On Integrating Scientific Resources through Semantic Registration*

Shawn Bowers Kai Lin Bertram Ludäscher
San Diego Supercomputer Center, UCSD
La Jolla, CA 92093-0505, USA
{bowers,klin,ludaesch}@sdsc.edu

1 Introduction

In many data-centric scientific applications it is common to register datasets and computational services with a *federation registry* (also commonly called a *catalog*, *directory*, or *repository*). For example, the scientific data-handling system under development in the SEEK project[1] must consider various dataset registries, including: MCAT, for access to SRB-registered datasets [9]; Metacat, for KNB-registered datasets [6]; DiGIR, for UDDI-registered data [3]; and Xanthoria, an XML-based data registry [10]. A challenge for SEEK, and similar efforts such as GEON[2], is to provide uniform access to registries and registered resources, based on emerging web and grid standards.

Providing uniform access is especially difficult for scientific resources due to their inherent structural and semantic heterogeneity. We focus on the use of ontologies for *semantically registering* scientific resources, and consider the implications of semantic registration for enabling uniform registry-based operations. In general, the use of ontologies offers richer contructs and more flexibility for classifying, discovering, and integrating scientific resources when compared with typical keyword-based metadata approaches.

Background: Why Registration? The purpose of resource registration is to facilitate the discovery of resources and the execution of operations over them. A *resource* can be a dataset (file), a collection of datasets (including nested collections), a database, or a web (or grid) service. Common examples of operations using registered resources include the following.

• *Data and service discovery*: A registered dataset d can be found by issuing a search query against d's semantic information stored in the registry. Often the result of such an operation will include a handle[3] to the resource, which may be a (possibly persistent) unique object identifier for d. Similarly, a web or grid service can be discovered and its input and output parameters understood if its WSDL description has been registered.

• *Data integration*: If the registered resource can be viewed as a database, the registry information includes the export schema (e.g., relational or XML Schema), and possibly other constraints (e.g., foreign keys, semantic integrity constraints, etc.) and query capabilities (e.g., full SQL vs. template-based queries only). Specific source queries can be executed directly by the end user (using an ontology), or indirectly via a database mediator system.

• *Remote service invocation and workflow enactment*: The web, including grid environments, is becoming widely used as distributed computation platform. For example, standard web services can be invoked based on their registered WSDL descriptions.

• *Data transformation*: With the availibility of distributed services comes the possibility of chaining them together to create complex scientific workflows. To connect and compose heterogeneous services, input and output data types must be aligned and data must be transformed as the services are executed [2]. The new workflow may also be stored and registered.

The Problem. There are various standards and implementations of resource registries for distributed data and services, however, the principles of resource registration and its implications for scientific-data integration are not well understood. The advent of semantic-web standards such as OWL and RDF provides new opportunities for *semantic* registration of resources—where resource descriptions go beyond descriptive (free form) metadata, database schemas, and WSDL descriptions.

We propose a framework for semantic registration of data sources that makes the following contributions. First, we register datasets and services with their structural descriptions (*structural registration*) and associated query capabilities. Second, ontologies become registered resources themselves, allowing us to (1) associate data objects (i.e., the logical data items within resources) with specific concepts and properties of registered ontologies (*semantic data registration*), and (2) express inter-ontology constraints (*articulation registration*), which are ontologies themselves, and can thus can be registered as well. Finally, using the framework, we identify desirable properties of semantic registration mappings for data sources, and illustrate their use for our goal of enabling the integration of scientific data.

*Work supported by NSF grants No. ITR 0225673 (GEON) and ITR 0225676 (SEEK).

[1] See http://seek.ecoinformatics.org

[2] See http://www.geongrid.org

[3] A URL, DOI, SRB-id, LSID (life-sciences-id), a GRI (grid-id), etc.

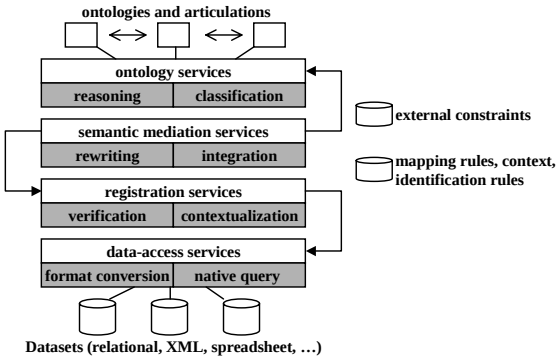

Figure 1. Architecture overview.

2 Preliminaries

Figure 1 shows an overview of the architecture of our proposed framework. The main components include services for ontological reasoning, mediation, registration, and data access. We assume the actual federation registry (not shown) stores the core registry information, including ontologies, ontology articulations, external semantic constraints (e.g., unit conversion rules and other scientific formulas), registration mapping rules, database schemas, and service descriptions. Here, we provide preliminaries for formalizing the components of our framework. We discuss resource registration in more detail in the next section.

Underlying Formalism. We use first-order logic as a standard, underlying formalism. *Syntax:* We consider signatures Σ with predicate symbols Σ_P and function symbols Σ_F. By $\Sigma_{P.n}$ ($\Sigma_{F.n}$) we denote the subsets of n-ary predicate (function) symbols; $\Sigma_C = \Sigma_{F.0}$ are constants. *Semantics:* A first-order structure $\mathcal{I}$ interprets predicate and function symbols as relations and functions, respectively; constants are interpreted as domain elements. Given $\mathcal{I}$ and a set of formulas Φ over Σ, we say that $\mathcal{I}$ is a *model* of Φ, denoted $\mathcal{I} \models \Phi$, if $\mathcal{I} \models \varphi$ for all $\varphi \in \Phi$, i.e., all formulas in Φ are satisfied by $\mathcal{I}$. We can implement constraint checking by evaluating the query $\{\bar{x} \mid \mathcal{I} \models \varphi(\bar{x})\}$.

Ontologies. An ontology O is a set of logic axioms Φ_O over a signature $\Sigma = \mathbb{C} \cup \mathbb{R} \cup \mathbb{I}$ comprising unary predicates $\mathbb{C} \subseteq \Sigma_{P.1}$ (*concepts*), binary predicates $\mathbb{R} \subseteq \Sigma_{P.2}$ (*roles*, *properties*), and constants $\mathbb{I} \subseteq \Sigma_{F.0}$ (*individuals*). Φ_O is usually from a decidable first-order fragment; most notably *description logics* [1]. A structure $\mathcal{I}$ is called a *model* of an ontology Φ_O, if $\mathcal{I} \models \Phi_O$.

We can view controlled vocabularies and metadata specifications as limited, special cases of ontologies. A *controlled vocabulary* can be viewed, e.g., as a set $\mathbb{I} \subseteq \Sigma_{F.0}$ of *individuals* (constants); a set of named concepts $\mathbb{C}$; or even a full ontology signature Σ (if it contains relationships between terms of the controlled vocabulary). In either case, there are no axioms and hence no defined logical semantics. A *metadata specification* can be seen as an instance of an ontology having only binary predicates $\mathbb{R}$ denoting the metadata properties (e.g., *title*, *author*, *date*; cf. Dublin Core). Again, the absence of axioms means that no logical semantics is defined.

Resource Renaming. In the federation registry, we avoid name clashes between vocabularies from different scientific resources (datasets, services, etc.) by assuming each resource has a globally unique identifier i (e.g., implemented as a URI). We then rename symbols accordingly: Every symbol in Σ_i is prefixed with its resource-id i to obtain a unique vocabulary $\Sigma_i' := \{i.s \mid s \in \Sigma_i\}$, allowing new resources to join the federation without introducing identifier conflicts. A resource-id is also commonly referred to as a *namespace*. By $\mathtt{id}(s)$ we denote the globally unique *resource identifier* of a symbol s.

3 Resource Registration

Registering Ontologies and Articulations. An ontology O is registered by storing its logic axioms Φ_O and its signature Σ_O in the federation registry.[4] An *articulation ontology* A links ontologies O_i and O_j and is given as a set of axioms Φ_A over $\Sigma_A = \Sigma_{O_i} \cup \Sigma_{O_j}$, thereby logically specifying inter-ontology correspondences. For example, $i.\mathtt{C} \equiv j.(\mathtt{D} \sqcap \exists \mathtt{R}.\mathtt{E})$ is an *articulation axiom* $\varphi \in \Phi_A$ and states that the concept $\mathtt{C}$ in O_i is equivalent—in terms of O_j—to those $\mathtt{D}$ having at least one $\mathtt{R}$-related $\mathtt{E}$. This is expressed equivalently as follows (using first-order logic syntax):

$$\forall x : i.\mathtt{C}(x) \leftrightarrow j.\mathtt{D}(x) \wedge \exists y : j.\mathtt{R}(x, y) \wedge j.\mathtt{E}(y) \qquad (\varphi)$$

Note that φ is an *intensional* definition: we have not said how we can access instance objects (implicitly referred to via variables x and y), i.e., how to populate $\mathtt{C}, \mathtt{D}$, etc., as classes. Finally, expressing inter-ontology articulations as ontologies achieves closure within the framework: There is no need to manage a new type of artifact and we can reuse the given storage, querying, and reasoning techniques.

Structural Data Registration. When registering a database, schema-level information and query capabilities should be included to facilitate queries by the end user or a mediator system. Specifically, the database registration information contains:

• The database *schema* Σ_D. In the case of a relational database D, Σ_D is a schema $\mathbf{D} = \{\mathbf{R}_1, \ldots, \mathbf{R}_n\}$, where each $\mathbf{R}_i$ is the schema of an exported relation.

[4] For example, OWL is the W3C standard for representing ontologies and thus can be used as a concrete syntax for Φ_O.

• A set of local *integrity constraints* Φ_D. We can distinguish different types of constraints, e.g., structural (such as foreign key constraints) and semantic constraints.

• A *query capability specification* Π_D. For example, Π_D may be a set of access patterns [7], prescribing the input/output constraints for each exported relation. More generally, Π_D may be given as a set of view definitions (possibly with access patterns) supported by the source D. If D provides full-fledged SQL access, this may simply be encoded by a reserved word: $\Pi_D = \{\texttt{SQL}\}$.

To register the structural definition of the data source, a data-access handler must also be provided (similar to a wrapper). The data-access handler provides basic services for executing underlying queries and converting data to a common format for use by the registration and mediation service.

Semantic Data Registration. A semantic data registration registers the association between data objects in a database D and a target ontology O. Let $k = \texttt{id}(D_k)$ and $j = \texttt{id}(O_j)$ be the unique resource identifiers of D_k and O_j, respectively. The *semantic data registration* of D_k to O_j is given by a set of constraints Ψ_{kj}, where each $\psi \in \Psi_{kj}$ is a constraint formula over $\Sigma_D \cup \Sigma_O$. For example, the semantic data-registration formula $\psi =$

$$\forall x \forall y : j.\texttt{D}(x) \wedge j.\texttt{R}(x, y) \leftarrow \exists z : k.\texttt{P}(x, y, z) \wedge k.\texttt{Q}(y, z)$$

is a constraint that states ontology O's concept D and its role R can be "populated" using certain tuples $\texttt{P}(x, y, z)$ from D. When the semantic data-registration constraint ψ and the above articulation φ are combined, we see that data objects from D_k can almost be linked[5] to concepts like $i.\texttt{C}(x)$ in the ontology O_i, despite the fact that D_k was registered to O_j and only indirectly to O_i (via an articulation $O_i \leftrightarrow O_j$).

As shown in Figure 1, we define the semantic context of a data source as consisting of the portion of the ontology the source is registered to (labeled *contextualization* in the figure). Using context information, a discovery query can often be answered using the ontological context implied by a semantic registration. The registration service also provides operations to verify the correctness of semantic registrations.

4 Properties of Semantic Data Registration

This section further defines the steps involved in semantic data registration. In particular, we clarify the result of semantically registering a dataset as a *partial model* and motivate the need for additional, data-object *identification steps* (see Figure 2).

[5]For the body of φ to fire, we also need to establish that $\texttt{E}(y)$ holds.

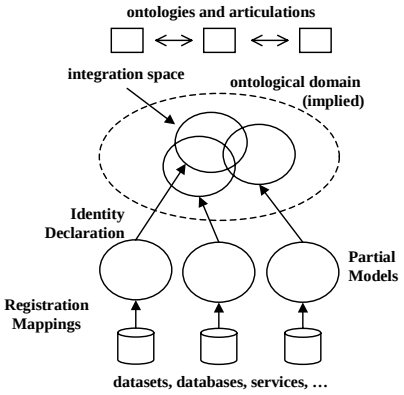

Figure 2. Result of semantic data registration.

Registering Partial Models. A dataset D that is registered to an ontology O contributes to the extent of the federation relative to O. Registered datasets are not materialized according to the ontology; instead the registration mappings are used to access the underlying sources when needed. A dataset D is eligible to be registered to an ontology O in the federation if and only if it can be interpreted as a *partial model* $\mathcal{I}$ of O, denoted $\mathcal{I} \models_p \Psi_O$, which implies $\mathcal{I} \cup \mathcal{I}' \models \Psi_O$ for some unknown $\mathcal{I}'$.

A partial model differs from a true model of the ontology in that some required information may be missing. We denote the interpretation induced by applying a semantic data registration Ψ_D of database D to an ontology O as $\mathcal{I}_D$. If the latter is a partial model $\mathcal{I}_D \models_p \Psi_O$, then the model $\mathcal{I}_D \cup {\mathcal{I}'}_D \models \Psi_O$ contains an unknown or hidden part ${\mathcal{I}'}_D$. As more sources are registered, more of ${\mathcal{I}'}_D$ may become known.

When an interpretation induced by a semantic data registration is not a partial model of an ontology, we say that the interpretation is *inconsistent*. An inconsistent interpretation often violates a datatype, cardinality, or disjoint constraint in the ontology. When possible, we wish to automatically verify that a semantic data registration is consistent, e.g., by ensuring that the dataset is a model of the ontology. If the model is only partial we may notify a data provider that the dataset is incomplete with respect to the ontology.

Identification Declaration. Semantic data registration allows a dataset to be interpreted as a partial model of an ontology, but does not necessarily provide enough information to identify the same individuals across multiple datasets, which is essential for data integration.[6] The ability to identify equivalent data objects across different datasets is needed in practice because each dataset may only provide a portion of the information concerning a particular object. As shown in Figure 2, we consider an additional

[6]The problem is general to metadata and controlled vocabulary approaches as well, neither of which provide the ability to associate semantic information to data objects *within* a resource.

registration step called *identification declaration* that allows data providers to state how data objects should be identified across data sources.

Object identity can be defined in a number of ways. First, a semantic data registration can be augmented with *mapping tables*, which map individual data items to recognized individuals in $\mathbb{I}$, i.e., individuals that are established instances within an ontology and come from an authoritative registry. For example, an ontology may prescribe a particular species taxon, where individuals represent globally-identified species, and a mapping table within a semantic data registration may then associate dataset species codes to these global species identifiers. Second, external rules may be used for determining identity, similar to keys in a relational database.[7] As an example, we may have a rule that ISBN codes uniquely identify publications, thus, registering to an ISBN property uniquely identifies the data object. Finally, a data provider may give data-object correspondences between registered data sets. Thus, a data object is explicitly given as equivalent to another data object (although the specific identifiers of the objects may not be authoritative). We store identity information as part of the semantic constraints repository shown in Figure 1.

5 Data Integration via Semantic Registration

We identify four classes of semantic data-registration expressibility (in terms of data integration) as follows.

• ***Concept-as-keyword registration***. We can consider metadata annotations using keywords from a controlled-vocabulary as (a weak form of) registration mappings. For example, we can assign a concept such as *geologic-age* to the dataset as a whole.[8] Such a mapping states that the dataset contains data objects, and those data objects refer to individual geological ages. However, we cannot consider or obtain each such separate (geologic-age) data object in the dataset. Clearly, such a registration cannot be used for integration, however, it can be used for dataset discovery: We do not have access to the individual objects, so the best we can do is find the dataset that contains such objects.

• ***Local data-object identification***. Local data-object identification is the typical result of a registration mapping, where local identifiers are used to identify logical data items within a dataset. In this case the identities of the individuals are local to the source, and thus, cannot be used to combine data objects from multiple sources.

• ***Global data-object identification***. The result of globally identifying data objects is that it becomes possible for a mediator to recognize identical individuals in multiple datasets. The result of global data-object identification is the ability to perform object fusion [8] at the mediator.

• ***Property identification***. If within a given dataset we relate two globally-identified data objects with an ontological relation in $\mathbb{R}$, it becomes possible to join information across datasets (assuming at least one data object occurs in at least one other relation in another source). This situation represents a stronger form of integration compared to simple object fusion, and is required for wide-scale data integration.

[7] Alternatively, some description logics include keys definitions [5].

[8] For example, by registering the dataset's resource identifier with the concept, or registering all rows of the dataset with the concept.

6 Conclusions

We have presented a general logic-based framework for semantically registering resources with ontologies, and discussed some properties of the resulting semantic registrations. In [4], an early implementation of a semantic registration procedure is described in the context of an ontology-enabled geologic map integration system. It allows the user to register a spatial dataset (shapefile) to one rock classification ontology (such as the British Geological Survey), and allows the user to query the data using a second ontology (such as the Geological Survey of Canada). We are currently extending our framework to include registration of services, as found, e.g., in scientific workflow systems.

References

[1] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[2] S. Bowers and B. Ludäscher. An ontology-driven framework for data transformation in scientific workflows. In *Proc. of the 1st Intl. Workshop on Data Integration in the Life Sciences (DILS)*, volume 2994 of *LNCS*, pages 1–16, 2004.

[3] Distributed Generic Information Retrieval (DiGIR). `http://digir.sourceforge.net/`.

[4] K. Lin and B. Ludäscher. A system for semantic integration of geologica maps via ontologies. In *Proc. of the Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data (SCISW)*, 2003.

[5] C. Lutz, C. Areces, I. Horrocks, and U. Sattler. Keys, nominals, and concrete domains. In *Proc. of the 18th Intl. Joint Conf. on Artificial Intelligence IJCAI*, 2003.

[6] Metacat. `http://knb.ecoinformatics.org/software/metacat/`.

[7] A. Nash and B. Ludäscher. Processing unions of conjunctive queries with negation under limited access patterns. In *Proc. of the 9th Intl. Conf. on Extending Database Technology (EDBT)*, volume 2992 of *LNCS*, pages 422–440, 2004.

[8] Y. Papakonstantinou, S. Abiteboul, and H. Garcia-Molina. Object fusion in mediator systems. In *Proc. of the 22nd Intl. Conf. on Very Large Data Bases (VLDB)*, pages 413–424, 1996.

[9] SDSC Storage Resource Broker (SRB). `http://www.npaci.edu/DICE/SRB/`.

[10] Xanthoria: A distributed query system for XML encoded data. `http://ces.asu.edu/bdi/Subjects/Xanthoria`.