From XML to Semantic Web

Changqing Li and Tok Wang Ling

Department of Computer Science, National University of Singapore
{lichangq, lingtw}@comp.nus.edu.sg

Abstract. The present web is existing in the HTML and XML formats for persons to browse. Recently there is a trend towards the semantic web where the information can be can be processed and understood by agents. Most of the present research works focus on the translation from HTML to semantic web, but seldom on XML. In this paper, we design a method to translate XML to semantic web. It is known that ontologies play an important role in the semantic web, therefore firstly, we propose a genetic model to organize ontologies, and based on this model, we use three steps, viz. semantic translation, structural translation and schematic translation, to translate XML to semantic web. The aim of the paper is to facilitate the wide use of semantic web.

1 Introduction

Today, the web is increasingly moving to the semantic web [13], where the web information is annotated with concepts from sharing ontologies [10], thus the semantics of information can be understood and consumed by agents. Researchers [5, 7] have focused on how to annotate the HTML information. XML is another important format to store the current web information, but seldom researches are about how to translate XML to semantic web.

The rest of the paper is organized as follows. Section 2 reviews the related work, and Section 3 describes the preliminary and motivation of this paper. In Section 4, we propose a genetic model for ontology organization. Section 5 discusses the three steps of translations. The conclusion to this research is in Section 6.

2 Related Work

Resource Description Framework (RDF) [8] organizes information in a Subject-Verb-Object (SVO) (or Resource-Property-Resource triples) form, thus the RDF files can be processed semantically.

Some primitives are defined in RDF Schema (RDFS) [1] and the successors of RDFS: viz. DARPA Agent Markup Language (DAML) [11], Ontology Inference Layer (OIL) [6], DAML+OIL [4] and Web Ontology Language (OWL) [3]. OWL has three increasingly-expressive sublanguages: OWL Lite, OWL DL (Description Logic) and OWL Full. These languages all follow the RDF structure.

When the ontology languages are ready, the ontologies for different domains can be created. And based on the ontologies, the web information can be annotated for agents to process, which is today's semantic web. Most of the current techniques focus on the annotations of the information in HTML. SHOE Knowledge Annotator [5] and AeroDAML [7] are two tools to annotate HTML.

SHOE Knowledge Annotator is a manual tool. The user needs to manually select ontologies and concepts to annotate the HTML information, which is tedious. On the other hand, AeroDAML annotates HTML information based on the using frequency of an ontology concept and it is an automatic tool without any human interference. This tool uses a single predefined ontology to include all the concepts for different domains in. Therefore, when searching ontologies for annotations of HTML, all the concepts in this predefined ontology have to be traversed.

Another important format for the current web is XML. OntoParser [2] is a tool which only translates the structure of XML to satisfy the RDF structure.

3 Preliminaries

If an XML file confirms to an XML schema, it is said to be a valid XML file, otherwise it is an invalid XML file. For valid XML files, we can firstly translate their schemas to satisfy the semantic web requirement, then the valid XML files can be easily translated when confirming to the new translated schema, i.e. keep the changes of the XML schemas and update them into the XML files. For invalid XML files, we have to translate them individually.

3.1 ORA-SS Model

The XML Schema and DTD are two main schema definition languages for XML data. But they lack semantics. We employ the semantic rich model ORA-SS (Object-Relationship-Attribute Model for Semi-structured Data) [9] which distinguishes whether the relationship among the elements is binary or n-ary, and whether an attribute belongs to an element object class or to the relationship type among elements.

In Figure 1, student, course and part_time are treated as object classes. The id, name, contact_no, grade and position are treated as attributes. The filled circles are the object identifiers. The label "sc, 2, 3:8, 4:n" means: there is a binary relationship type named "sc", where one student may take 3 to 8 courses and one course should be taken by at least 4 students.

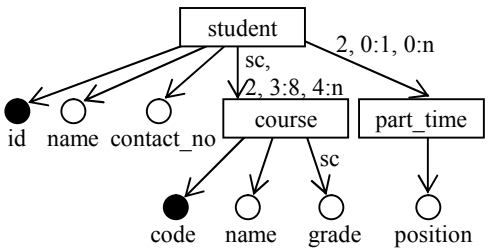

Fig. 1. The ORA-SS schema diagram.

The "sc" label near the edge from course to grade indicates that the grade is an attribute belonging to the relationship type "sc" rather than the object class course.

3.2 Motivating Example

Figure 1 shows that both the "student" and "course" have the "name" attribute, but they are different in semantics. It is easy for a person to distinguish the semantics of the two "name"s, but the agent will identify them as the same string if there is no semantic processing to the two "name"s.

The "part_time" and "position" show that the "student" may also be a part time employee, thus the semantics of "student_employee" is clearer than "student".

There is a relationship type "sc" between "student" and "course", but the semantics of "sc" is not clear. We do not know whether the student takes a course or drops a course. And the ORA-SS schema does not require the relationship to be an element name in the XML file, for example, the relationship type "sc" does not necessarily appear in a student XML file. That is to say, the semantics of the relationship between "student" and "course" is not clear.

We will address these problems in this paper.

4 A Genetic Model for Ontology Organization

We propose a genetic model to organize ontologies. This genetic model includes the following operators, viz. inheritance, block, atavism and mutation.

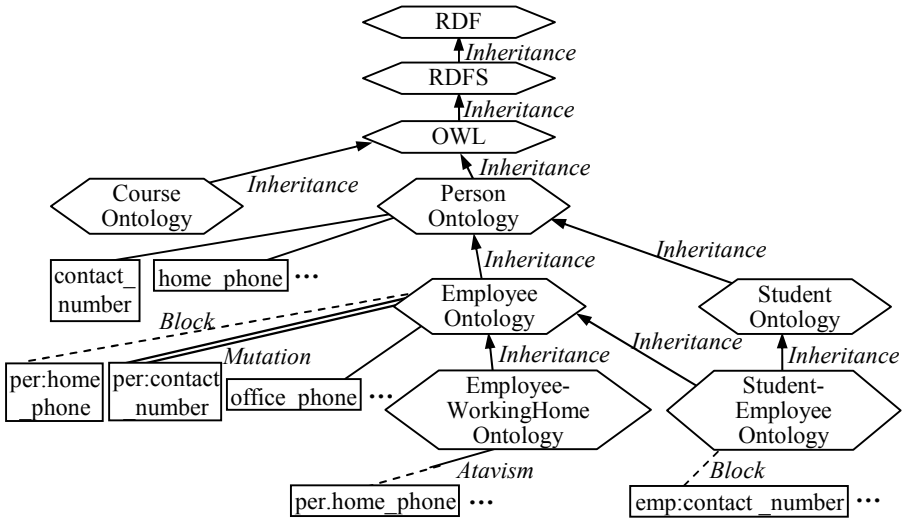

Fig. 2. Ontology hierarchy

In Figure 2, Ontologies reuse the primitives of ontology languages e.g. RDF, RDFS and OWL based on the inheritance operator. And lower level ontologies reuse the concepts of higher level ontologies, e.g. Employee Ontology inherits Person Ontology. The concept "home_phone" of Person Ontology is blocked by Employee Ontology because the home phone of an employee should not be public. The "contact_number" in Employee Ontology (refers to "office_phone") is a mutation of the "contact_number" in Person Ontology (refers to "home_phone").

The atavism operator is used to show that the "home_phone" of grandparent (Person) ontology blocked by parent (Employee) ontology is reused in the grandchild (EmployeeWorkingHome) ontology. The *atavism* operator in this model is absent in other inheritance mechanisms.

Furthermore, StudentEmployee ontology inherits both Student and Employee ontologies. StudentEmplyee Ontology blocks the "emp:contact_number" and inherits the "contact_number" from Student Ontology in the multiple inheritance. For the multiple inheritance, we assume that the concept from only one parent ontology is inherited, and the conflict concepts from all the other parent ontologies are blocked. Course Ontology does not inherit Person Ontology etc., but directly inherits OWL.

The "per", "emp" etc. in Figure 2 are namespaces [12] referring to Person and Employee ontologies etc.

5 The Translations

5.1 Semantic Translation

The *Semantic Translation (SemT)* from an XML file or schema to a semantic web file or schema in this paper means that the XML elements, attributes and values are replaced with concepts from ontologies.

Rule SemT 1 (Rule for that only one matched concept is returned from ontologies). The XML element, attribute or value is replaced with this only returned concept.

Rule SemT 2 to 4 are used for that more than one matched concepts are returned.

Rule SemT 2 (Rule for Multiple Inheritance and Block). If the child ontology inherits several parent ontologies, the concept from that unblocked parent ontology is selected for the replacement.

Rule SemT 3 (Rule for Atavism). If the concept of the grandparent or ancestor ontology is an atavism in the grandchild or descendant ontology, the concepts in the grandchild or descendant ontology are used for the replacement.

Rule SemT 4 (Rule for Mutation). If a concept in the parent ontology is a mutation in the child ontology, the concept in the child ontology is used for the replacement.

Example 1. If an XML is about student employee, the StudentEmplyee ontology is specified for search, and the ancestor ontologies of this specified ontology will be searched also. The "contact_number" from Student Ontology is used for replacement.

Rule SemT 5 (Rule for that no matched concept are returned from ontologies). If the element, attribute or value cannot be found in the ontologies, our system suggests adding new concepts into the ontologies (adding new concepts needs the confirmation from the domain expert).

Rule SemT 1 to 5 can be applied to the XML values also. The next two rules are for some special values.

Rule Sem 6 (Rule for Numbers). If the values in the XML are numbers, such as the contact_no "9876543", they need not be searched in ontologies.

Rule Sem 7 (Rule for Person Names). If the values in the XML are person names (or company names etc.), such as "John", they need not be searched in ontologies.

Since we use a top-down method for the replacement, the student "name" element will be replaced to "per:name" firstly ("per" is the namespace [12]), then later we know the value ("John") of element "per:name" is a person name. We do NOT need a person name dictionary for this case.

5.2 Structural Translation

Structural Translation (StrT) in this paper refers to the translation of an XML file or schema to a file or schema complying with the RDF structure, i.e. SVO format.

Rule StrT 1 (Rule for checking structure). For any path of the XML from the root to the leaf, if the nesting is not resource, property, resource, property, resource etc. interleaved, this XML does not satisfy the RDF structure.

Rule StrT 2 (Rule for modifying structure). If resources or properties are required to be inserted in the XML to satisfy the RDF structure, the resources or properties are searched in the ontology hierarchy based on the domain and range of properties (not based on name).

5.3 Schematic Translation

Schematic Translation (SchT) in this paper means that some features of the XML schema are translated to follow the RDF, RDFS and OWL languages.

Rule SchT 1 (Rule for ID and ID reference). For the object identifier of ORA-SS or the ID attribute of DTD, it will be translated to the "rdf:ID" (an identification primitive of RDF) and the value for the object identifier will be kept unchanged. We use the "rdf:resource" to refer to the referenced object.

Rule SchT 2 (Rule for default and fixed values). If the value of an attribute is a default or fixed value, it is kept unchanged.

Rule SchT 3 (Rule for order sensitive, composite and disjunctive attributes). The order sensitive attribute is translated to the "rdf:Seq", the composite attribute to the "rdf:Bag", and the disjunctive attribute to the "rdf:Alt".

Rule SchT 4 (Rule for cardinality). The cardinality to constraint the objects and attributes is kept unchanged after translation. Thus the structure information of the original XML schema can be kept.

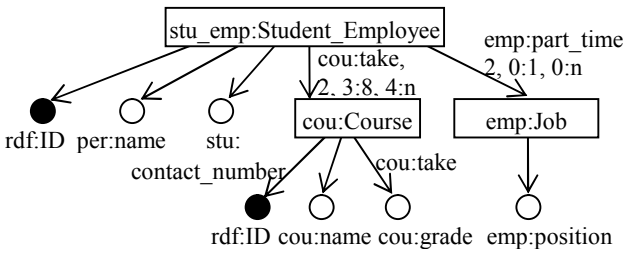

Fig. 3. The ORA-SS schema diagram after the three-step translations.

After the semantic, structural and schematic translations, the ORA-SS schema in Figure 1 becomes the schema in Figure 3 where "stu_emp", "per", "cou" and "emp" are namespaces to refer to Student_Employee, Person, Course and Employee ontologies, and "rdf" is the namespace to refer to the RDF ontology language. The "take", "Job" etc. are concepts defined in Course, Employee ontologies etc.

6 Conclusion

In this paper, we use three steps, viz. semantic translation, structural translation and schematic translation, to translate XML to semantic web. For a valid XML, their schemas are translated firstly, then the XML files confirming to the schemas can be translated easily, which improves the efficiency of translation. More important, we organize ontologies based on the genetic model. The searching to ontologies is only at several related paths of the genetic model, thus less concepts need to be traversed and less confused concepts will be returned, and the rules introduced in this paper make the semantics of the returned concepts clearer.

References

1. Dan Brickley and R.V. Guha. Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation 27 March 2000.
2. Avigdor Gal , Ami Eyal, Haggai Roitman, Hasan Jamil, Ateret Anaby-Tavor, and Giovanni Modica. OntoParser: an XML2RDF translator of OntoBuilder ontologies, OntoBuilder project. 2004.
3. Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider and Lynn Andrea Stein. OWL Web Ontology Language Reference.
4. Frank van Harmelen, Peter F. Patel-Schneider, and Ian Horrocks. Reference description of the DAML+OIL (March 2001) ontology markup language
5. Jeff Heflin and James Hendler. A Portrait of the Semantic Web in Action. IEEE Intelligent Systems, 16(2), 2001.
6. I. Horrocks, D. Fensel, J. Broekstra, S. Decker, M. Erdmann, C. Goble, F. van Harmelen, M. Klein, S. Staab, R. Studer, and E. Motta. The Ontology Inference Layer OIL.
7. Paul Kogut, and William Holmes. AeroDAML: Applying Information Extraction to Generate DAML Annotations from Web Pages. K-CAP 2001 Workshop, October 21, 2001.
8. Ora Lassila and Ralph R. Swick: Resource description framework (RDF). 1999.
9. Tok Wang Ling, Mong Li Lee, Gillian Dobbie. Semistructured Database Design, Springer, 2005
10. Robert Neches, Richard Fikes, Timothy W. Finin, Thomas R. Gruber, Ramesh Patil, Ted E. Senator, William R. Swartout: Enabling Technology for Knowledge Sharing. AI Magazine 12(3): 36-56 (1991)
11. Lynn Andrea Stein, Dan Connolly, and Deborah McGuinness. DAML Ontology language specification. October 2000
12. Namespaces in XML, World Wide Web Consortium 14-January-1999. http://www.w3.org/TR/REC-xml-names/
13. The SemanticWeb Homepage. http://www.semanticweb.org