Semantic Web and Linked Data Technologies for E-Learning

Tobias Gleißner

March 14, 2018

During classes at school or university we still pursue ancient methods of educating people. The classic ways of conveying information to the student are usually through books, lectures, and exercises. From a perspective of connecting pieces of information these kinds of media do not have much to offer. Usually lectures and also books are somehow sequential hence one cannot go back and repeat its content in-depth and with more detail or further explanation. In addition the lecture or book or even exercise is the same for every student thus lacking adapted content for the user's prerequisites. The ideal medium would address these issues by providing an environment that interconnects concepts and details to be learned and lets its user browse the information space as he or she pleases and needs. That means preparing and presenting information and exercises according to the needs of the individual user namely their current state of knowledge, certain strengths or weaknesses in their learning behaviour, interests, etc. This information about the user is further referred to as user context.

The question is: with today's technological advances can we do better in creating an individual and therefore effective learning experience compared to a traditional class room? The answer might very well be, yes we can. In the remainder of this article we will see a simple example of such individualized learning environment and explore the technologies powering those systems. This article is based on *Semantic Web Technologies for the Adaptive Web* [2] by Dolog and Nejdl.

The e-learning software Personal Reader [3] will serve as an introductory example to demonstrate a central method called link generation. Link generation means creating a suitable interlinking of pieces of domain information that is to be learned for presentation to the user and taking the user context into account during this process. In the case of Personal Reader these generated links can be observed on the left side of the window of figure 1. The most interesting ones here are *Generalizations* that result from putting the currently displayed information into some kind of taxonomy and *Details* that links to concepts of this learning session. The user context comes into play by placing green boxes next to certain generated links, creating a recommendation for the user on how to proceed based on prerequisites which the user covered in previous learning sessions. This is only a simple demonstration of link generation but it should be obvious how more powerful applications could be built employing this method.

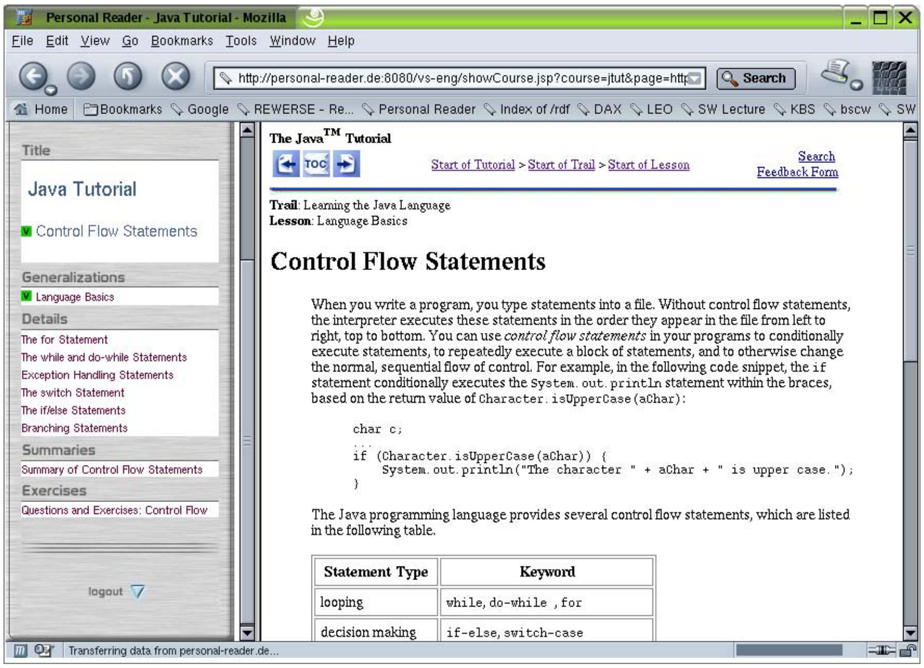
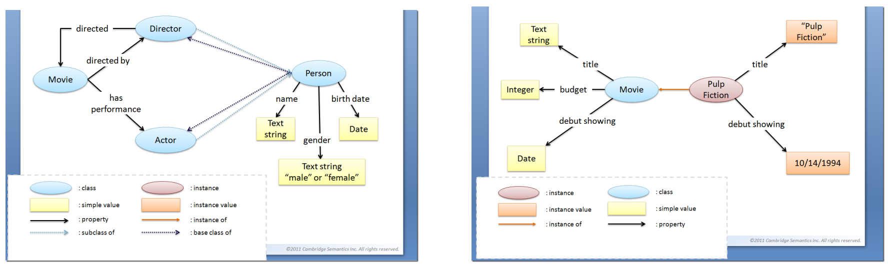


Figure 1: A Java tutorial with Personal Reader demonstrating link generation. Source: *Semantic Web Technologies for the Adaptive Web* [2]

Modelling Linked Information

To create such an e-learning application, the domain to be learned and the information about the user have to be organized. As a modelling tool for defining and connecting the concepts of the domain and the user context so-called ontologies have emerged. An ontology is a contolled i.e. defined vocabulary to describe concepts of the real world as classes of things, instances of classes and relationships between all components of an ontology. A distinction is made between two types of ontologies: the domain ontology which models a certain domain like the e-learning domain or a movie domain as demonstrated in figure 2 and the upper ontology which tries to define generalized concepts such that specialized ontologies can be reduced to it.

Usually an ontology of a domain is created by a domain expert e.g. an e-learning specialist. Subsequently an e-learning ontology can be supplemented with metadata that can be used to create suitable links for the presentation of the information. For the actual e-learning content the metadata is created by the author of the resource most of the time but of course there is room for



Classes, subclasses and properties in an ontology

Instances of classes in an ontology

Figure 2: A visualized movie ontology. Source: `wwww.cambridgesemantics.com`

machine learning techniques e.g. keyword extraction, etc. Data for the user context might either come from explicitly asking the user for information e.g. participating in a survey, rating tasks, etc. or by observing the behaviour of the user e.g. what contents he or she already learned, how successful he or she was doing the exercises etc.

After creating the ontology making it accessible to computer software is the next step in the endeavour building an e-learning application. There already exist standardized machine-readable formats such as the *Resource Description Framework* (RDF)[4] which is by far the most popular one and can be easily embedded into Websites since it has has the same underlying syntax (XML). RDF comes with predefined concepts for constructing ontologies (RDF Schema) and instantiating them (RDF). Ontologies are described by triples of subject, predicate, and object as demonstrated in figure 3. RDF has a popular extension called *Web Ontology Language* (OWL) [5] that overcomes some drawbacks of pure RDF such as the lack of a mechanism for specifying sufficient conditions of class membership or defining properties of properties among others. Also OWL comes with different flavors in terms of computational complexity: There is the decidable fragment of the classical first-order logic (FOL) that means one may not be able to model everything, but there is a guarantee that an algorithm is able to answer any question about the model in a fixed amount of time. Also OWL has an option for a more expressive logic sacrificing this guarantee in favour of better models.

```
<rdf:Description rdf:about="#Mephistopheles">
        <rdfs:type rdf:resource="#Cat" />
</rdf:Description>
<rdf:Description rdf:about="#Mephistopheles">
        <ownedBy rdf:resource="#Harry" />
</rdf:Description>
```

RDFS ontology description example: there are cats and persons. A cat might be owned by a person.

```
<rdf:Description rdf:about="#Mephistopheles">
        <rdfs:type rdf:resource="#Cat" />
</rdf:Description>
<rdf:Description rdf:about="#Mephistopheles">
        <ownedBy rdf:resource="#Harry" />
</rdf:Description>
```

RDF ontology instatiation example: Mephistopheles is a cat. Harry owns Mephistopheles.

Figure 3: An RDF demonstration

Reasoning in Ontologies

The machine-readable ontology of the domain of interest is now ready to be exploited by defining a query which in the e-learning domain is the learning goal. One wants to know how the e-learning environment should proceed i.e. makes suggestions to the user on what to do next or which resources should currently be displayed to the user for his or her benefit. For this assessment a reasoning engine is employed to deduct if a potential next step of the learning environment might be useful for the user e.g. does the user have the necessary prerequisites or should he or she better do another exercise on some other topic in order to achieve his or her top-level learning goal etc. Technically the machine-readable ontology is usually stored in a specialized database that has a reasoning engine built-in and comes with a query language like SPARQL [7] as a frontend to find out about the things one wants to know as shown in figure 4.

```
SELECT ?c
WHERE {
        "Harry" rdf:type ?c
}
```

Figure 4: Deducing the classes to which "Harry" belongs in the ontology of figure 3 using the query language SPARQL. The reasoning engine will find out that "Harry" has to be a person according to this ontology.

Aggregating the Technologies for the E-Learning domain

In the beginning a domain expert models his or her e-learning resources as an ontology at best reusing large parts of already existing ontologies to ensure interoperability. If resources from other sources modelled with different ontologies should be included they have to be converted to some upper ontology that can express the available information about all involved resources. Such upper ontologies for the e-learning domain already exist and one of them is presented in Dolog et al. [2]. The conversion is usually achieved by defining a transition from one ontology to the other or using machine learning techniques as demonstrated in *Linked education: interlinking educational resources and the web of data*[1]. The domain expert might use graphical tools like Protégé [6] that help in the process of designing an ontology and supplement it with metadata from the e-learning resources and automatically create a machine-readable output in RDF or OWL. An educationist analogously builds an ontology for the user context and combines it with the e-learning ontology. He or she establishes rules and queries concerned with pedagogical context within the resulting ontology. This will allow a reasoning engine output recommendations and useful supplements for the user from which a representation of the e-learning resources is created.

Hopefully this approach will lead to the development of new and better e-learning platforms that create an effective and individual learning experience at low cost. The available technologies and research suggest that this is entirely possible. Educational systems in different countries might very well benefit from this endeavour allowing teachers to shift their focus from lecturing crowded class-rooms to tutoring people if needed and creating high-quality learning material.

References

[1] Stefan Dietze, Hong Qing Yu, Daniela Giordano, Eleni Kaldoudi, Nikolas Dovrolis, and Davide Taibi. Linked education: interlinking educational resources and the web of data. In *The 27th ACM Symposium On Applied Computing (SAC-2012), Special Track on Semantic Web and Applications*, March 2012.

[2] Peter Dolog and Wolfgang Nejdl. *Semantic Web Technologies for the Adaptive Web*, pages 697–719. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[3] Nicola Henze. Personal readers: Personalized learning object readers for the semantic web. In *Proceedings of the 2005 Conference on Artificial Intelligence in Education: Supporting Learning Through Intelligent and Socially Informed Technology*, pages 274–281, Amsterdam, The Netherlands, The Netherlands, 2005. IOS Press.

[4] Ora Lassila and Ralph R. Swick. Resource Description Framework (RDF) Model and Syntax Specification. W3c recommendation, W3C, February 1999.

[5] Boris Motik, Bijan Parsia, and Peter F. Patel-Schneider. OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax. W3C recommendation, W3C, October 2009.

[6] protege.stanford.edu. Protégé, 2012.

[7] Eric Prud'hommeaux and Andy Seaborne. SPARQL Query Language for RDF. W3C Recommendation, January 2008. `http://www.w3.org/TR/rdf-sparql-query/`.