Using CIDOC CRM for dynamically querying ArSol, a relational database, from the semantic web

In the MASA Consortium context, the Laboratoire Archéologie et Territoires propose to open the ArSol database, a system for processing archaeological data, to the semantic web, using the CIDOC-CRM ontology and tools that implements Ontology-Based Data Access (OBDA) principles. After designing a set of mappings from ArSol fields to the CIDOC CRM ontology, we implement the software architecture to query ArSol via a SPARQL endpoint. We used ontop-, a Protégé plugin, to design the OBDA mappings that are necessary for the SPARQL-to-QL rewritings. Our final goal is to devise an application that will offer a single interface to query several distributed and independent archaeological databases, with heterogeneous structures, using CIDOC CRM to relate them to each other. Querying ArSol in SPARQL via the CIDOC CRM is an important step towards this goal.

CIDOC CRM ontology

Initiated in 1994, this ontology was originally designed for museums, but its scope is so broad that we can find most of the elements and properties specific to archaeological data, in particular, entity "E22 Man-Made Object" corresponding to artefacts and entity "E23 Man-Made Feature" corresponding to features.

We see this ontology as a consensual description overlay through which the query can pass, allowing communication with each database thanks to a system of equivalence, or "mapping". By using ontology, data become free from the database management system software and from the own database structure.

Mapping

The mapping consists in matching each field in the database with an entity of ontology respecting the whole hierarchical tree and the properties that organize the entities in relation to each other.

The RDF tripled "subject-predicate-object" becomes "source Entity-Property-Target Entity". The target entity at the end of each branch is the element that must be associated with the field in the database. That field is collected in the mapping using a SQL query in the database.

Software architecture

For communicating the ontology with database using mapping as key match, a solution lies into the principles of OBDA systems (Ontology Based Data Access), recently introduced in the semantic web. With this OBDA layer, ontology is the access point to query the data, while the information remains in the databases used by researchers. OBDA system consists of: an ontology (semantic level), various data sources (in our case, ArSol), a set of mappings expressing the relationship between the data source and ontology, and an application layer to manipulate and query the system.

Protégé and -ontop-

We use Protégé (free software dedicated to ontologies) and -ontop-, an extension developed at the University of Bocconi (Italy). It first allows us to specify the matches between ontology and database, then to evaluate SPARQL queries about ontology, whose results are searched for in the database. The integrated user interface of Protégé helps us to make the OBDA file containing statements of correspondences between ontology and data source.

SPARQL endpoint

To offer online query for web applications, which is called "SPARQL endpoint", we need then a HTTP server like jetty (java) with the web application Sesame-Workbench that contains Quest-, ontop-version independent of Protégé and ensuring the evaluation of SPARQL querires. Ontology is loaded as an XML-OWL file. The data source is defined by indicating the required drivers to allow Java to communicate with the source database software.

Interoperability

Now, to give it the meaning to interoperability, we expand our number of data sources, doing the same process with others archaeological databases. The only constraint is that each database should have a minimum of descriptor fields, which is the case for most databases used in archaeology, and that it must be web-accessible.

The SPARQL endpoint is essential for the web services that will query ArSol but we also have to allow humans to access ArSol and other databases via this semantic level. So we will develop a web interfaces for users, with the SPARQL endpoint.