Conceptualizing Representational Semantics: A Multiple Layered Spatial Data Integration Framework Based on Ontology

Chen-Chieh Feng[1] and Liang Yu[1],
[1] Department of Geography, National University of Singapore
1 Arts Link, Singapore 117570
{geofcc, geoly}a@nus.edu.sg

Abstract

Domain ontologies have been used as an effective mean to reconcile the heterogeneities between different spatial data sources. Most domain ontologies focus on the concepts from real world, but do not specify the semantics of the way to represent them. However, heterogeneities exist in forms of different frameworks (standards), data models (schemas), measurement backgrounds, algorithms, spatial-temporal features, etc, which makes seamless data integration a non-trivial work. Even if we have solved the problem of identifying domain concepts from different sources, we still have to deal with fusing the data associated with the same concept but from different representations. Representational semantics are about the concepts which give the semantic of how we measure and organize the result of observation of concepts from real world, e.g. property and relation, unit, coordinate system and process model. They are, while always implicit in all kinds of data and systems, very important for engineers responsible for the data management and integration, and users who want to evaluate the viability of the data for a specific task, for their indispensability for supporting semantic alignment, data structure conversions, and mathematical computations, all of which are crucial steps toward a successful integration of spatial databases. In this paper we argued that both domain ontologies and their conceptual representations are essential to spatial data integration and assuming complimentary roles. We proposed to capture and formalize representational semantics for the spatial data, together with domain ontologies, to facilitate semantic-dependent data integration with heterogeneous data sources and processing models. By separating the two-level ontologies, more restrictions and axioms can be added to them, the enhanced reasoning ability can improve the seamless spatial data integration. We studied several cases to show how to leverage representational ontologies to solve the data integration problems.

Keywords: Representational ontology, reasoning, multiple-layer

1. INTRODUCTION AND MOTIVATION

In the past decade the scientific community have seen significant improvement in data collection devices and methodologies. A wide variety of sensors, from stationary to mobile, from dependent to autonomous, and from wired to wireless, have been used in scientific research to collect data at different spatial and temporal resolutions. Integrating of the data generated by these sensors intelligently and effectively have been challenging due to four types of heterogeneities (PaoloBouquet et al., 2004):

1. Syntactic heterogeneity, caused different representation languages being used.
2. Terminology heterogeneity, caused by different terms, such as river and stream, being used to refer to the same entity.
3. Conceptual heterogeneity or semantic heterogeneity, caused by different ways that an entity is perceived. For example, a building may be perceived as a container that bounds a three-dimensional space one the first person but a land feature with a two-dimensional footprint by a second person.
4. Semiotic/Pragmatic heterogeneity, caused by different contexts being imposed on the intended use of an entity. For instance, two ontologies containing concepts land use classifications may be considered to be the same. However, the meanings of a concept in one ontology may be different from that in a second ontology if the first ontology is for environmental conservation while the second ontology is for rural area planning.

The heterogeneities 1 - 3 were identified as the main obstacles for data integration. Many solutions have been introduced to handle these heterogeneities with the exception to the semiotic heterogeneity where human intervention is still very much needed. Among these solutions for 1 - 3, ontology has been used to formalize semantics and to eliminate the heterogeneities existing in different sources such as applications and data sources. In GIS domain, ontology-based method has been proved an effective way to solve the terminology heterogeneity and part of the conceptual heterogeneity (PaoloBouquet et al., 2004), and then assist to create correspondence between sources. Domain ontology is developed by the collaboration between domain and ontology experts, representing the formal understanding of the domain in terms of concepts, relations, and axioms. Ontology matching is used to reconcile the second and portion of the third heterogeneities between overlapping domain ontologies. The conceptual heterogeneity is always the key problem to solve because it contains many aspects as coverage, granularity, and perspective. As ontology has been used as the knowledge basis for real-world things, the problem of integrating different sources of information is shifted to integrating different ontologies corresponding to these sources or domains (Buccella et al., 2009).

Despite these efforts to solve the terminological and conceptual heterogeneities, the integration and eventually the sharing of data among different users require a better means to handle syntactic and semiotic heterogeneities, which are handled by some manual means or specific processes that are not adaptable and scalable. We argue that there exists a kind of semantics that can help recognize the methods we

use to represent the concepts in an information system, and help us to find a way to reconcile these heterogeneities. Our rationales are based on the following:

1. Both ontology and conceptual model (schema) are indispensible components of an information system. Ontology deals with general assumptions concerning the explanatory invariants across domains, while conceptual model involves the specification of these invariants for a particular domain (Fonseca and Martin, 2007). Thus, being able to match concepts across ontologies does not necessarily mean being able to integrate data of these concepts. For example, there are many ways to represent the concept location associated with the concept building, e.g. street number, coordinates, or using relations to other geographic entities. Being able to match buildings does not mean being able to match their locations.
2. The goal of data integration is to convert the real data from one form to another form that can be used directly by the users. This process always involves computation on top of one-to-one matching. For example, the integration of two data sets of water temperature might involves unit conversion, spatial reference conversion (e.g. between different geo-reference systems), and model conversion (e.g. vector and raster).
3. The data is a representation of concepts from real world. Thus, to precisely understand the nature and the intended usage of the data, the encoding of the context under which it is collected is vital. For example, one researcher might record the (x, y) coordinates together with the water temperature value while another one records (x, y, z), where "z" stands for the depth of the measuring point, for that the data is used for a 3D fluid analysis.
4. An ontology is often based on the open world assumption whereas a data model is often based on the close world assumption. The use of an ontology in data integration thus requires bridging the differences between the two assumptions.
5. Not all data associated with the same ontological concept can be integrated, depending on the target representation, which is significantly affected by the intended usage. The restrictions imposed on concepts can be more concrete than semantics expressed by domain ontology.

For the same ontological concept, there are various ways and contexts of representations for it, which is related to data models, schemas, measurement backgrounds, algorithms, spatial-temporal features, and so on. They are significant for the users to evaluate the suitability of a dataset and the method to convert it to the required form. In this paper, we defined the representational semantics as a range of ideas to represent the concepts from real world, consisting of features like model, context, reference, etc. We proposed to use the representational semantics to capture the ways and the context that we used to describe datasets. It can help both of the humans and computer systems to recognize the intended purpose of the data creator and the possible ways to use them, and to automate the process of the data and the overall workflow. The proposed approach adds a representational layer to a

multiple-layered ontology driven system, where the ontologies in the layer are linked to relevant domain ontologies and are capable of performing tasks with different levels of semantics. We pay more attention to the spatial data since most of the data in our research is spatial related, which needs us to deal with spatial related concepts. The representational ontology proposed in this paper is an example of how the proposed framework is applied. It is not exhaustive but offers an example how the framework can be improved by practitioners from various domains.

The paper is organized as follows. Section 2 introduces related work and then elicits the approach we will use. Sections 3 proposes the framework of our representational ontology for different aspects of IS. Section 4 discusses the axioms and restrictions we can add to the ontology to enhance the ability of query/search. Section 5 discusses the semantic information implicit in existing standard and possible way to extract them. At last we conclude the contents and remarks and the future work are presented.

2. RELATED WORK

Ontology has been one important component for facilitating data integration. Various aspects of using ontologies for data integration had been explored and studied. Torres et al. (2007), for example, examined two fundamental issues in geospatial ontology design – which geospatial entities should be collected and how they should be organized – and developed formal ontology for describing essential entities of geographical domain. Their framework proposed to use a set of formalized terms to describe a geographic context and encoded them as concepts in ontology.

Wache et al. (2001) discussed applications of ontology for data integration. Three approaches, i.e., single ontology, multiple ontology, and hybrid ontology, were identified. The single ontology approach is based on a set of common vocabularies across different applications. It is suitable for a tight-coupled environment such as inside one company. The multiple ontology approach permits separate development of individual ontologies. Matching methods are then employed to reconcile differences between every pair of them. Last, the hybrid ontology adds a common ontology to link all shared ontologies. It achieves a balance between confining all domain ontologies under the same knowledge framework and allowing flexibility for individual domains to encode their own knowledge bases. Similar to the hybrid approach, Ludacher et al. (2001) proposed to use multiple-tier mediation for different types of data sources such as database and XML file, and then integrate data from them. Their solution used a conceptual model wrapper layer (CM-Wrapper) and a generic conceptual model (GCM) layer between the integrated view and the real database, instead of directly using data structure. The direct data access is encapsulated by the wrapper layer, and different CM-Wrappers expressed in XML syntax are mapped to GCMs. In turn, these GCMs are mapped to an integrated view.

The benefit of multi-layer ontology, such as adaptability and higher degree of interoperability, was also explored. Beran et al. (2009), for example, developed a multiple-layered ontology for a water data system. Their ontology layers are classified by the scope of the semantics of the concepts. From narrower to broader, four layers of ontology were proposed: navigation, compound, core, and detail. The four layers are in the same domains, while providing multiple choices for users to specify or generalize their query criteria. Villa (2007) proposed a framework that unifies diverse representations of simulation model applications. The framework distinguishes the ontological characters and the observation contexts of the objects in simulation models. Modelers can thus focus on with the conceptual objects of a simulation domain, leaving complexities connected to the data formats and scale to the framework. Benslimane (2000) proposed a method for classifying different layers of ontologies based on dependency, which reflects the granularities of their concepts.

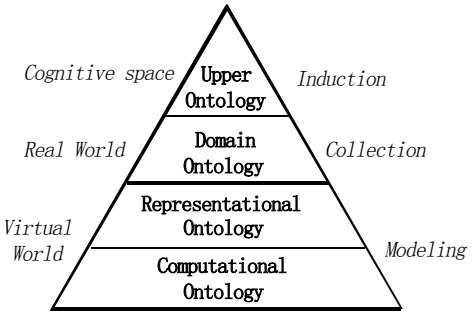
Calvanese (2002) classified the ontology matching and integration methodologies as two categories: Global as View (GAV) and Local as View (LAV), which both has a single ontology layer directly mapped to resources. Based on this, Fagin (2003) proposed a global-local-as-view (GLAV) methodology, which is a 3-layer architecture, the global ontology and local ontology are distributed in two layers, and the latter is matched to resource layer. This methodology takes the advantages of the efficiency of GAV and flexibility of LAV.

How ontologies can facilitate mappings of the entities at the same layer was also explored. Quix et al. (2006), for example, explored how ontology can be used to improve schema mappings efficiency in geographical information systems. They showed how ontologies are useful for detecting incorrect mappings thereby reducing the overhead of verifying the schema matching results. Last, approaches to bridge ontologies at two adjacent layers were explored. Fonseca et al. (2003) proposed a way to link ontologies and conceptual schemas in geographical information system. He proposed a way to link the formal representation of semantics to conceptual schemas describing information stored in databases, and the result is a mapping between terms and relations of computational ontology and geographic conceptual schema. Further, he investigated the essential roles of computational ontology and conceptual model (Fonseca and Martin, 2007), argued that they originates from different level of conceptualization and act as different roles in information systems.

The existing research efforts showed a general acceptance of multi-layered ontology framework for data integration. They also indicated the importance to differentiate ontological concepts and their representations for any information system supporting data integration. Our work recognizes these two points by introducing a four-layer ontology framework that has four layers (Figure 1). The upper, domain, and computational ontology are used in most existing applications. The upper ontology layer contains concepts related to how the things in the real world are conceptualized and are collected by induction. The most popular upper ontologies are SUMO, BFO,

and DOLCE, among which SUMO is more comprehensive that it can be applied to many different domains. Domain ontology contains concepts of specific domains, or part of the world, which represents the particular meanings of terms as they apply to that domain. The concepts in this layer are collected from domain practitioners. In the realms of medical, biology, earth science, there are many famous ontologies such as Gene ontology, Unified Medical Language System, USGS Geology Time Scale Ontology, and SWEET. Computational ontology is an ontology or domain specification which has been automatically derived by computer software using modeling, data mining, textual analysis, and statistical techniques. In other words, computational ontology is a result of data analysis and modeling, e.g. a UML data model which can be converted to database schema, or a result of a data mining process, which can be formalized as axioms for ontology and converted to restrictions for database design.

Figure 1: Multiple-layer Ontology Model



The framework differentiates itself from others with the addition of the representational ontology layer between the domain ontology and computational ontology layer. It contains concepts used in modeling the domain concepts. For example, digital elevation model and benchmark height are two concepts in the representational ontology that are linked to the elevation domain concept. Our approach put the effort on the conceptualizing of representational ontology, and linking them together with the ontologies from the domain ontology layer and computational ontology layer to provide a scalable data integration. The methodology to develop this layer includes the following three steps:

1. By investigating the semantics commitment of the way the spatial data is represented, extract the representational concepts and form the ontology. Separating the concepts into domain concepts and representational concepts is the main principle of our ontology design.
2. Develop restrictions and axioms for the concepts. Restrictions and axioms applied to representational ontology can help us validate the data representation, and infer new relations based on the existing one, which could significantly alleviate our effort on deciding whether and how two datasets can be integrated.

3. Use this ontology to analyze practical data application issues in a distributed environment, including data integration workflow, data integration process, provenance analysis.

3. REPRESENTATIONAL ONTOLOGY FOR SPATIAL DATA SHARING

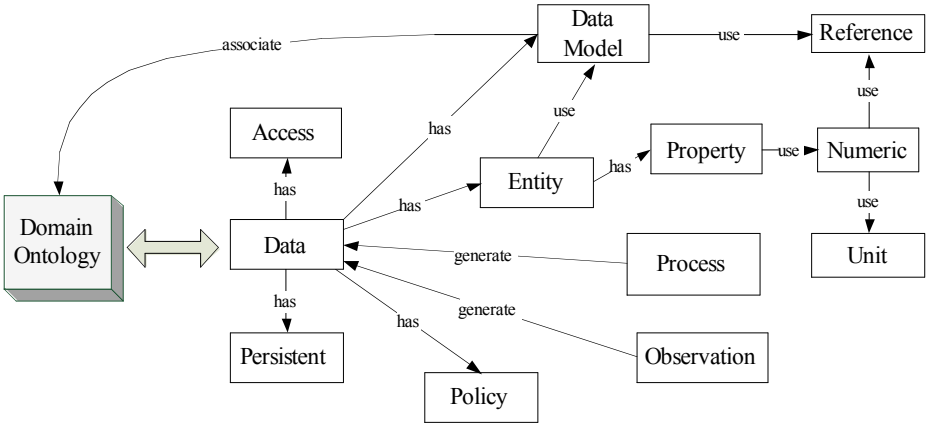
We take the Data as the core concept of our ontology because it is the most fundamental concept for accessing data in an information system. For accessing the needed data, the information system typically needs to answer the following questions:

1. What are the data about?
2. How were the data created?
3. Under what context were the data created?
4. How can the data be accessed?

The first question is a traditional question in most ontology-driven applications, usually solved by domain ontology. Question two concerns a broader issue, which includes how we map the concepts from domain ontology and create the computational ontology or schema. Question three concerns the environment and methodologies in which a measurement is taken. It is important for evaluating the results generated by the data. Question four is related to the interface with which the data is accessed and the protocols/policies that have to be checked before the data are accessible.

In order to answer these questions, we propose a framework of 12 main categories, shown in rectangles in Figure 2. The concept *Data* and its related concepts are encompassed into an individual domain, which is separated from but has a connection with the traditional domain ontology.

Figure 2: Overall View of Representational Ontology



The specified meanings of those categories are as follows:

1. **Data**. The term *Data* means a set of information that represent the qualitative or quantitative attributes of a variable or set of variables.
2. **Observation**. The observation refers to the data collected or measured. In scientific research efforts, the data is usually collected by observation or measurement and then processed by models, of which the later is classified as *Process*. The separation of these two types of data is important for determining the suitability of the data.
3. **Entity**. The entity describes the details of the content of the dataset, including the entity type, the attributes, and the ranges of the attributes(FGDC, 1998). Since different applications have their own implementations for entity description, we define the types of entities as concepts in domain ontology in our research.
4. **Numeric/Unit/Reference**. Numeric is associated with qualitative attributes such as interval and point (JPL). Numeric concept is different from the usual primitive numeric data type because it is always associated with *Unit* and *Reference*, the measurement context of an observation. Unit is associated with a numeric value to indicate a measurable property; Reference is always referred as a coordinate system in geographical related areas.
5. **Property**. Property specifies the relation between two objects. It can be specialized into different property subtypes such as identity, physical property, and spatial property.
6. **Data model**. Data model describes the way data is organized. There are three types of data models in a general computational modeling realm: Conceptual data model, logical data model and physical data model. In most applications, physical data model is not an issue of concern since it is isolated by data management systems such as database. In our research, we focus on spatial data model, which is actually sub-category of both conceptual and logical data model.
7. **Access**. The concept refers to the way by which we can access the data. It is dependent on the *Persistence* and *Policy* concepts. For instance, normally a dataset stored in database will support an OCBC access interface, while a data file in a file system with Windows can be shared through the NETBIOS protocol.
8. **Process**. Process is another activity besides observation which generates new datasets. Data are used in various processes, such as refinement, simulation, modeling, etc, each of which can generate new data, or add new properties to the existing data. A process takes an *input* and generates an *output* with a specific algorithmic module. Process is also an essential element for workflow system. By adding restrictions and rules, single process models can be combined as a process chain. Concepts related to process are also important for provenance information which basically consists of the information of lineage between datasets and processes that generated these datasets. Provenance is important for scientific researchers who want to evaluate the suitability of a dataset before use it.

9. **Policy**. Policy is an important concept for SDI (Spatial Data Infrastructure) (Hjelmager et al., 2008), which includes standards, best practice, pricing, intellectual property and accessibility. In our practice, Policy is used to control the accessibility of data, which can vary in different systems. Thus, we take out the basic concepts related to it and SDIs can implement them with different details. .
10. **Persistence**. This is about how the data are persisted, including format, media, location, etc. Persistence occurs throughout the scientific research.

Figure 3: Concepts of Data, Observation, Process, Entity, Property and DataModel

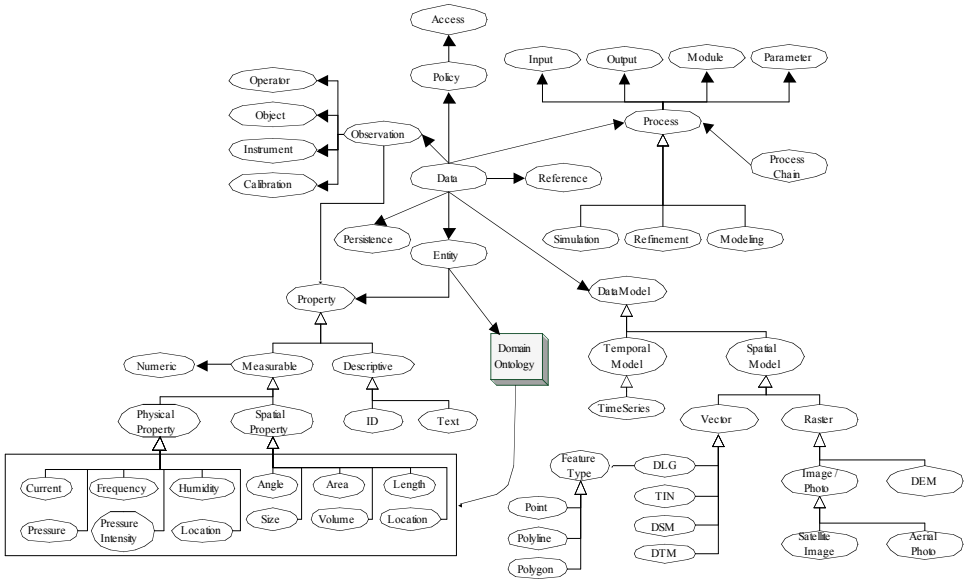


Figure 3 is a detailed diagram including the main sub-concepts of *Data*, *Process*, *Observation*, *Entity*, *Property* and *Data Model*. Based on this framework, more sub-concepts can be customized by users. For example, *Temperature Data* can be captured as a sub-concept of *Data* with the definition “Temperature Data is the data which contains one or more *Temperature* entities”. Data has direct relations with six concepts as *Reference*, *Data Model*, *Observation*, *Process*, *Policy*, and *Entity*. One thing worthy of mention is the relation between this ontology and domain ontology. Ontology was used to annotate data and metadata in many applications, e.g. keywords of a metadata document. However, simple annotation might cause ambiguities because the relation between the annotated entity and the ontology is not clear. For example, a dataset annotated with *Sensor* might contain the data of sensor, e.g. hardware components, function, calibration, etc., or data generated by sensor. With the representational ontology, it is clear what an ontological concept really means to a dataset. Annotation *Sensor* related to concept *Observation* means that the dataset is generated by sensor, while to Entity means its content is about the sensor.

Figure 4: Concepts of Category Numeric

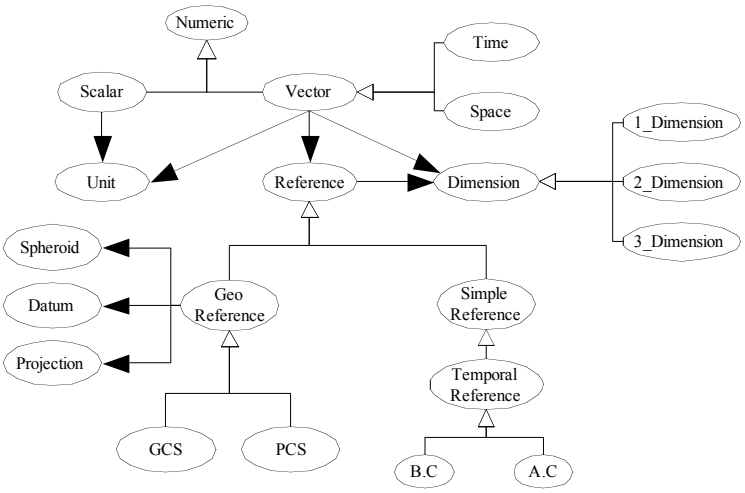


Figure 4 is a detailed diagram of the numeric concepts. Numeric is about the semantics of the quantitative data. The top level branches of the numeric concept are *Scalar* and *Vector*. A scalar value has no orientation factors so it needs no reference information, e.g., quantity, length, temperature, humidity, etc. Vector value is always associated with reference system to make it meaningful, e.g. space and time. Spatial reference system is more complicated and is associated with spheroid, datum and projection. However, there are many vectors which are often treated as scalars, e.g. velocity, strength, but actually are vectors and associated with orientation information in many simulation models.

Figure 5: Concepts of Category Persistence

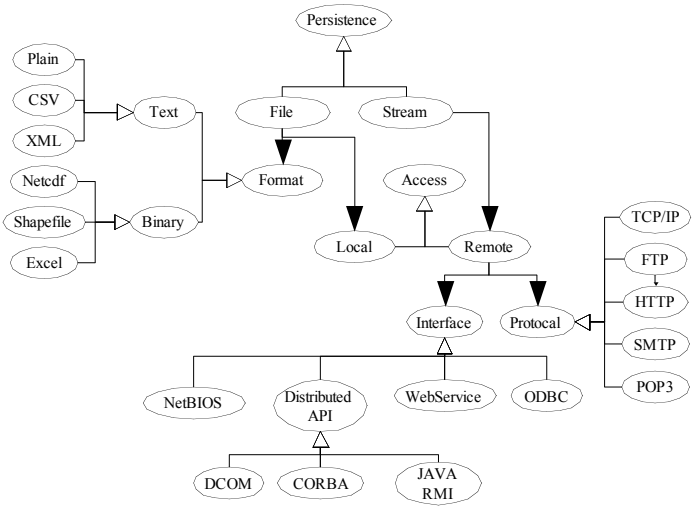


Figure 5 shows another portion of ontology of the Persistence category. Its related concepts are *Format*, *Access*, *Interface* and *Protocol*. These concepts can be used to access and parse the data. Restrictions can be applied to the relations between them. T-Box restriction will be elaborated in the next chapter as one of the most important ones. For example, a data file is always associated with local location or a file sharing

interface as NetBIOS, or HTTP/FTP protocol, while a data stream is always associated with a remote access, such as a web service, DAPI, or ODBC connection.

The concepts listed in the figures are not exhaustive as we can extend them by introducing more concepts under this framework. Datatype properties are not shown in the figures. Examples of properties for the concept *Data* are name, size, created date, last edited date.

4. REASONING WITH REPRESENTATIONAL SEMANTICS

Reasoning is an important feature of ontology technology. It can be used for various purposes, such as validation or inference. For example, if we want to evaluate a dataset, usually we need to trace all the processes and data sources related to it and analyze them one by one, which could be intricate work considering the complexity of various forms of input files. Ontology reasoning can alleviate this process by making the output dataset inherit properties from related processes and datasets, and infer new properties for the new dataset. For example, a *precsion* property in the input dataset can be the maximum precision for all dataset generated from it.

In our research, when representational semantics is extracted from the metadata and then reorganized, sometimes there are some erroneous, inconsistent, or incomplete information, where reasoning can be used for validation. Rules for validation are mainly defined by T-box definition. T-box is a term from descriptive logic terminology, which describes a system in terms of controlled vocabularies. For example, *Data* has relations to other concepts as Figure 3 shows, some of which are compulsory and the others are optional. This kind of restrictions can be encoded as T-box. T-box restricts the ontology in two aspects:

1. Cardinality. Cardinality indicates a specific number of values for a required property. For example, a *Parent* is defined as who has at least one *Child*. In our research, for example, all datasets should have a *generatedBy* property and the range of it could be instances of either *Observation* or *Process*; a geographical dataset must have a geo-reference definition; a DLG (Digital Line Map) dataset must have a feature type specification.
2. Property Range. It is to constrain the range of a property to a specified scope. It is used when a property is applied to a more concrete concept than what it was originally defined. For example, a vector has a *has_dimension* property, while its sub-concept Time has a value of *1_dimension*, and *Space* has a value of *2_dimension* or *3_dimension*.

Table 1. Examples of T-Box Axioms

T-Box Definition
1. Data⊆„has-source.□□*sour* Observation∪,has-source.□□□ *P* Process.

2. Vector⊆,Numeric∩,has-reference.,□□□□□□□-1..
3. Space⊆,Vector∩,has-reference.□□□□ GeoReference.
4. Process⊆„has-input.□□□ 1∩,has-output.,□□□-1.∩,has-processor.,min-1..
5. GeographicalData≡,Data ∩ ,has-model.□□□□ SpatialDataModel.
6. ElevationData≡{Data ∩(,has-model.□□□□ DEM∪,has-model.□□□□ DTM∪,has-model.□□□□ DSM ∪,has-model.□□□□ Contour)}

Table 1 lists examples of T-box definitions. Definitions 1-4 are necessary conditions usually used for validation. Definition 1 requires all data to have a source either by an observation activity or a process; definition 2 requires each numeric entity to have a unit definition; definition 3 requires a geo-reference system to be associated with a space vector; definition 4 requires a process to have at least an input, an output, and a processor. Definitions 5-6 are equivalent conditions used to populate new concepts.

Validation begins when the new data are imported and their semantics is extracted. This process helps both the data manager and provider to recognize the possible errors and inconsistencies of encoding of the data specification.

Another application of reasoning is inference, which is to infer new properties or relations based on the existing ones. For example, a user might want to compare if two datasets are compatible by defining the comparing algorithm using their properties, or to find all the datasets used to generate these datasets. While relations between datasets are not explicitly specified when they were imported, users need to develop algorithms to recursively analyze the direct properties or relations. The efforts can be relieved by using rules to assign new properties and relations to datasets, i.e., adding relations between datasets to indicate if they are compatible or have a parent-child relation, and run the reasoning for all the datasets. This work is often assisted by two methods: inverse/transitive property, and rule-based inference.

Inverse property in ontology technology is defined as a pair of properties which have the inverse subject and object. For example, *has_parent* and *has_child* is a pair of inverse properties. Given dataset A, B, if A *has_parent* B, then B *has_child* A.

1. Transitive property is defined as whenever an entity *a* has relation *r* with *b*, and *b* in turn has the same relation r with c, then a has relation *r* with *c*. A typical case for transitive property is “contain/include”, and its inverse relation “in/part-of”. For example, the located-in relation between spatial features is a transitive relation. The parent/child relations between datasets are also transitive properties.
2. Rule based-inference is more flexible in describing more complicated reasoning logic, e.g. properties inheritance as aforementioned. Both inverse and transitive property can be regarded as particular form of rule expression.

Table 2. Examples of Rules-based Axioms. The expression r(x,y) means that binary relation r has the subject x and the object y. The "ist(x,y)" means that x is an instance of y. "sub(x,y)" means x is a subclass of y. "sup(x,y)" means x is a super class of y.

	Rule
1.	annotated,?x,?c1.∩annotated,?x, ?c2.∩(sub(?c1,?c2)∪sup(?c1,?c2)∪eql(?c1,?c2))→has_compatible_concept(?x, ?y)
2.	ist,?p,Process.∩has_input(?p,?x)∩has_output(?p,?y)→has_parent(?x,?y)
3.	ist(?x,TemperatureUnit)∩ist(?y,TemperatureUnit)→convertible_unit(?x,?y)
4.	ist(GeoReference,?x)∩ist(GeoReference,?y)→convertible_reference(?x,?y)
5.	ist(?x,Contour)∩ist(?y,DEM)→convertible_model(?x,?y) ist(?x,DEM)∩ist(?y,TIN)→convertible_model(?x,?y) ist(?x,DTM)∩ist(?y,DEM)→convertible_model(?x,?y) ist,?x,DLG.∩ist,?y,DLG.∩has_feature_type(?x,?f)∩has_feature_type(?y,?f)→convertible_model(?x,?y)

Table 2 lists a portion of rules for inferring properties. Rule 1 is used to decide if two datasets are annotated with the same domain concepts, which indicates if they are compatible or not. Rule 2 indicates the input and output datasets in the same process have the parent/child relation. Rule 3 indicates that units under the same category are compatible and amenable to conversion. Rule 4 indicates that geo-reference systems are convertible to each other. Thus, a dataset with a local coordinate system cannot be directly converted to geographic data before it is geo-referenced. Rule 5 indicates four pairs of model which are considered as compatible.

5. STUDY CASES

In this section we demonstrate the usefulness of the representational semantics in solving practical problems in a sensor-network based environment monitoring system. CyberInfrastructure (CI), which aims at providing a one-stop data accessing service, was designed with a background of SDI with the support of ontology technology. Data providers register their datasets to CI, which can be encoded with different languages and standards, e.g. XML and CSDGM (Content Standard for Digital Geospatial Metadata) standard. Both domain and representational semantics contained in the metadata is extracted and reorganized. The following are typical user cases in the process of data exchange:

1. Access and manage the data sharing workflow in a Spatial Data Infrastructure (SDI)

There are basically three roles of participants: data provider, system manager, and data user. Different roles can be assigned to the same user, i.e. one can be both data provider and user. Ontology is adopted to reconcile the heterogeneity caused by their different backgrounds. Data providers annotate their data with concepts from

ontology, or leave it to system manager and some semi/automatically process programs. When a user starts the query process, the first step is to query the domain concepts which he/she is interested in, and then, he/she might want to integrate the datasets having the same semantics, or query data by its properties. The original request might be translated into different forms corresponding to various data sources. With the representational semantics of the data, the integration program knows if the target data sources can be integrated, and how to do it.

Figure 6: Data Access from a Data User's Point of View

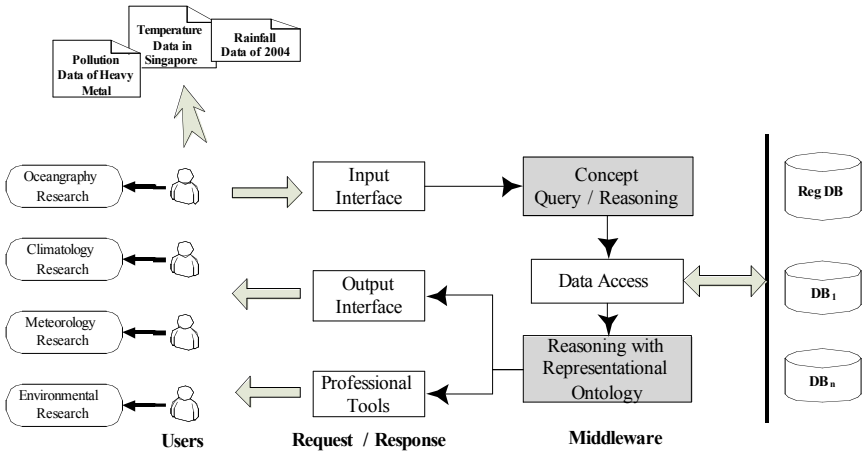


Figure 6 shows the process for data search/query/integration from the data users' point of view. The two dark boxes represent the two phases where ontology participates, where the first one is assisted by domain ontology and the second one by representational ontology. Data access is also assisted by semantics such as *Access* and *Policy*.

2. How to integrate multiple datasets?

This process consists of four steps: First, do these two datasets have the same domain concepts? For example, do they both contain temperature or humidity data? Then, the second step is to decide if those datasets are amendable to integration. This is because not all datasets with the same domain semantics can be integrated. For example, road data can be represented by polygon or polyline. Though both of them can be displayed in the same geographical map, they cannot be integrated as a new dataset for a specified application, e.g. calculation of the transportation area of a city. This process can benefit from the representational semantics to decide if they have the same or compatible representations, e.g. data model, entity/property definition, spatial data type, spatial reference system, etc. With reasoning ability, users can easily get the compatible relation between every two datasets. The third step is to decide how to integrated them, including data access and data conversion, which can make use of the representational semantics such as *Access*, *Unit*, *Reference*, etc. The last step is to do the integration with help of tools with semantic

support such as SPARQL (W3C, 2008) query language, which can greatly automate the process.

3. How to query the provenance information and evaluate the data for an application.

Provenance means the origin, or the source of something, or the history of the ownership or location of an object. In our research, provenance is represented by the relations between data, process or process chain, and the context under which the relations were generated. Context is associated with process. Thus, by recursively navigating the relations between processes and datasets we can eventually get the provenance tree. Or, by the reasoning ability, we can simply get all data entities and processes directly or indirectly associated with current data entity. To evaluate a dataset, we need significant properties of the target data, which are affected by its related data, process, and context. These properties can be inherited and reorganized as representational semantics of the output datasets by reasoning.

Figure 7: A Data flow from Data Collection to Processing

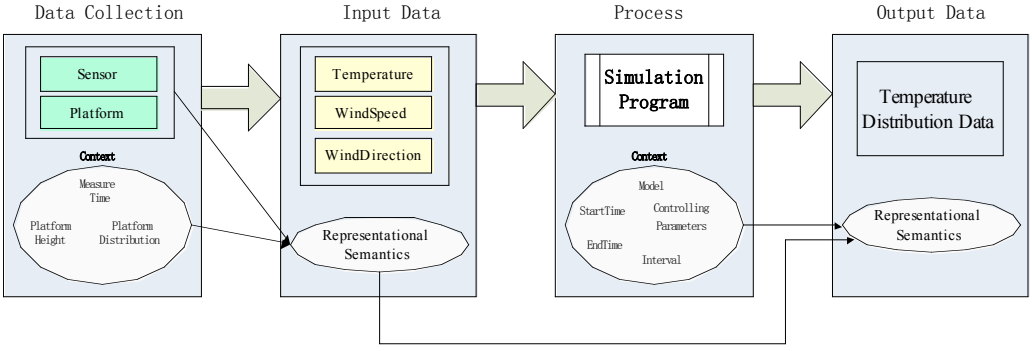


Figure 7 shows a data flow from the data collection to the result of simulation process. Lines in this graph indicate the dependency between metadata, contexts and representational semantics in different phases. The context of observation constructs the initial representational semantics, and then by combining with contexts of simulation and properties of the simulation model, the semantics for the output data is generated. Similar to rules in Table 2, here is an example how to make the output data inherit the semantic properties of input data.

$$has\_parent(?x,?y) \cap has\_platform\_distribution(?y,?d) \rightarrow has\_platform\_distribution(?x,?d)$$

The above means the sensor platform distribution information of the raw data will be inherited by all the output data. Then, we can operate the output data as an individual to check its representational properties for evaluation. An example of SPARQL to select the platform distribution is as follows:

select ?d where {dat100:output1000 *has_platform_distribution* ?d}

5. CONCLUSION

By making a clear distinction between the domain concepts and their representations, we propose to conceptualize the representational semantics to help both humans and computer systems understand the data better with their forms. Semantics are formalized in the form of ontology, which is specialized for the spatial data application areas, such as the SDI. With the ontology reasoning, we leverage different type of axioms to validate the data and infer additional properties for datasets. We demonstrated how to use this ontology to solve practical issues we faced in our research, which alleviate efforts for users to scrutinize the data, or the engineers to design program logic to analyze the data.

Yet more remains to be done to complete this functionality. First, most data providers will not provide metadata according to the ontology model, which requires an additional process to extract information from existing metadata standard, ontologies, and even text description. Second, friendly user interface is needed to hide the details of representational semantics from common users. Finally, to make this ontology work well with others, alignments between different ontology layers as depicted in Figure 1 are needed.

REFERENCES

Articles in journals

Benslimane, D., E. Leclercq, M. Savonnet, M. N. Terrasse, and K. Yétongnon, 2000, On the definition of generic multi-layered ontologies for urban applications: Computers, Environment and Urban Systems, v. 24, p. 191-214.

Beran, B., and M. Piasecki, 2009, Engineering new paths to water data: Computers & Geosciences, v. 35, p. 753-760.

Buccella, A., A. Cechich, and P. Fillottrani, 2009, Ontology-driven geographic information integration: A survey of current approaches: Computers & Geosciences, v. 35, p. 710-723.

Fonseca, F., C. Davis, and G. Camara, 2003, Bridging ontologies and conceptual schemas in geographic information integration: Geoinformatica, v. 7, p. 355-378.

Fonseca, F., and J. Martin, 2007, Learning the differences between ontologies and conceptual schemas through ontology-driven information systems: Journal of the Association for Information Systems, v. 8, p. 129-142.

Hjelmager, J., H. Moellering, A. Cooper, T. Delgado, A. Rajabifard, P. Rapant, D. Danko, M. Huet, D. Laurent, H. Aalders, A. Iwaniak, P. Abad, U. Duren, and A. Martynenko, 2008, An initial formal model for spatial data infrastructures: International Journal of Geographical Information Science, v. 22, p. 1295-1309.

Villa, F., 2007, A semantic framework and software design to enable the transparent integration, reorganization and discovery of natural systems knowledge: Journal of Intelligent Information Systems, v. 29, p. 79-96.

Web-based articles

JPL, Semantic Web for Earth and Environmental Terminology (SWEET). http://sweet.jpl.nasa.gov

FGDC, 1998, Content Standard for Digital Geospatial Metadata. http://www.fgdc.gov/metadata/csdgm/

PaoloBouquet, MarcEhrig, JérômeEuzenat, EnricoFranconi, PascalHitzler, MarkusKrötzsch, LucianoSerafini, GiorgosStamou, YorkSure, and SergioTessaris, 2004, Specification of a common framework for characterizing alignment: Deliverable D2.2.1, Knowledge Web NoE. http://www.aifb.uni-karlsruhe.de/WBS/phi/pub/kweb-221.pdf

W3C, 2008, SPARQL Query Language for RDF. http://www.w3.org/TR/rdf-sparql-query/

Proceedings

Fagin, R., P. G. Kolaitis, R. J. Miller, and L. Popa, 2003, Data exchange: semantics and query answering: Database Theory - ICDT 2003. 9th International Conference. Proceedings, 8-10 Jan. 2003, p. 207-24.

Ludascher, B., A. Gupta, and M. E. Martone, 2001, Model-based mediation with domain maps: Proceedings of 17th IEEE International Conference on Data Engineering, 2-6 April 2001, p. 81-90.

Quix, C., L. Ragia, L. L. Cai, and T. Gan, 2006, Matching schemas for geographical information systems using semantic information, in R. Meersman, Z. Tari, and P. Herrero, eds., On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, Pt 2, Proceedings: Lecture Notes in Computer Science, v. 4278: Berlin, Springer-Verlag Berlin, p. 1566-1575.

Torres, M., and S. Levachkine, 2007, Obtaining Semantic Descriptions Based on Conceptual Schemas Embedded into a Geographic Context, in V. Popovich, M. Schrenk, and K. Korolenko, eds., Information Fusion and Geographic Information Systems, Proceedings: Lecture Notes in Geoinformation and Cartography: Berlin, Springer-Verlag Berlin, p. 209-222.

Wache, H., T. Vogele, and U. Visser, 2001, Ontology-based integration of information-a survey of existing approaches: 17th International Joint Conferences on Artificial Intelligence, p. 108-117.

Books

Calvanese, D., G. De Giacomo, and M. Lenzerini, 2002, A framework for ontology integration, in I. Cruz, S. Decker, J. Euzenat, and D. McGuinness, eds.,

Emerging Semantic Web: Frontiers in Artificial Intelligence and Applications, v. 75: Amsterdam, I O S Press, p. 201-214.