JECET; June 2014-August 2014; Vol.3.No.3, 1401-1406. E-ISSN: 2278–179X

Journal of Environmental Science, Computer Science and Engineering & Technology

An International Peer Review E-3 Journal of Sciences and Technology

Available online at www.jecet.org

Section B: Computer Science

Research Article

Development of Ontology for Software Engineering Course: A Recommendation

J. Meenakumari*and V.N. Jinesh

Department of Computer Science and Applications, the Oxford College of Science, Bangalore, India

Received: 17 May 2014**; Revised:** 29 May 2014; **Accepted:** 2 June 2014

Abstract: The core objective of ontological representation is to provide better domain knowledge in specific areas. Ontologies are being developed to share common understanding regarding the structure of information among the people concerned in particular domain. This paper explains about the ways in which ontological representation can be inherited into the teaching-learning process of Software Engineering course. The process starts with concept design followed by concepts hierarchy, setting relations, and constraints in the software engineering domain.

Keywords: Ontology, domain knowledge, teaching-learning, concept design, concept hierarchy.

INTRODUCTION

Ontology provides definitional information about entities in a specific domain to get more clarity on to the required topic. In the computing context, ontology is a framework for representing concepts, things, or ideas about things and their relationships. Ontology have a wide range of potential benefits and applications in higher education sector which includes information sharing, enhanced content representation, providing frameworks for learning object reuse, and enabling intelligent and personalized decision support system etc. Ontology-based representation of various subject are increasingly becoming popular due to the inherent advantages.

Ontological representation can be inherited into various tasks of teaching-learning processes like developing instructional resource, delivery of the course content, assessment design, and for any other information support system. Ontology-based teaching-learning process facilitates to enforce overall standards in the entirety of the process thereby bringing in lot of flexibility and transparency. Ontology represents a multi-dimensional map of a domain of learning. At the top level, the core concepts and principles that are essential to competency in a domain are specified. Drilling down from these core concepts are the supplementary knowledge and skills which are expected to develop as they progress toward competency. Ontology thus represents a multi-dimensional depiction of how learning is defined and the method through which knowledge and skills are developed. It also provides an organizational scheme for integrating expectations, mapping progress, and coordinating learning resources and other support system.

THEORETICAL BACKGROUND

Ontology has been developed in the artificial intelligence community to describe a variety of domains, and has been suggested as a mechanism to provide applications with domain knowledge and to facilitate the sharing of information[1-4]. Ontology is a formal, explicit specification of a shared conceptualization [5]. Ontology is typically defined as an abstract model of a domain of interest with a formal semantics. In the simplest case, ontology consist of a set of concepts or classes which are relevant for the domain of interest as well as a set of relations defined on these concepts. It serves as semantic reference for users or applications that accept to align their interpretation of the semantics of their data to the interpretation stored in the ontology[6]. As a new kind of knowledge organization tool, ontology has attracted more and more attention. Ontology has been widely used in many fields, such as knowledge representation, knowledge sharing, knowledge integration, knowledge reuse, information retrieval, and so on. Hence the development of ontology is seriously impeded[5]. When ontology is applied to specific field, it refers as domain ontology and is the specification of a particular domain conceptualization. The development of ontology is normally carried out by a team of people, such as domain experts, ontological engineers and pedagogues. The main reasons for developing ontology are to share a common understanding of the structure of information among people or software agents, to enable reuse of domain knowledge.

Developing Course Ontology: The field of education is one of the first, where understanding ontology as a cognitive tool came around. In many respects it was due to the wide spread of the constructivist paradigm of learning and the broad use of such knowledge technologies as concept maps, mind maps and others for learning purposes[7].

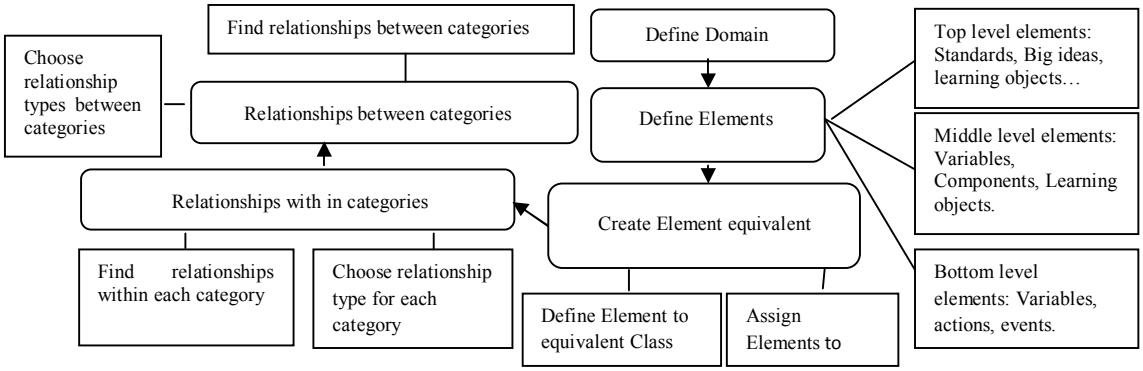

Figure 1: Illustrating the procedure to create ontology[8].

Ontologies have emerged as a central technique[9] Daconta *et al.*[9] for knowledge integration, sharing and reuse. There is a long history of the application of knowledge engineering techniques in educational technology more recently, ontology have attracted widespread attentionDevedžić[10]; Sampson *et al.*[11] Ontology helps us to make the knowledge that is represented in learning content explicit. Knowledge is central in learning; learners consume content to acquire knowledge. Knowledge is also important for the content developer, as content can be an elaboration of explicitly represented knowledge, and therefore a central ingredient for the development of content ontologies Mizoguchi & Bourdeau[12]. The foremost goal in this paper is to determine the most suitable ontological modelling notation for Software Engineering course.

PROBLEM STATEMENT AND SCOPE

Software Engineering is considered to be one of the most difficult subjects to learn by students because of its vast and wide range of domain contents. There is a huge gap in setting up the relationship between the concepts. Hence there is a need to present the relations between the concepts and the course materials and subject related contents in a proper way.

Design of Course ontology for Software Engineering: The first activity involves classifying or categorizing a set of things using a hierarchical structure, with the most general category as the root of the tree [14]. Each node, including the root node, is an information entity that represents some object in the subject that is being modelled. Each link between two nodes represents a “sub classification-of” relation or a “super classification-of” relationship. Thus ontology in software Engineering subject can be used to support course and assessment design to bring more clarity into the subject domain. One property of ontology is that the importance of a node is conveyed by how “connected” the node is, the set of concepts and how they relate to each other. For example, if the purpose of the assessment or course is to measure what students understand about the subject, it would be reasonable to assess whether students understand the fundamental elements and operations in the domain.

A second function of the ontology is to help to decide the grain size of the assessment or course work. Presumably, the cognitive demand of the assessment should be appropriate to the relative position of the concept in the domain[13]. Depending on the structure of the domain; instructional sequences can be derived from the nature of the links. Content can be “attached” to concepts in the ontology in almost any way. For example, the definition of a concept could include properties that tie in instructional content (e.g., as URLs to content; as filenames; as verbal descriptions). The richness of the representation is determined by the instructional requirements as there is no a prior limitation to the representation. The design of the ontology and class structure is quite flexible. These structures are elaborated, tagged, and embedded in the ontology. They can be accessed by commands or queries that can show relevant instruction. This hierarchical structure will help instructors to deliver an effective lecture by understanding the related instruction fields and prepare accordingly with the help of ontologically arranged materials. A process of design in the software engineering ontology refers to the process of design concepts, concepts hierarchy, relations, and constraints in the software engineering domain. Sources of software engineering knowledge are from the software engineering textbooks upon which the design is focused. The software engineering ontology contains concepts and relations. The ontological steps that could help to understand the hierarchical activities of all the task mentioned bellow are provided in the diagrammatic representation. The main tasks in Software Engineering are

1. Market survey and Domain Study
2. Feasibility study
3. System and Requirement analysis
4. Design
5. Implementation
6. Testing
7. Post Implementation.

Figure 2: An ontology for Software Engineering.

Market survey and domain analysis: Market survey and domain analysis is the initial step that has to be followed and after this step one can decide whether the proposed project is possible to develop or not. In this there are various tasks like SRS preparation, use case definition etc. Taxonomy in this task will contain major components required for the above tasks.

Feasibility Study and estimation: The feasibility study includes a detailed study on technological feasibility together with financial feasibility. Estimation will give a detailed report about financial as well as resource cost of the required product in the current context. Taxonomy in this task will contain major components such as statistical tools to perform various feasibility analyses such as cost estimation and financial feasibility, resource requirement estimation.

Analysis: Analysis involves the identification of functional and non-functional requirements pertaining to the specific domain. A thorough investigation of what needs to be developed is determined in this phase.

Design: The major components of this task are modular design and system design. This task will provide a platform for the designer to showcase their innovative skills and transform the items identified under analysis to design. Based on the design solution any object oriented design tool can be selected appropriately.

Implementation: This task involves the actual implementation of the project as per the end user requirements. The major taxonomy in this task is the coding standards that are to be followed by the developer.

Testing: Once the implementation done the developer has to validate and verify the product through different testing methodologies applicable time to time.

Post implementation: After confirming the performance of product through validation and verification, the product is officially released to the customer. A good user manual, technical support and maintenance will also be extended to the customer. While domain models are clearly driving the core of every software system, their importance in current SE processes decreases after the analysis phase. The core purpose of ontologies is to provide the formal descriptions of a domain and also encourages a broader usage of process model throughout the SDLC life cycle [15].

CONCLUSION

This paper presented conceptual ontology for the Software Engineering course in the context of course delivery and assessment. This framework would definitely enhance the subject understanding and assessment activities there by facilitating better teaching-learning process.

REFERERENCES

1. J. Chen, Q.M. Zhu, and Z.X. Gong, (2007),*Overview of Ontology- Based Information Extraction, Computer Technology And Development*, **17**(10), 84–91.
2. F. Gu, C.G. Cao, Y.F. Sui, and W. Tian, (2004)Domain-Specific Ontology of Botany, *Journal of Computer Science and Technology*, **19**(2), 238–248.
3. H.C. Jiang, D.L. Wang, and D.Z. Zhang, (2008) Approach of Chinese Medicine Knowledge Acquisition Based on Domain Ontology, *Computer Engineering*, **34**(12), 16–18, 21.
4. J.D. Yu, X.Y. Li, and X.Z. Fan, (2008) Design and Implementation of Domain Ontology for Information Extraction, *Journal of University of Electronic Science and Technology of China*, **37**(5), 746–749.

5. T.R. Gruber.(1993),Towards principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, **43**, 5-6, 907-928.
6. Z.H. Deng, S.W. Tang, M. Zhang, D.Q. Yang, and J.Chen, (2002), Overview of Ontology, *Acta Scientiarum Naturalium Universitatis Pekinensis*, **38**(5), 229–231.
7. Darina Dicheva 1, Sergey Sosnovsky2, Tatiana Gavrilova3, Peter Brusilovsky2. *Ontological Web Portal for Educational Ontologies*.
8. Sinead Boyce and Claus Pahl (2007), Developing Domain Ontology for Course content, *Educational technology and society*, 10(3), 275-288.
9. Daconta *et al*. The Semantic Web – a Guide to the Future of XML, Web Services, and Knowledge Management, Indianapolis, USA: Wiley & Sons2003.
10. V.Devedžić, Semantic Web and Education, Berlin: Springer-Verlag.2006
11. D.G.Sampson, M.D. Lytras, G. Wagner, & P.Diaz, Ontologies and the Semantic Web for E-learning, Journal of Educational Technology and Society, 2004,7(4), 26-142.
12. R.Mizoguchi & J.Bourdeau, Using ontological engineering to overcome common AI-ED problems. International Journal of Artificial Intelligence in Education, 2000, 11(2), 107-121.
13. Catherine Roussey, Francois Pinet, Myoung A.H. Kang, and Oscar Corcho,(2011), Ontologies in Urban Development Projects, *Advanced Information and Knowledge Processing*. 1, DOI 10.1007/978-0-85729-724-2_2, © Springer-Verlag London Limited.
14. M.Q. Zhou, G.H. Geng and S.G. Huang. Ontology Development for Insect Morphology and Taxonomy System, ‖ IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2006, 324–330.
15. I. Sommerville. Software Engineering, 8th edition: Pearson Education limited, 2004.

***Corresponding Author: Dr. J. Meenakumari;**
Department of Computer Science and Applications. The Oxford College of Science, Bangalore, India