TEM Journal. Volume 7, Issue 3, Pages 693-697, ISSN 2217-8309, DOI: 10.18421/TEM73-29, August 2018.

Ontology Concept in Courses on Students

Daniela Orozova [1], Magdalina Todorova [2]

[1] Faculty of Computer Science and Engineering, Burgas Free University, 62 "San Stefano" str., Bulgaria
[2]Faculty of Mathematics and Informatics, Sofia University, Bulgaria 5"James Bourchier", blvd., Sofia, Bulgaria

***Abstract* – The article is based on our experience with introducing the concept "ontology" as a form of representing knowledge about the real world, or parts of it, in the courses in Computer Sciences in some Bulgarian universities. Ontologies are introduced as dictionary definitions in the context of knowledge sharing. The simplest case is when they consist of hierarchy of types, specifying classes, and relations among them. In contrast to the general dictionary, the ontology system is characterized by internal unity, logical interrelations, and logical conformity. The notion "ontology" is presented, in the courses of Computer Sciences, as an inseparable component of the Semantic Web.**

***Keywords* – Data representation, Ontology, OWL ontology, RDF, Reasoner.**

1. Introduction

The term "ontology" stems from philosophy and presents the nature of the things, which exist in real life. Contemporary philosophy literature uses it to denote a given system of categories, which results from a given view point on the world. Borrowed from philosophy, ontology is an inseparable part of metaphysics: the science about everything existing, its most general philosophy categories, such as existence, substance, cause, action and phenomenon. The task "How to describe, design and realize ontologies?" is of utmost importance. Solving this task is a subject of informatics.

Ontology is defined in informatics as an attempt for comprehensive and detailed formalization of a given area of knowledge via a conceptual scheme. This scheme usually consists of hierarchical data structures, containing respective objects, the relations between these, and the rules specific for this particular area. It is applied as a form of representation of knowledge about the real world or parts of it. The world consists of specific objects, which can be grouped in abstract classes on the basis of shared characteristics. Formally, an ontology consists of terms, organized in a taxonomy, their definitions and properties, as well as the related axioms and rules.

The article presents the realization of three main tasks, which are solved in the process of training. The first task is related to the description of the ontology nature. This is achieved by researching the origins and meaning of an ontology, and different types of ontologies are presented. Students are familiarized with the main components of ontology structure. The second task is related to presenting schemes which realize ontologies. This part of the education is based on examples, comparison between different sample ontologies, as well as analysis of each one of them. In order to do this, the components of an OWL ontology are introduced. The third task is developing a sample example of an OWL ontology. This is done by using the Protégé software.

In addition, analyses of some issues arisen during the training are also presented in the article. Possible ides of improving the results of education in the field are discussed.

2. Introducing ontologies in courses for students

The education in this field for students in the area of Informatics and Computer Sciences is realized mainly through the disciplines related to knowledge

DOI: 10.18421/TEM73-29
https://dx.doi.org/10.18421/TEM73-29

Corresponding author: Magdalina Todorova,
Faculty of Mathematics and Informatics, Sofia University, Bulgaria 5"James Bourchier", blvd., Sofia, Bulgaria
Email: todorova_magda@hotmail.com

Received: 24 April 2018.
Accepted: 10 August 2018.
Published: 27 August 2018.

representation, in the 3rd or 4th semester, depending on the specialty.

Introducing ontologies is started by presenting their classification based on different classification properties. Some authors in the field view ontologies as oriented towards notions (concepts) [1, 2], others – as oriented towards the objects in the subject area [3, 4]. We have utilized a classification, according to which the ontologies are divided into: general ontologies, subject area oriented ontologies, ontologies oriented towards a specific task, and applied ontologies [5].

General ontologies (the ontologies of the highest rank) are subject independent, containing general categories (concepts) such as space, time, object, event, action, quantity, dimension, etc., which do not depend on the task or the area. These ontologies have to be applicable to each specific field (which is what distinguishes them from the specialized ontologies).

Subject-oriented and task-oriented ontologies describe a respective glossary, related to, respectively, the subject area (medicine, trade, etc.), or to the particular task or activity (diagnostics, sale, etc.). This glossary is based on the specialization of terms, introduced in a general ontology.

Applied ontologies describe concepts, which depend on both the particular subject area and the tasks to be solved. The concepts of these ontologies often correspond to the roles, which the objects play in the subject area when executing a given activity.

It can be concluded that ontologies are used in describing reality and are related to a specific field (domain of interest). They are developed in order to separate the knowledge about the domain from the operative knowledge, with the aim to analyze and to reuse knowledge [6].

Ontologies are commonly applied for more precise description of web resources in order the latter to become more easily machine processed.

The education continues by studying the following:

- Ontology structure;
- Methodology for designing an ontology;
- Studying Web Ontology Language (OWL) and its application for designing an ontology;
- Using Reasoner for working with ontologies.

2.1. Ontology structure

There are two distinguished components in the ontology structure: names of existing concepts, and relations in the domain. For example:

- “Elephant” is a concept, whose members are a kind of an animal;
- “Herbivores” is a concept, whose members are animals feeding only on plants or parts of plants.

A number of restrictions can be imposed on the domains. For example, “The adult elephants weigh more than 2000 kg.”, “There is no animal that is both herbivore and carnivore”.

An ontology, together with a set of given instances of classes forms a knowledge base.

The design of an ontology goes through the following phases [7]:

a) Definition of the classes in the ontology;

b) Classification of the classes into an hierarchy;

c) Definition of properties and describing the possible values they can have;

d) Filling in concrete values of the properties in order to receive particular instances.

Ontology design is an iterative process. There is no single way of developing an ontology. The main rule is the ontology concepts to be close to the objects and the relations in the given domain.

The methodology for creating an ontology presupposes defining the goals and area of applicability. Therefore, it has to be identified in advance: what is the purpose of designing this ontology, what types of questions should be answered through it, how it will be used and supported.

2.2. Methodology of ontology design

One of the approaches used in the process of training is teaching through examples. During the classes, dedicated to introducing the ontology notion, the students create a particular sample, following the steps below:

1. Defining the ontology domain and goals;
2. Analysis and discussion of existing reusable ontologies;
3. Listing the important terms of the ontology;
4. Defining the classes and the class hierarchy;
5. Defining the class slots;
6. Defining the slots aspects;
7. Creating class instances.

When designing an ontology, the following main principles are followed:

- Formalization – describing the subject area in unified and strictly defined samples (terms/notions, models, etc.);
- Introduction of a limited set of basic notions, based on which the rest of the (complex) notions are created;
- Completeness and consistency of the used notions;

• Applying tools for creating ontologies, which support editing, visualization, documentation, import and export of ontologies.

2.3. Applying OWL for creating ontology structure

In order to create the ontology, we use OWL, a language for defining Web ontologies. An ontology designed by OWL will be referred to as OWL ontology hereafter. An OWL ontology can include describing classes, class properties, and their instances.

The components of an ontology, created on OWL, are briefly described below:

- Individuals: individuals, or objects, are the instances of a class;

- Classes: classes are a concrete representation of the concepts;

- Attributes (properties, slots): characteristics of objects and classes, also known as roles in descriptive logic;

- Relations: the ways of connecting classes and objects;

- Function terms: complex structures, formed by concrete relations, which can be used instead of an individual member of a statement;

- Restrictions: a formally defined description of what must be true in a given statement in order it to be accepted as an input;

- Rules: if-then expressions, describing logical conclusion, which can be drawn from a statement in a particular form;

- Axioms: statements (including rules) in a logic form, which gathered together cover all the theory which is described by the ontology in its domain of application;

- Events: give the changes of attributes or relations.

The specification of OWL defines how logical conclusions are drawn, i.e. how facts are created, which are not explicitly presented in the ontology, but results from its semantics. These conclusions can be based on one or more documents, which are combined by OWL tools. Knowledge of the ontologies is applied via conclusions. Examples of possible conclusions:

- X is the author of Y; therefore Y is written by X

- Cars are a type of vehicles; Vehicles have >= 2 wheels; therefore Cars have >= 2 wheels.

An OWL ontology is created for the purposes of education in this field, by using the Protégé 4.3 software.

Protégé (developed by a team of experts in medical informatics at Stanford University [8]) is a Java-based freeware, designed to create ontologies. It consists of an editor of ontologies, which allows creating a hierarchical structure of a general catalogue. Based on it, forms for knowledge acquisition can be generated. It features graphic interface, capabilities of providing references and examples. It supports OWL language for presenting the ontology structure.

The students perform the following concrete actions using the Protégé 4.3 software:

1. Creating classes. Having created classes, e.g. Student and Teacher, we can make them disjoint – this means that an object cannot be an instance of more than one of these two classes.

2. Creating a class hierarchy. Creating subclasses for Teacher: Lectures and Assistants.

3. Defining OWL Properties, which are connections of individuals. The two main property types are the following:

- Object properties – connect an individual with an individual;

- Data properties – connect individual with a value of a data type (XML or RDF literal).

The next step is introducing OWL characteristics of object properties. The description below features A, B and C to denote individuals, and P property.

Characteristics of the object properties:

• Each object property can have an inverse property. If a property connects A with B, what follows is that its inverse property connects B with A.

• Functional property – for a given individual, there is no more than one individual connected by this property.

Example: If hasBirthMother is functional and
Ivan hasBirthMother Maria
and Ivan hasBirthMother Neli => Maria ≡ Neli

• If a Property is inverse functional, what follows is that the inverse property is functional.

Example: Neli isBirthMotherOf Ivan
and Maria isBirthMotherOf Ivan => Maria ≡ Neli

• If P is transitive and P connects A with B and B with C => P connects A with C. If a property is transitive, what follows is that its inverse is also transitive. If a property is transitive, it cannot be functional.

• If P connects A with B and P is symmetrical, what follows is that P connects B with A.

• If P connects A with B and P is asymmetrical, what follows is than the individual B cannot connect to the individual A via P.

• P is called reflexive if it connects the individual A with itself.

• If P is irreflexive, it can be described as connecting A with B, where A and B are not one and the same individual.

• Property Domains and Ranges connect individuals of the domain with individuals of the range.

• Property Restriction is a class of individuals, which is described or identified through the connections, in which these individuals are. It describes an anonymous (nameless) class.

2.4. Using Reasoner (classifiers)

One of the key features of the ontologies described via OWL is that they can be processed by reasoner.

One of the main services offered by reasoner is to test if a class is a subclass of another or not (subsumption testing).

Another built-in capability of reasoner is to check the ontology for consistency. It checks if the class can or cannot have instances.

The next step in the training is defining of primitive and defined classes based on necessary and sufficient conditions:

• Classes which have only necessary conditions are called primitive. This means that if something is a member of a given class, it must satisfy given conditions. However, we cannot say that if a given individual satisfies given necessary conditions, it is a member of a given class, if we have not defined a sufficient condition for this.

• Classes which have at least one set of necessary and sufficient conditions are called defined classes.

3. Analysis of the education and the results

In order to improve students' understanding of ontologies and overcoming the difficulties, we apply different scenarios in the process of e-learning.

• A possible scenario consists of designing an ontology system in the subject area of the course and immediately applying this as an additional resource for the e-learning training. The dynamic relation between such a resource and the particular e-learning system can be used to collect data on the difficulties the students encounter in the given discipline on topic, based on test tasks and questions.

• During the training, it may be needed to improve the ontology system: adding new notions, relations, or attributes. A scenario could contain including the trainees in developing the ontology in use for the given subject area. For example, the language OWL allows for initial creation of an ontology with basic concepts in the researched field and gaining inductive knowledge on the discipline. Such an approach can be used for designing new or developing existing ontologies for different courses.

• The education is based on Project-Based Learning as a pedagogical model of interdisciplinary activities, related to real life problems [9, 10, 11, 12]. The strategic goal we have includes involving students in researching and developing skills, such as: doing research and practice in a particular subject area, identifying the stages of project design; planning the activities timing; keeping deadlines; self-evaluation; participation in discussions in the field, defending opinions and arguing against ideas.

• An attractive area of learning and education is that of cloud computing. Federated clouds share resources across institutions like established Nebula cloud at our university described in [13]. The knowledge on security for distributed systems and especially for clouds can be represented in OWL. Such an effort is initiated with PhD students in above mentioned cloud project.

4. Conclusion

The article presents education in the field of Ontologies as a tool for knowledge presentation, as practiced in two Bulgarian universities. The training supports authors' belief that new approaches for problem solving must be found and introduced, and new techniques, tools and design environments must be applied, at each stage of student education in the field of Informatics and Computer Sciences.

References

[1]. Todorova, C., & Stefanov, K. (2006) Selection and use of domain ontologies in Learning Networks for Lifelong Competence Development. *Learning Networks for Lifelong Competence Development* (Eds. R. Koper and K. Stefanov), Sofia, Bulgaria, 11-17, ISBN 9549174344.

[2]. Stefanov, K., & Yordanova, K. (2003). Computing ontology creation. In *International Congress MASSEE (MAthematical Society of South-Eeast Europe) 2003* (pp. 6-pages).

[3]. Georgiev, A., Grigorov, A., Bontchev, B., Boytchev, P., Stefanov, K., Bahreini, K., ... & Hollins, P. (2016, September). The RAGE software asset model and metadata model. In *Joint International Conference on Serious Games* (pp. 191-203). Springer, Cham.

[4]. Boytchev, P., Bocconi, S., & Stefanov, K. (2010) An Ontological Approach for Multicultural Digital Repositories, Second International Conference S3T, September 11-12, 2010, Varna, Bulgaria, 97-105, ISBN 978-954-9526-71-4.

[5]. Gruber, T.R. (1993). Translation Approach to Portable Ontologies, *Knowledge Acquisition*, 5, 199-220.

[6]. Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *The knowledge engineering review*, *11*(2), 93-136.

[7]. Gomez-Perez, A., Fernando-Lopez, M., & Corcho, O. (2004). *Ontological Engineering with examples from the areas of Knowledge management, e-Commerce and the Semantic Web*. 2nd. Ed. London, Springer-Verlag.

[8]. The Protégé Ontology Editor and Knowledge Acquisition System, Retrived from: http://protege.stanford.edu/

[9]. Todorova, M., & Orozova, D. (2016). How to Build up Contemporary Computer Science Specialists – Formal Methods of Verification and Synthsis of Programs in Introduction Courses on Programming, *Proceedings of 9-th Annual International Conference of Education*, Research and Innovation, Seville, ISBN: 978-84-617-5895-1, 4249-4256.

[10]. Kaloyanova, K. (2014) An Implementation of the Project Approach in Teaching Information Systems Courses, 8th International Technology, Education and Development Conference,Valencia, Spain, 7090-7096.

[11]. Jecheva, V., & Orozova, D. (2011). Ontology-Based Electronic Test Results Evaluation. In *Third International Conference on Software, Services and Semantic Technologies S3T 2011* (pp. 213-214). Springer, Berlin, Heidelberg.

[12]. Orozova, D., & Atanassov, K. (2012). Generalized Net Model of the Process of Selection and Usage of an Intelligent e-Learning System, *Comptes rendus de l'Académie bulgare des Sciences*, *65*(5), 591-598.

[13]. Korenkov, V., Kutovskiy, N., Balashov, V., Dimitrov, V., Hristova, R., Kouzmov, K., & Hristov, S. (2017) Clouds of JINR, University of Sofia and INRNE join together, Proc. of the XXVI International Symposium on Nuclear Electronics & Computing (NEC'2017) Becici, Budva, Montenegro, September 25-29, 2017.
Retrived from: http://ceur-ws.org/Vol-2023/122-128-paper-19.pdf.