Defining Several Ontologies to Enhance the Expressive Power of Queries

Bich-Liên Doan and Yolaine Bourda
Computer Science Dpt.
SUPELEC, 3 rue Joliot Curie, 91192 Gif-sur-Yvette, France
Bich-Lien.Doan@supelec.fr, Yolaine.Bourda@supelec.fr

ABSTRACT
The use of ontologies is a key step forward for describing the semantics of information on the Web. It is becoming more and more important to make the information machine-readable, since the volume of data is continuously growing. In the educational area, metadata are considered to be helpful in such a process. We propose to enrich the description of educational resources by introducing several levels of description of concepts, and to make them machine- readable by using a formal language of ontology, OWL. Using both this ontology and the expressive power of an OWL query language to query pedagogical resources will improve the retrieval and interchange of educational resources.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]:

General Terms
Standardization, Languages

Keywords
Semantic Web, ontology, OWL, information retrieval

1. Introduction

Educational resources available on the Web are intended to be shared, accessed or reused. Because of the ambiguity of the natural language (synonymy, polysemy, homonymy, multilingualism) the answers are spoilt by noise. Actually, keywords of the query are matched with indices extracted from the Web pages, but neither the semantics nor the structure are taken into account by the search tools. Some solutions have been proposed in order to explain the semantics of the Web: we note the recommendation of metadata Dublin Core [1] and more specifically the LOM [2] for e-learning resources. The W3C proposed the RDF standard [3] which aim is to represent the knowledge about the available Web resources. Using ontologies [4] is a further step to encourage authors to clarify the domain and the content of the resources, so that search tools could improve the precision and recall and agents could infer some knowledge. The Web Ontology Language (OWL) [5] was carried by the W3C to formalize ontologies on the Web. In this paper, we propose first to explicit a part of the pedagogical ontology of our engineer school, Supélec. Subsequently, we present some examples of queries with OWL-QL [6], using the predefined ontology.

2. Creating several views of an educational ontology

The pedagogical ontology concerns both the organization of an engineer school: Supélec and the content of a teaching program.

2.1. Description of an educational ontology

We first present a part of a UML model of the teaching organization at Supélec.

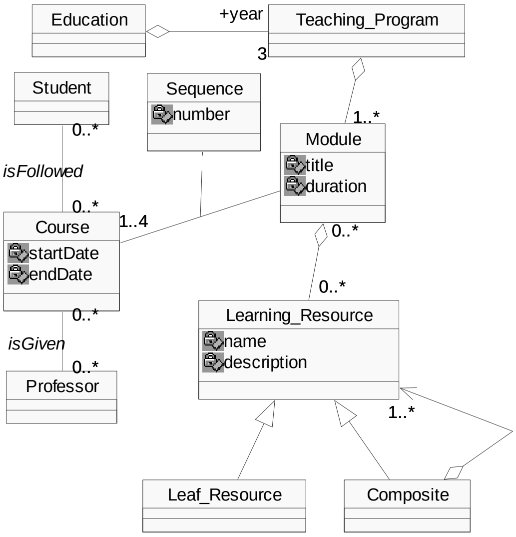

Figure 1. Education's organization

At Supélec, the education lasts three years. Each year is divided into four sequences and contains several teaching modules (each module corresponding to one course per sequence). A module contains learning resources which are either atomic or composite. In the LOM terminology, a learning object is considered as a learning resource, this equivalence can be expressed with an OWL restriction. The UML schema can be transformed into an RDF representation by the way of XPetal [7]. Because exact cardinalities cannot be expressed with RDF, we added an example of a cardinality constraint upon a property of the Education class. With OWL it is possible to specify that one member of Education has exactly three Teaching_Programs corresponding to year 1, 2 or 3. This is an extract of the OWL schema that we get:

```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:owl="http://www.w3.org/2002/07/owl#"
  <owl:Ontology rdf:about="file:/C:/BLD/Recherche/Articles/2003-
2004/onto1-supelec.owl "/>
  <owl:Class rdf:ID="Education">
<rdfs:subClassOf>
<owl:Restriction>
  <owl:onProperty rdf:resource=”#contains />
  <owl:cardinality rdf:dataype=
”&xsd:nonNegativeInteger”>3</owl:cardinality></owl:Restriction>
</rdfs:subClassOf></owl:Class>
<owl:ObjectProperty rdf:ID=”contains”>
<rdfs:domain rdf:resource=”#Education”>
<rdfs:range rdf:resource=”#Teaching_Program”>
</owl:ObjectProperty>
  <owl:Class rdf:ID="Teaching_Program"/>
  <owl:oneOf rdf:parseType=”Collection”>
   <Teaching_Program rdf:about=”#year_1”>
   <Teaching_Program rdf:about=”#year_2”>
   <Teaching_Program rdf:about=”#year_3”>
</owl:one of></owl:Class>
<owl:Class rdf:ID="Learning_Object" >
  <rdfs:subClassOf>
  <owl:Restriction>
  <owl:onProperty rdf:resource="#isComposedOf"/>
  <owl:allValuesFrom>
    <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Composite"/>
    <owl:Class rdf:about="#Leaf_Resource"/>
    </owl:unionOf>
    </owl:Class></owl:allValuesFrom>
    </owl:Restriction></rdfs:subClassOf>
</owl:Class>
```

The domain and scope of the second part of the pedagogical ontology are the learning resources participating in a teaching program, created by teachers or educational organizations. In order to preserve the semantics given by the LOM, we mention some definitions:
Learning Object: any entity that may be used for learning, education or training.
Category: a group of related data elements.
Data element: a data element for which the name, explanation, size, ordering, value space and datatype are defined in the LOM standard.

Now we present two other UML representations of a learning objects view and a LOM metadata view.

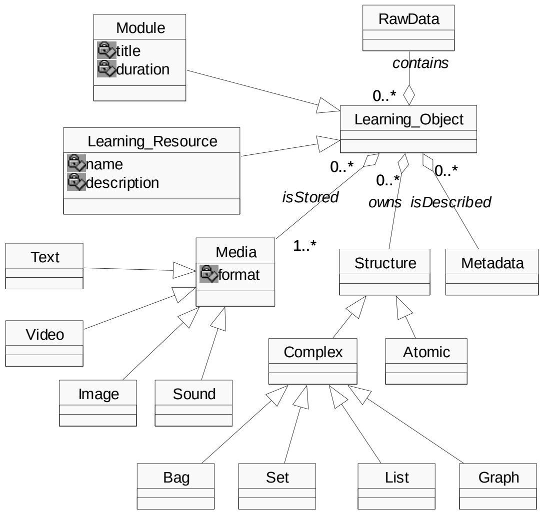

Figure 2. Model of Learning Objects

In figure 2, the Module and Learning resources are two types of Learning Objects in the terminology of LOM. A learning object is composed of raw data, media, structure and metadata. The media is text, sound, image or video. Each media type has a format (for example jpeg for an image, MP3 for a sound). The structure of a learning object is either atomic or complex (for example a definition, an example or a theorem is an atomic learning object whereas a module of software engineering is a complex one). Each learning object is described by a set of metadata which are detailed in figure 3. This representation reflects the view of LOM metadata with the concepts of categories, data elements and types of data element (the structure to represent the logical relationships between learning objects and the content to represent the content of a learning resource).This is an extract of the OWL representation of the learning object's view.

```
<owl:Class rdf:ID="Text">
  <owl:disjointWith>
    <owl:Class rdf:about="#Video"/>
  </owl:disjointWith>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Media"/>
  </rdfs:subClassOf>
 </owl:Class>
<owl:Class rdf:ID="Bag">
 <rdfs:subClassOf rdf:resource="#Complex"/> </owl:Class>
<owl:Class rdf:ID="Set">
 <rdfs:subClassOf rdf:resource="#Complex"/> </owl:Class>
<owl:Class rdf:ID="List">
 <rdfs:subClassOf rdf:resource="#Complex"/> </owl:Class>
<owl:Class rdf:ID="Graph">
 <rdfs:subClassOf rdf:resource="#Complex"/> </owl:Class>
</owl:Class>
<owl:class rdf:ID="Bag">
  <owl:disjointWith rdf:resource="Set"/>
  <owl:disjointWith rdf:resource="List"/>
  <owl:disjointWith rdf:resource="Graph"/>
</owl:Class>
```

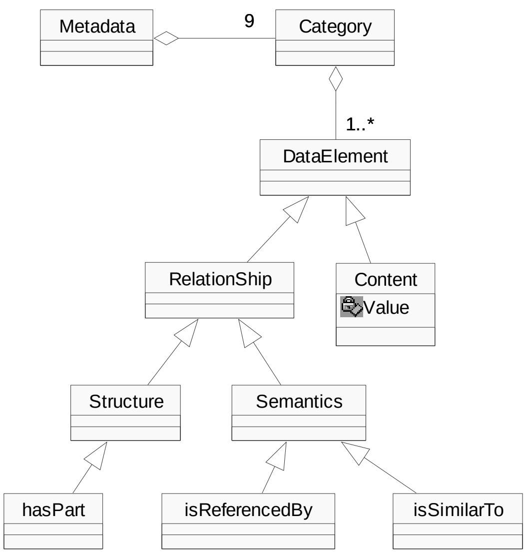

Figure 3. Model of LOM Metadata

A domain view (for example a thesaurus of the computer science) is illustrated by a hierarchy of terms, that guaranty there is no ambiguity in terms of understanding. The following extract of the classification of computer science built by ACM can be also translated into OWL:

D SOFTWARE
D.0 GENERAL
D.1 PROGRAMMING TECHNIQUES (E)
D.2 SOFTWARE ENGINEERING (K.6.3)
D.2.0 General (K.5.1)
D.2.1 Requirements/Specifications (D.3.1)
Elicitation methods (e.g., rapid prototyping, interviews, JAD) (NEW)
Languages
Methodologies (e.g., object-oriented, structured) (REVISED)
Tools
D.2.2 Design Tools and Techniques (REVISED)
D.2.3 Coding Tools and Techniques (REVISED)
Object-oriented programming (NEW)
D.2.4 Software/Program Verification (F.3.1) (REVISED)
Assertion checkers
Class invariants (NEW)
D.2.5 Testing and Debugging
Testing tools (e.g., data generators, coverage testing) (REVISED)
Tracing

This hierarchy of terms may be represented in OWL with subclass and equivalent relations.

```
<owl:Class rdf:ID="D">
   <rdfs:label>Software </rdfs:label></owl:Class>
<owl:Class rdf:ID="D2">
   <rdfs:label>Software engineering </rdfs:label></owl:Class>
   <rdfs:subClassOf rdf:resource="#D"/> </owl:Class>
   <owl:equivalentClass rdf:resource="#K.6.3"/> </owl:Class>
</owl:Class>
```

Each of the view was translated and refined by the OWL formalism.

2.2. Description of learning resources with the LOM semantics

The preliminary task consisted in translating the model of the LOM into a schema in OWL. We did it with the Protégé 2000 editor [8] in figure 4. We considered the Learning Object as a class, the categories and data elements as the properties of the Learning Object, and we explained the constraints on the space value. The following task consisted in classifying the concepts of our pedagogical ontology, integrating the two parts of ontologies, and specifying the properties and constraints:

```
<rdf:RDF
   xmlns:dc="http://purl.org/dc/elements/1.1/"
   xmlns:lom="http://ltsc.ieee.org/wg12/"
<owl:Ontology rdf:about=" ">
<owl:imports rdf:resource="file:/C:/BLD/Recherche/Articles/2003-2004/onto1-supelec.owl "/>
```

In our example, the concepts introduced in section 2.1: education, teaching_program, module, and learning_resources are considered as learning objects. The Learning_Object class is divided into two subclasses: Atomic_Object and Composite_Object. To express the level of granularity of the different learning objects, we used the following data elements of the LOM: General.Structure with value space in {atomic, collection, networked, hierarchical, linear} and General.AggregationLevel with value space in {1,2,3,4}. Thanks to OWL, we can easily specify that an Atomic_Object must values General.Structure = atomic, General.AggregationLevel = 1, or that a teaching program has value General.AggregationLevel > 2.

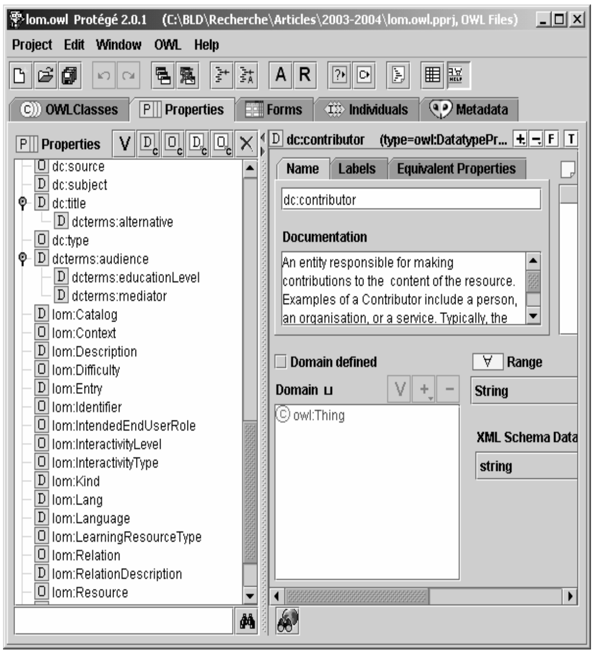

Figure 4. Edition of classes and properties

2.3. Description of the relationships between learning objects

Let's go further with the composition of learning objects which has been evoked in sections 2.1 and 2.2.
We define two categories of links among learning objects: the structural and the semantical ones. The structural links correspond to the logical structure of resources ("hasPart" and "sequence" links) whereas the semantical links correspond to the semantics of the associations among resources (besides the various relations defined in Dublin Core [1] we establish additional semantical links such as "summarization, reason, rephrase, negative, example" links). The structural links are particularly important because they participate in the reasoning mechanisms as we will see in the next section.

Figure 5. simplifies an example of the description of learning resources with two levels of representation: the schema level and the instance one. The schema level is described thanks to an ontology, the instance level is the knowledge base.

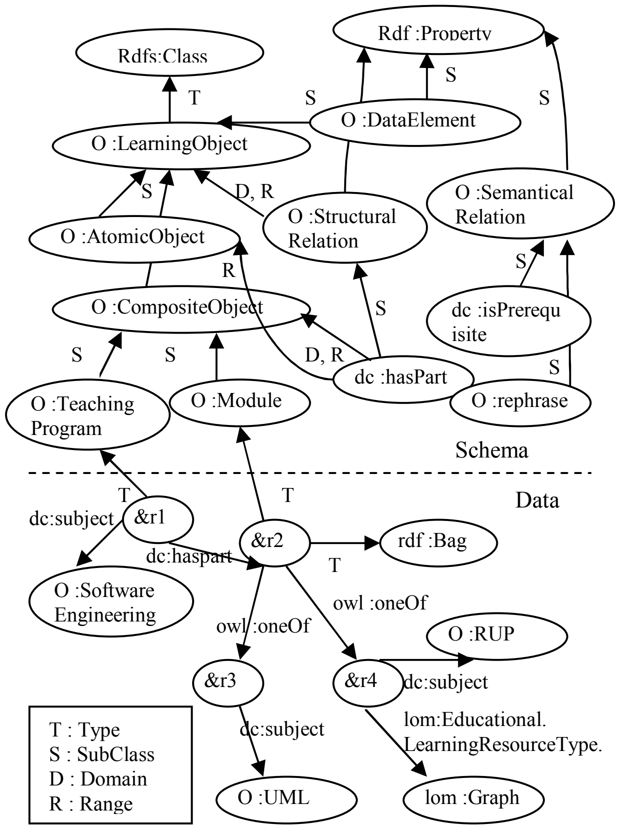

Figure 5. Schema and data example

3. Querying the pedagogical ontology

OWL-QL is a formal language and it is intended to be a candidate standard language for query-answering among semantic web computational agents. An OWL query contains a query pattern that specifies a collection of OWL sentences in which some URIrefs are considered to be variables. These answers provide bindings of URIrefs or literals to some of the variables in the query. For example, we could ask "Is there any course module whose the author is Mike?" The query can have the form: "(type ?c module) (author ?c mike)" where each query pattern is represented by a set of triples of the form (property subject object) and the variables are prefixed by the character "?". Inference mechanisms enable to deduce new information from some properties (symmetry, transitivity…).The OWL language allows us to specify property characteristics, which provide a powerful mechanism for reasoning about a property. The property can be exploited in the query part. For example consider the transitive property in OWL. If a property P is specified as transitive then for any x, y and z: P(x,y) and P(y,z) implies P(x,z). The structural relation "isPartOf" is transitive. This allows us to define simple query for asking about any learning object linked to a module by a direct or indirect "isPartOf " structural relation :

Q1 : (type ?x LearningObject) (type ?y module) (isPartOf ?x ?y)
From the example Figure 5., finding a graph resource illustrating a module or a course in software engineering is expressed with an OWL-QL like language as:
Q2: (type ?c Graph)(or (isPartOf ?c module) (isPartOf ?c course))
Finding all semantical links related to the &r1 resource:
Q3 : (type ?c SemanticalRelation) (rdf:Range ?c &r1)
As we illustrated in some of these examples, possibilities in expressing various powerful queries widen from schema and data queries, metadata, structural and semantic links, and reasoning forms.

4. Conclusion

In the building of a pedagogical ontology at Supélec, we distinguished two domains. The first one represented an education's organization. It has been enriched with the second ontology which represented a pedagogical content using standardized metadata (LOM). We showed the importance of the relations among learning objects to infer additional knowledge in the querying step. We use Protégé 2000 for our examples. We edited the entire LOM schema, the schema and instances of our pedagogical ontology. Protégé 2000 enabled us to detect and solve some inconsistencies in the classes and relations and therefore to validate our schema. It is possible to query some simple facts and to make some inferences. We gave some examples of queries in order to show the expressive power of a query language exploiting the benefit of the ontologies. OWL-QL syntax was chosen to show some examples of queries but we consider other query formalisms. We are currently implementing an OWL query language to test further our pedagogical ontology.

5. References

[1] Dublin Core Metadata Initiative. *Dublin Core Metadata Initiative*. http://purl.org/dc, 1995.
[2] IEEE Learning Technology Standards Committee. “Draft Standard for Learning Object Metadata” *http://ltsc.ieee.org/wg12/index.html*
[3] W3C Recommendation. “Resource Description Framework (RDF) Model and Syntax Specification”, www.w3.org/TR/REC-rdf-syntax
[4] T. R. Gruber. “A translation approach to portable ontologies”, *Knowledge Acquisition*, 5(2):199-220, 1993.
[5] World Wide Web Consortium. “ OWL Web Ontology Language Reference“, 10 February 2004. http://www.w3.org/TR/2004/REC-owl-ref-20040210/
[6] Richard Fikes, Pat Hayes, Ian Horrocks. OWL-QL: A Language for Deductive Query Answering on the Semantic Web. KSL Technical Report 03-14.
[7] Xpetal. http://www.langdale.com.au/styler/xpetal/
[8] Protégé 2000. http://protege.stanford.edu/