Using Semantic Web Methods to Improve Information Resource Quality

John Barkley

Abstract

The quality of information resources can be improved using Semantic Web methods instead of, or in conjunction with XML (Extensible Markup Language) and relational database systems. With Semantic Web methods, more of the semantics of an information resource can be made explicit in a formal language while the representation remains amenable to fully automated validation and testing. This capability can improve the quality of information resources while reducing the need for developing specialized validation and testing tools for each application, thus, resulting in a return on investment in Semantic Web technology. For those familiar with XML and relational database systems, this paper introduces this capability with an example.

Introduction

Ensuring the high quality of an information resource requires two validation processes:

1. Data validation: validating data within the information resource according to specified constraints
2. Consistency validation: validating that the collection of constraint and query statements themselves are consistent with each other, i.e., there are no contradictions.

The collection of constraint and query statements constitutes information resource semantics.

Both XML[1] and relational database (RDB) systems[2] are rich with methods and tools for data validation. Semantic Web methods and tools not only have this capability, but are also capable of carrying out consistency validation in a fully automated manner.

This capability of Semantic Web methods for automated validation of information resources results from the work that has been done in Description Logic[3] theory on which the Semantic Web rests. Description Logic provides the mathematical foundation on which formal languages for describing information resources may be defined. OWL[4] is the standard description logic formal language of the Semantic Web. In the terminology of the Semantic Web, an information resource is called a “knowledge base”. The knowledge base consists of an information model called an “ontology” and its associated data.

A knowledge base described in OWL has the following characteristics:

- Whether the knowledge base is amenable to fully automated validation can readily be determined by a grammatical examination of the features of OWL used in the knowledge base representation. OWL DL[1] is the sublanguage of OWL which supports fully automated validation.

[1] OWL Full is the OWL language that has features of OWL that are not part of OWL DL. Knowledge bases represented in OWL Full are likely not amenable to fully automated validation (see section 1.3 of [5]). The

- Given that a knowledge base is amenable to fully automated validation, a "reasoner" performs such validation.

Specification of an information resource in a formal language with the certainty that both data validation and consistency validation can be achieved by fully automated processes can improve the quality of the resource and reduce the cost of development.

This paper[2] introduces this concept with an example to those familiar with XML and RDB. For those applications where OWL cannot be used exclusively, it can often be used in conjunction[3] with XML or RDB for designing and developing information resources. For those who may be unfamiliar with the Semantic Web, the example can be understood with minimal effort.

Example

This example is a variation of the textbook classic “Student Registration”. Constraints include:

- Each course has exactly one instructor
- Each course has at least three students.

In XML, one might have:

```
<?xml version="1.0"?>
<Student_Registration>
  <Courses>
    <Course name="PHY499" >
      <Instructor name="Nobel" />
      <Students>
        <Student name="Cornel" />
        <Student name="Hall" />
        <Student name="Phillips" />
      </Students>
    </Course>
  </Courses>
</Student_Registration>
```

---

sublanguage OWL DL is not as expressive as OWL Full. There are information resources that cannot be expressed in OWL (see section 5.4 of [6]). Often such resources can be partitioned into subparts which can be expressed in OWL DL. In these cases, overall resource quality can still be improved by applying Semantic Web fully automated tools to validating the partitions.

[2] Certain trade and company products are identified in this article to specify adequately the computer products used in the example. In no case, does such identification imply endorsement by the National Institute of Standards and Technology (NIST), nor does it imply that the products are necessarily the best available for the purpose.

[3] For more information about the relationships between XML, RDB, and Description Logic, see section 4.3 and chapter 16 of [3].

One might use XSLT or XQuery to validate the data in this information resource according to the constraints listed above. In order to do this, the constraints must be specified explicitly. For this example, XPath expressions[4] are used:

- Exactly one Instructor: count(//Course[@name="PHY499"]/Instructor)=1
- At least three Students: count(//Course[@name="PHY499"]/Students/Student)>=3

Furthermore, one may want to test queries such as "Find all Courses that do not have just the minimum number of Students" using the XPath expression: //Course[not(count(//Course/Students/Student)>= 3)]. The list of Courses returned by this expression is the subset of all Courses.

In a RDB system, one might have the TABLEs:[5]

course_schedule

course	instructor
PHY499	Nobel

student_registration

student	course
Cornell	PHY499
Hall	PHY499
Phillips	PHY499

For this RDB representation, the constraints "each course has exactly one instructor" and "each course has at least three students" can be expressed as CHECK constraints on the class_schedule and student_registration TABLEs. The following expression might be used to ensure that "each course has exactly one instructor":

```
CHECK(   (SELECT count(course_schedule.instructor) FROM course_schedule
              WHERE course_schedule.course = 'PHY499')  = 1 )
```

The following expression might be used to ensure that "each course has at least three students":

```
CHECK(   (SELECT count(student_registration.student) FROM student_registration
              WHERE student_registration.course = 'PHY499')  >= 3 )
```

The query "Find all Courses that do not have just the minimum number of Students" in SQL might be:

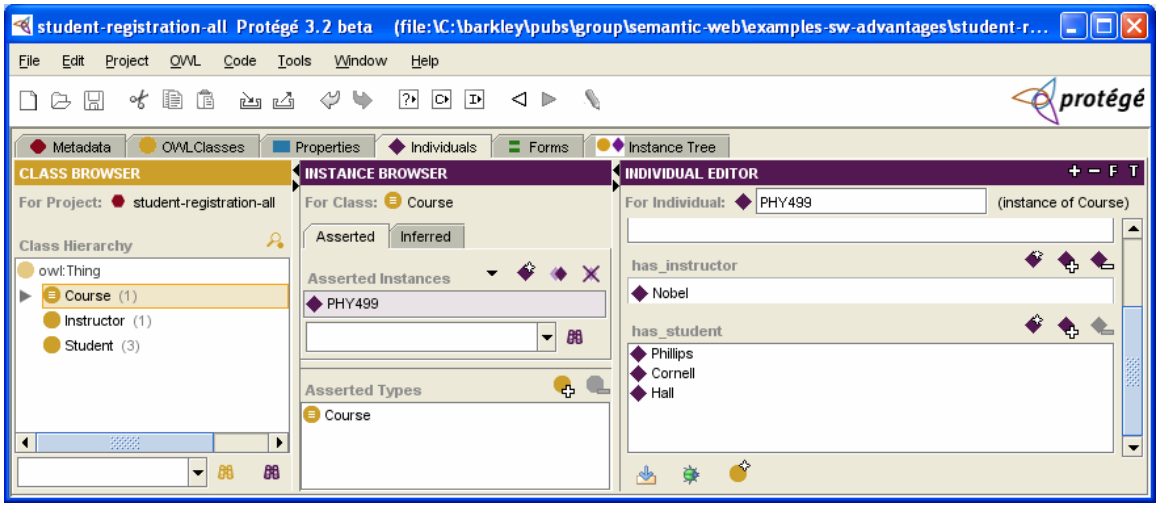
```
SELECT student_registration.course FROM student_registration
        GROUP BY student_registration.course HAVING NOT COUNT(*) >= 3
```

[4] XML Schema cardinality constraints can also be used.

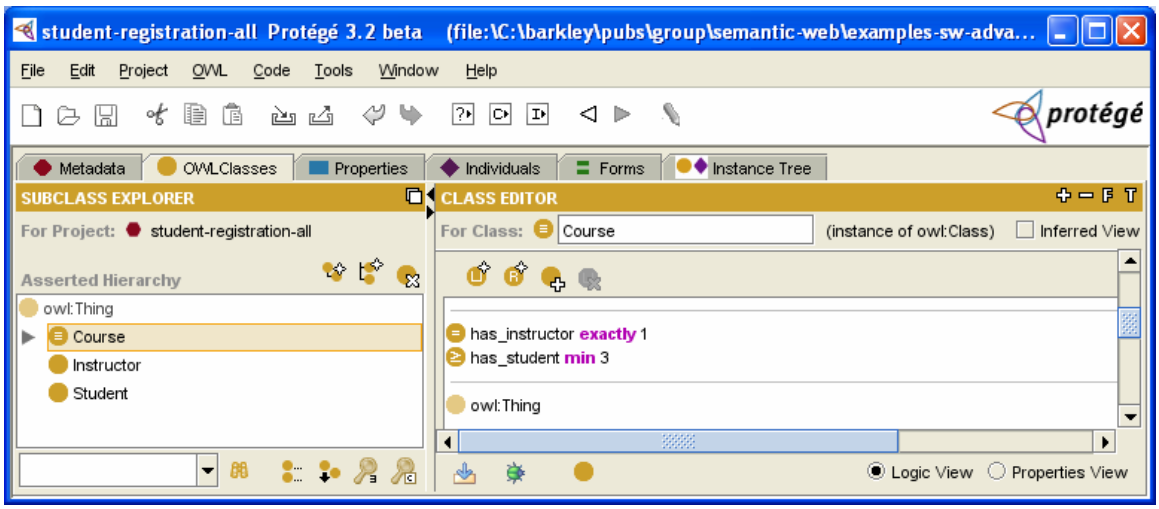
[5] For the example in RDB, it is assumed: field entries uniquely identify the student, instructor, and course; all fields are not required to be unique; all fields are non-null; and no columns are designated primary or secondary keys.

Having defined constraints and queries, how is their consistency to be assured, i.e., how can it be determined that this set of constraints and queries has no contradictions? With XML and RDB representations of information resources, this is typically done by manual inspection and semi-automated testing procedures often specialized for each application.

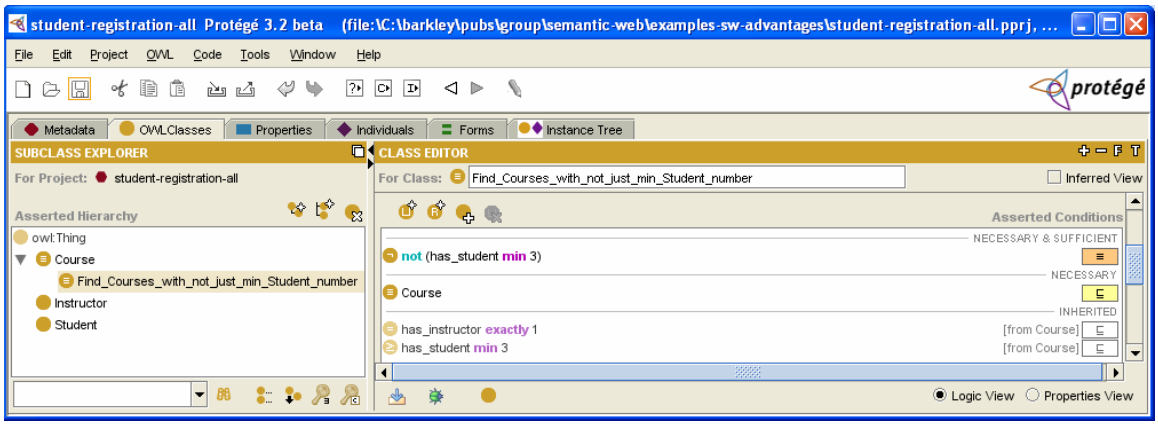
However, when an information resource is represented in OWL DL, constraints and queries can be validated for their consistency in a fully automated manner. Using the Semantic Web tool Protege[7], this information resource might appear as illustrated below:



There are three OWL Classes: Course, Instructor, and Student. There is one individual in the Class Course: PHY499 that has Instructor Nobel and three Students, i.e., Cornell, Hall, and Phillips.

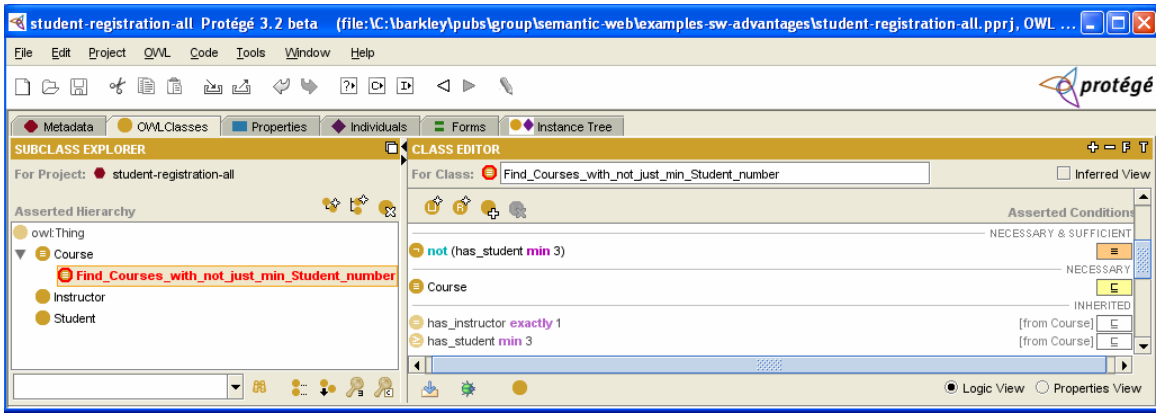


As shown in the display above, the Class Course is specified as: exactly one instructor per Course, and at least 3 Students in each Course.

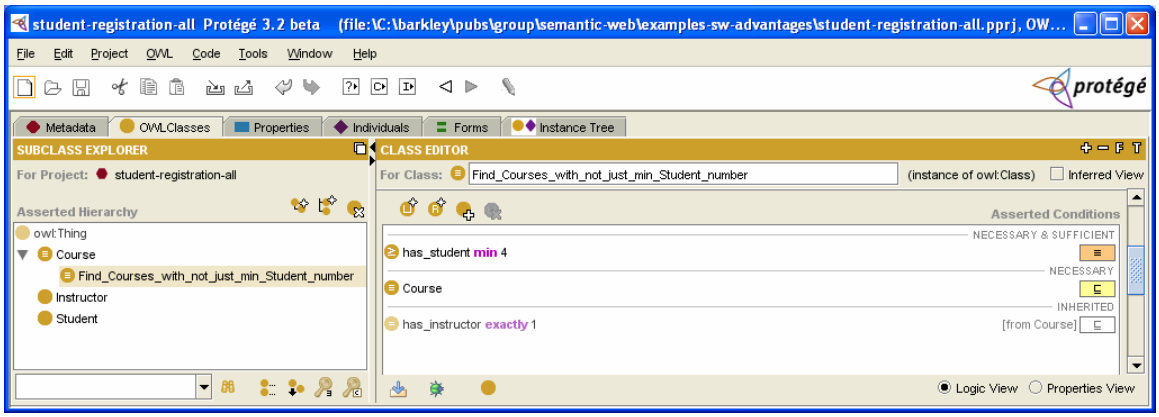


This display shows the specification of the Class Find_Courses_with_not_just_min_Student_number representing the query "Find all Courses that do not have just the minimum number of Students". This Class is specified as a subclass of Course since all courses which do not have just the minimum number of Students is a subset of all Courses.

In order to do data and consistency validation, a reasoner, in this case, Racer[8], is applied yielding the following result:



As shown in the above, the Class Find_Courses_with_not_just_min_Student_number has been marked in red, indicating an inconsistency. In this case, the reasoner identifies this Class is always empty. A brief examination locates the contradiction arising from its definition, the definition of the Class Course, and the fact that it is a subset of Course. The Class of all Courses which do not have just the minimum number of students is not the Class of all Courses whose count is not greater than or equal to 3 (i.e., less than 3). It is the Class of all Courses whose count is greater than or equal to 4. Semantic Web tools have revealed a consistency error in the specification of constraints and queries which resulted from a mistranslation of the English expression of the query - not an uncommon occurrence.



This display shows the results of the data and consistency validation once the error has been corrected.

Note that the XPath and SQL expressions for the query "Find all Courses that do not have just the minimum number of Students" were also incorrect. The XPath expression should be: //Course[count(//Course/Students/Student)>= 4]. The SQL query should be:
SELECT student_registration.course FROM student_registration
GROUP BY student_registration.course HAVING COUNT(*)>=4;

Summary

Ensuring the high quality of an information resource requires both data and consistency validation. Semantic Web methods and tools enable fully automated data and consistency validation for information resources. This improves the quality of information resources and reduces the need for developing specialized validation tools for each application. These goals can be achieved not only when developing information resources exclusively with Semantic Web methods, but also when using Semantic Web methods in conjunction with XML or RDB methods.

Information resource semantics represented in XML, RDB, or OWL can be explicitly stated in a form that enables automated processing. With XML and SQL, automated data validation can be achieved when constraints are specified in languages such as XPath and SQL. With OWL DL representations of information resources, both data and consistency validation are fully automated. It has been shown mathematically that for constraints and queries expressed in OWL DL, their consistency is decidable.

There are information resources that cannot be expressed in OWL DL, but can be expressed in OWL Full. For such resources, fully automated tools for data and consistency validation are likely not possible. However, such resources can often be partitioned into subparts which can be expressed in OWL DL. In these cases, overall information resource quality can still be improved by applying Semantic Web methods to partitions expressible in OWL DL.

References

1. http://www.w3.org/XML/
2. Ramakrishnan, R., Johannes Gehrke, J. Database Management Systems, 3rd Edition. (McGraw-Hill, 2002)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. The Description Logic Handbook: Theory, Implementation, and Application. (Cambridge University Press, 2003).
4. http://www.w3.org/2004/OWL/
5. http://www.w3.org/TR/owl-guide/
6. Horrocks, I., Patel-Schneider, P. F., van Harmelen, F. From SHIQ and RDF to OWL: The making of a web ontology language. J. of Web Semantics, 1(1):7-26, 2003.
7. http://protege.stanford.edu/
8. http://www.racer-systems.com/