation-driven ontology approach adopts the separation of concerns with respect to concepts.

The SWEET ontologies [15] were developed to capture knowledge about Earth System science. There are two main types of ontologies in SWEET: facet and unifier ontologies. Facet ontologies deal with a particular area of Earth System science, e.g., earth realm, non-living substances, living substances, physical processes; unifier ontologies were created to piece together and create relationships that exist among the facet ontologies.

7 Summary

The computation-driven ontology was devised to support scientists ability to capture discipline-specific knowledge that supports their research. Such an ontology focuses on the capture of processes as well as data and reduces the dependence on a technologist to construct an ontology. Computation-driven ontologies are distinguished from discipline-based ontologies that capture basic knowledge about a discipline by capturing concepts and relationships that are tied to how results are generated. In particular, all defined methods are tied to the inputs, outputs, and other computation-associated relationships required to generate a result from a specified method. The gravity data ontology is the first comprehensive ontology that was developed using this approach.

The work reported in this paper has transitioned to the development of a prototype WDO API [8] to facilitate the integration and reuse of WDOs by the WDO-It! tool and other WDO-related tools that are being prototyped. The WDO API is built on top of the Jena2 Ontology API [14] that provides functionality to access OWL ontologies through Java programming. The WDO API offers specific methods that facilitate the development of WDOs, as well as functionality to create abstract workflow specifications. The WDO-It! tool provides a GUI to assist scientists to create new WDOs. Work is in progress to extend domain ontologies into WDOs and to transform abstract workflows to executable workflows. Future work will examine the use of WSDL-S [1]and OWL-S [5] to refine abstract workflows into executable workflows implemented as web service compositions. Both WSDL-S and OWL-S are specifications targeted specifically to enhance web service technology with semantic information.

Acknowledgements. The work described in this paper was partially funded by the NSF GEON project EAR-0225670 and the NSF CREST project HRD-0734825.

References

1. Akkiraju, R., et al.: Web Service Semantics - WSDL-S. World Wide Web Consortium (W3C) recommendation, http://www.w3.org/Submission.WSDL-S, November (2005)
2. Baker, P.G., Goble, C.A., Bechhofer, S., Paton, N.W., Stevens R., Brass, A.: An Ontology for Bioinformatics Applications. Bioinformatics, 15(6) (1999) 510–520
3. Hinze, W. J., Aiken, C., Brozena, J., Coakley, B., Dater, D., Flanagan, G., Forsberg, R., Hildenbrand, T., Keller, G. R., Kellogg, J., Kucks, R., Li, X., Mainville, A., Morin, R., Pilkington, M., Plouff, D., Ravat, D., Roman, D., Urrutia-Fucugauchi, J., Vronneau, M., Webring, M., Winester D.: New standards for reducing gravity data: The North American gravity database. GEOPHYSICS, 70: 325-332 (2005)
4. Holom, D. I., Oldow, J. S.: Gravity reduction spreadsheet to calculate the Bouguer anomaly using standardized methods and constants. Geosphere, 3(2):8690; doi: 10.1130/GES00060.1 (2007)
5. Martin, D., et al.: OWL-S: Semantic Markup of Web Services. World Wide Web Consortium (W3C) recommendation, http://www.w3.org/Submission/OWL-S/, November (2004)
6. Noy, N. F.,McGuinness, D.: Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05, March (2001)
7. Pinheiro da Silva, P. et al.: Knowledge Provenance Infrastructure. IEEE Data Engineering Bulletin, 26(4), December (2003) 26–32
8. Salayandia, L., Pinheiro da Silva, P., Gates, A., Salcedo F.: Workflow-Driven Ontologies: An Earth Sciences Case Study. in Proceedings e-Science 2006, Amsterdam, Netherlands, December (2006)
9. Salayandia, L., Pinheiro da Silva, P., Gates, A., Rebellon, A.: Domain-Level Workflows for Scientific Applications. in Proceedings 6th OOPSLA Workshop on Domain-Specific Modeling, October (2006)
10. Salcedo, F.: A Method for Designing Computation-Driven Ontologies in the Geosciences. Master's Thesis, University of Texas at El Paso, May (2006)
11. Smith, B., Williams, J., Schulze-Kremer, S.: The Ontology of the Gene Ontology. in Proceedings AMIA Symp. 2003, (2003) 609–613
12. Smith, M. K., Welty C., McGuiness, D. L.: OWL Web Ontology Language Guide. World Wide Web Consortium (W3C) recommendation, http://www.w3.org/TR/owl-guide/ February (2004)
13. Wessel, P., Smith, W. H. F.: New version of Generic Mapping Tools released. EOS, Transactions American Geophysical Union, 76:329 (1995)
14. Jena: Jena2 Ontology API. http://jena.sourceforge.net/ontology/index.html, July (2006)
15. SWEET: Guide to SWEET Ontologies. http://sweet.jpl.nasa.gov/guide.doc, June (2007)