Integrating Semantic Technology with Legacy Databases

Virginia Boy

East Rochester High School

LLE Advisor: Richard Kidder

March 2013

Abstract:

A study was conducted to find the optimal approach for integrating semantic technology across the extensive data banks utilized by the Laboratory for Laser Energetics (LLE). In addition to the immense amount of information stored by LLE, there are multiple repositories of data, making it difficult to locate and process information critical to the operations of the laser facilities. Semantic technology facilitates information processing by linking data and associated properties into an ontology model. The use of Java frameworks such as Jena and D2RQ was explored in order to allow this scattered data to be imported into a single ontology without duplicating the information.

Introduction:

The Current LLE Database System

Over the years, the Laboratory for Laser Energetics (LLE) has accrued vast quantities of information relating to its operations, spanning from data on the diagnostics and optics used to the scientists and specialists who design and support them. There is also a large quantity of data that is vital to the safe and efficient operation of the Omega Laser Systems. Much of this information is contained in the LLE database system, which currently exists in a scattered and

decentralized form, requiring both knowledge of the nonlinear database setup and SQL queries in order to access. Querying for data in the system can be complicated and confusing; success depends greatly on the querying skills and reasoning of the user. As a result of these difficulties, few people are able to utilize this resource in its entirety. Usage of the database is generally limited to the specific area of LLE in which a scientist works.

An Overview of Semantic Web Technology

Semantic web technology is a primitive artificial intelligence that allows a computer to link meanings to data. This meaning is applied through tags known as metadata, and allows the system to associate relationships within the data and reason on the system [1]. Using this technology, each object is represented by an individual, which is then assigned properties. Various categories of properties exist that allow these individuals to be grouped with both other individuals and corresponding primitive datatypes, creating an interconnected web of information. Object properties link individuals, while datatype properties are used to assign values to individuals. Individuals that are similar or share relationships are grouped into units known as classes. This allows objects of the same type to all be grouped together in one place. Individuals can belong to multiple groups, giving the ontology the capabilities to sort the data in multiple ways. In addition to the categories of relationships that exist, there are also properties and restrictions that can be associated with these relationships, allowing for greater inferencing capabilities of the system. A functional property warrants that each individual can only be related to one other individual through this property. This allows the computer to infer whether multiple individuals refer to the same entity. Properties can also be labeled as transitive, simplifying the linkage of long chains of objects along the same property. Symmetric properties also facilitate groupings of individuals linked to each other by the same property. All of these

properties group together similar information, increasing the efficiency of data processing and eliminating the need for users to parse through large amounts of data in order to find the desired set of information [2].

One of the major benefits of applying semantic technology to the LLE database system is the ability of reasoners to classify and make inferences on the asserted relationships in the system. Reasoners use user-defined properties and a standard set of rules to classify the ontology. Its classification is used to both make inferences on relationships in the ontology and to populate defined classes with corresponding individuals. Reasoners commonly used with Semantic Web technology, such as Pellet and Fact++, use open world reasoning (OWR). OWR dictates that nothing can be assumed that hasn't been explicitly stated. If a reasoner following this assumption is classifying an ontology and detects any sort of ambiguity as to what classifications a class should receive, the reasoner makes no assumptions and the class is not sorted into the questioned groupings. This is an important feature of semantic reasoners, as it eliminates errors based on ambiguity. Although they are highly versatile, reasoners do require that the given ontology has no inconsistencies, and doesn't contain any conflicting data.

Framework Technologies

One of the major obstacles in applying semantic technology to the LLE database system is importing all of the data into a single ontology setup. As was determined by previous studies, it is impractical to attempt to transfer all of this data by hand [3]. This database would be impossible to keep up to date, and data inaccuracies could result. This system would also require duplication of all of the information at LLE. This data replication would be redundant and would require massive amounts of memory storage in order to be maintained.

Previous projects identified Apache Jena as a potential Java framework that would allow a more automated population of the ontology [4]. Jena is an open source semantic web framework that contains an application programming interface (API). It has capabilities to extract data from files, databases, URLs, or a combination thereof, and can export to resource data framework (RDF) graphs. In addition to being flexible enough to read data from various types of sources, it is easily queried through SPARQL, a query language for databases, and provides support for OWL, a language commonly used in ontologies. The D2RQ platform was also discovered for use in automating the population of the ontology. D2RQ is used to map non-RDF databases and allows them to be imported and manipulated using Jena. This is invaluable to the function of the LLE ontology as the majority of information currently held in the database system is contained in relational, non-RDF databases. Figure 1 represents a visual model of the framework software setup.

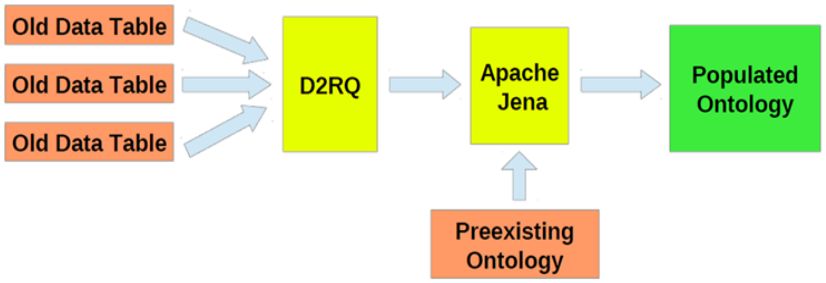

Figure 1: Graphical representation of the framework structure setup used to form the ontology. It is important to note that data is merely referenced through this process, and is not duplicated.

Research and Development

Implementing Jena and D2RQ

The first step in automating ontology population was to implement the Jena framework. This was done by running Jena in a Java programming environment. Jena was used to generate and manipulate an ontology during the runtime of the program, and upon its conclusion the ontology was exported and saved to a file. The saved file could then be opened using Protégé, an

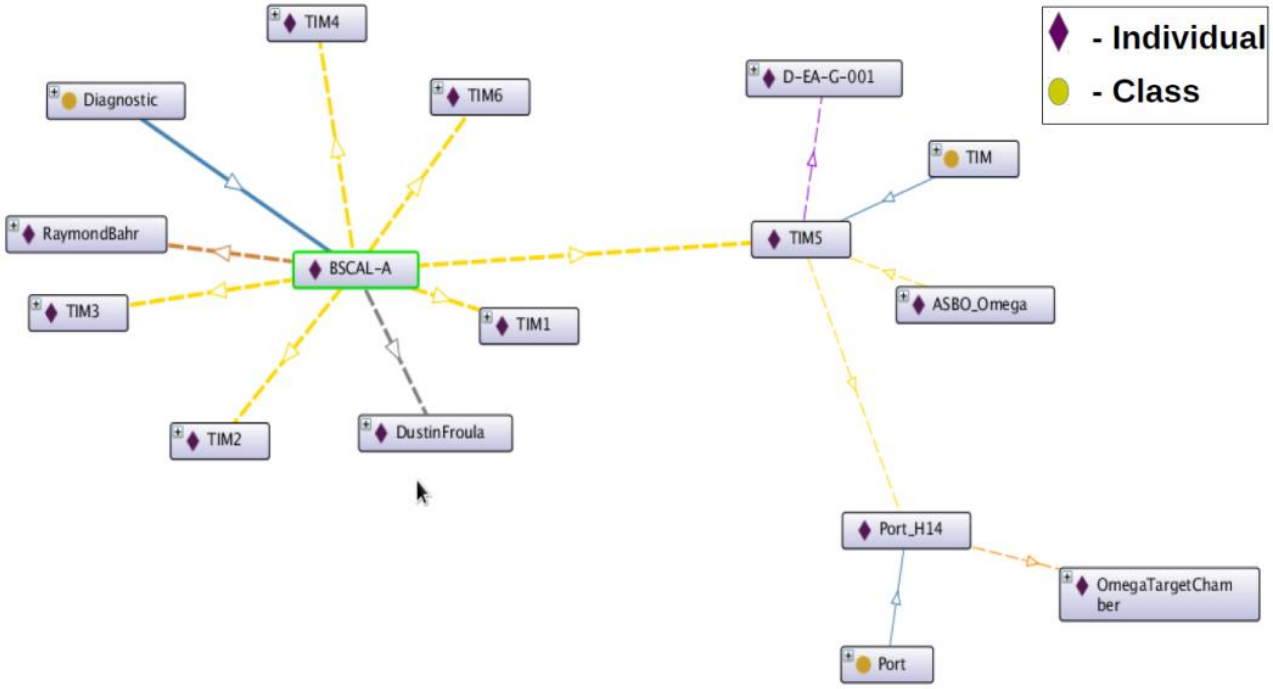

Figure 2: Graphical representation of relationships by Protégé. Each relationship, denoted by a color-coded line, works to join data into an interconnected web of information.

ontology mapping framework as shown in figure 2. Through this method, the produced ontology could be easily analyzed using a visual interface. Using a combination of the Jena frameworks and Protégé, it was demonstrated that Jena could easily import premade ontologies, as well as any other user-provided information. This conglomeration of data could be seamlessly merged to produce a single, coherent ontology. Once Jena had been successfully tested, D2RQ was added in order to expand the capabilities of the reasoning system. This program would allow all of the non-RDF database files in the current LLE systems to be imported into the Jena framework. D2RQ required that each relational database table be mapped, a simple process that required little time to create. The mapping file defined the information contained in the table, declaring the type of objects and relationships contained therein. Since D2RQ acts as a bridge between the program itself and the actual data files, each table needed to be mapped only once. The data contained within the tables was not replicated and stored by the system, so each time

the program ran, it referenced the files anew. This provided some amount of dynamicity to the created ontology, as any changes made to the tables were reflected in the ontology the next time the program ran. Several database tables were mapped, and this data also merged seamlessly to produce a large ontology with many classes of individuals.

Reasoning on and Querying the Ontology

Once an ontology was compiled from various sources, a reasoner was applied in order to draw inferences from the information. The reasoner used, known as Pellet, was applied within the programming environment. Pellet was selected as the reasoner as it was able to draw more conclusions from sets of data when compared to built-in reasoners such as Fact++. Pellet was able to classify and query the ontology almost instantaneously, providing the user with more relevant data, and requiring no extra time or effort from the user. Pellet assigned properties and direct instances to various classes, allowing the computer to draw basic conclusions that would otherwise have had to be made by a person. Most importantly, individuals were classified uniformly, with no real variance based on the source or type of data. This was a crucial aspect of the project, as this program was designed to centralize data.

Once the ontology had been explored using reasoners, the created ontology file was set up on the LLE server system and queried using SPARQL. The ontology was given some sample

BSCAL-A

TIM	Port	Azimuthal Angle	Polar Angle
TIM 4	P6	342	63.44
TIM 6	P7	162	116.57
TIM 2	H3	162	37.38
TIM 5	H14	270	100.81
TIM 1	P3	126	63.44

Figure 3: Results returned from a SPARQL query. This query requested data that was gathered from multiple sources using the reasoning capabilities of the system.

queries, including one for an LLE diagnostic, BSCAL-A. The computer was able to return key pieces of information relating to this diagnostic, as shown in figure 3. It is important to note that all of the returned data was referenced by the system through a single query to a single source.

The results were determined through the logical inference of facts and data asserted in the ontology. On current systems, the same data would have had to be referenced from various data tables scattered throughout the current LLE database system.

Oracle and Future Work

The integration of semantic technology across the LLE database is still in its early stages. This project successfully demonstrated the facility of applying this technology using external frameworks such as Jena and D2RQ in order to unify and reason on various types of data. This is only one of several identified methods to apply semantic technologies. Another potential method that should be explored is through Oracle, a commercial database system in use at LLE.

Oracle has several advantages over the use of external frameworks. In addition to having the same reasoning capabilities, the Oracle system is more dynamic in its setup than external frameworks, providing more-up-to-date information. Oracle has built-in programming to manage the import of various types of data including both RDF and non-RDF data tables. This programming accomplishes the same tasks as the framework setup, without many of the drawbacks, such as incompatibilities and bugs within the third party software used. Oracle also includes capabilities that are able to restrict user access to protected information, adding a level of security that may otherwise be difficult to attain [5].

Further work includes exploring the capabilities of Oracle. Additionally, more work must be done to continue improving upon and populating the LLE ontology with accurate and relevant data and relationships.

Acknowledgments:

I would like to thank Dr. Stephen Craxton and Mr. Richard Kidder for granting me the opportunity to work on this project for the Laboratory for Laser Energetics. I would like to thank Mr. Richard Kidder and Colin Kingsley for all of the assistance and support they gave me on my project. I would also like to thank Robert Cooper and Brandon Avila for their past work relating to my project.

References:

1. "W3C." *Semantic Web - W3C*. W3C, n.d. Web. July-Aug. 2012. <http://www.w3.org/standards/semanticweb/>.

2. Horridge, Matthew. "A Practical Guide to Building OWL Ontologies Using Protégé and CO-ODE Tools Edition 1.3." University of Manchester, 24 Mar. 2011. Web. July-Aug. 2012.

3. B. Avila, "Optimizing LLE Information Operations through Natural Language Processing," 2011 Summer High School Research Program at the University of Rochester's Laboratory for Laser Energetics.

4. R. Cooper, "Designing and Implementing an Ontology for LLE Experimental Diagnostics," 2010 Summer High School Research Program at the University of Rochester's Laboratory for Laser Energetics.

5. Ashdown, Lance, and Tom Kyte. "Oracle Database Concepts." *Oracle Database*. Oracle, Sept. 2011. Web. July 2012.