

Ontologic Design of Software Engineering Knowledge Area Knowledge Components

Bulat Kubekov^{1,2}, Anar Utegenova^{*,1,3}, Leonid Bobrov², Vitaliy Naumenko^{1,2}, Aibek Ibraimkulov¹

¹Institute Information and Computational Technologies, CS MES RK, 050056, Kazakhstan

²Faculty of Information Technology Novosibirsk state university of Economics and Management, Novosibirsk, 630000, Russia

³Department of Computer and program engineering, Turan University, Turan, 050056, Kazakhstan

ARTICLE INFO

Article history:

Received: 25 December, 2019

Accepted: 23 May, 2020

Online: 06 July, 2020

Keywords:

Knowledge area engineering

Support concept

Support concept ontology

Knowledge trend

Knowledge content

Educational resources semantic context

Semantic interoperability

ABSTRACT

The article sets forth the solution of the educational resources semantic context knowledge components development task, based on the learning technology project-oriented concepts, graduate's competency model and ontological; engineering. The being considered ontology model and knowledge display formalisms allow, firstly, relevantly image the educational resources semantic context in the support concepts ontology format, and their specifications in the form of knowledge expressions and knowledge components, and secondly, secure the knowledge components semantic interoperability withing the frame of their usage in educational environment and systems. The design process of disciplines knowledge content degree programs and individual learning guidelines is connected with specifying the knowledge content frame, the configuration of which is defined with competencies signatures and smart-contract parameters.

1. Introduction

Knowledge area engineering is the technique, which allows fulfill the educational resources and degree programs working out, based on the knowledge components collection.

Knowledge area engineering gives a possibility to define the knowledge components family members general features and distinct characteristics of each of them [1], [2]. In our researches, applying the knowledge area engineering is connected with developing the knowledge components and their repeated usage upon generating the degree programs new disciplines, taking into account new technological and instrumental trends of software.

In our creations the knowledge area engineering is understood as the activity on the analysis and representation of the semantic context of educational resources with necessary and sufficient support concepts set, their formalization and specification in the form of ontology model and knowledge expressions.

Being considered methodology of the SWEBOK (Software Engineering Body of Knowledge) knowledge domain knowledge

*Corresponding Author: Utegenova Anar, Institute Information and Computational Technologies CS MES RK, utegenova77@mail.ru

components ontological design is based on the educational resources engineering and it is targeted at fulfilling the implementation of knowledge trend and knowledge content of disciplines working program syllabuses and software engineering degree programs, based on the knowledge components family. Requirements generality to separate components, in one knowledge area framework provides such components characteristics similarity. Therefore, the knowledge, having been worked out within SWEBOK knowledge domains, might be used in future upon preparing the new courses, including adjacent directions, as well, upon formulating individual degree programs and corresponding curricula [3].

Software engineering component is some working mechanism, the development of which is executed in reliance on the repeated usage. Such a working means in our researches is a knowledge component, representing a composition from the knowledge expressions and having the interface in the smart-contract form.

Models, having been introduced, knowledge representation arrangement formalisms, thus, allow maintain image pertinency of SWEBOK knowledge domains educational resources semantic context in the form of support concepts ontology, as well, their

semantic interoperability, upon designing the knowledge content of degree programs disciplines and individual learning guidelines.

Semantic interoperability is the possibility to interpret the educational resources semantic context by means of support concepts ontologies and repeatedly used knowledge components and to use them further in the new contexts, linked with design of software engineering basic and profile disciplines knowledge content.

Outlined knowledge components configuration conceptions are taken as a basis of being developed educational milieu, in the form of the Web-application with an adaptive cross-browser makeup, database and possibility of integrating into other informational-educational systems, in PDF, OWL and RDF formats.

2. Software engineering knowledge areas

The world leading universities develop for their students the courses on the software engineering, which supplement existing programs on informatics and computer engineering. In order to formulate an efficient curriculum compiling principles, IEEE Computer Society and ACM have elaborated, in the frame of a larger scale project Computing Curriculum, the recommendations set Software Engineering 2004 (SE2004) [4].

SE2004 describes in detail the knowledge (Software Engineering Education Knowledge, SEEK), which in the academic process shall be rendered by the universities and colleges within the course on software engineering.

In the version (2004) SWEBOK breaks down the knowledge on software engineering into fifteen knowledge areas.

It should be noted, that SWEBOK, as distinguished from SEEK, envisages deeper knowledge, as it assumes the software engineering knowledge of a specialist, having several years' experience, while SEEK supposes the knowledge, necessary to render for the trainees. On the other hand, SEEK stipulates wider range of topics, which SWEBOK deliberately has left out of view, in order to concentrate at software engineering basics.

Despite the noted difference in interpreting and applying the knowledge areas, the methodology, used in our researches, anticipates usage of such conceptions and mechanisms, which allow on a consistent basis display and formulate the knowledge models with subsequent design of the knowledge components.

The article herein shows forth the methodology of arranging and representing the SWEBOK knowledge domain semantic context knowledge with the support concepts system, support concepts representation with the ontologies from identifying and concrete concepts and support concepts ontology specification in the knowledge expressions format. The methodology's final aim consists in SWEBOK knowledge areas representation in the form of knowledge components repositories and with the aim of their further usage for designing the degree programs disciplines knowledge content and individual learning guidelines.

For configuring the knowledge trajectory there is used a smart-contract, preconditions and postconditions of which are competencies signatures.

3. Engineering education project-oriented technology

One of the important principles of programs upgrade for preparing the engineering staff to engineering activity is usage of CDIO – worldwide initiative approach, MIT [5,6]. CDIO approach gives possibility to form the degree programs disciplines optimal structures, and jointly with competency approach, to optimize the knowledge content and engineering education level programs quality[7]-[10].

CDIO declared aim is in the fact, that a higher school graduate shall be able to come up with a new product or new technical idea, fulfill all design works on their implementation, to give useful instructions to those, who will deal with them, introduce into production and apply the result. CDIO ideology and standards are applicable to training any engineering profile specialists as they are project-oriented learning technologies, focused at a student and integrated into the problems and experience of real production.

To reflect vocational specifics, at the level of graduates skills, abilities and capabilities in software engineering, the degree program shall be governed, firstly, with fifteen SWEBOK knowledge areas, defining basic engineering knowledge, secondly, with CDIO stages and standards, in which definite learning outcomes are represented as a detailed list of competences or qualification requirements.

Knowledge, skills and personal features are determined and codified in the list of planned learning outcomes in the so-called plan of CDIO Syllabus, which specifies, what the trainees shall know and be able upon completing their degree program.

As the educational needs and educational resources determining complex technology is the base for learning planning outcomes, in our researches the CDIO Syllabus is a plan, in which the main learning results are given as professional, basic and additional competencies signatures, used for defining the disciplines knowledge trend or learning trajectories.

As it has been noted, the CDIO Syllabus plan includes the list of employers requirements to engineering education, which, in respect to, learning project-oriented technology is interpreted in the form of competencies expressions of the CDIO project execution stages.

On the one hand, a higher school teacher's task consists in transferring the big subject knowledge volume to the students. On the other hand, engineers need necessary various personal and interpersonal competencies, as well, skills for objects, processes and systems creation, which will allow them work in a real team of engineers and benefit the society. CDIO approach has been developed with the aim to eliminate the conflict thereof and satisfy all needs of the trainees.

CDIO Syllabus is, as well, the main document for engineering educational programs reforming. From vocational engineers' point of view, it represents the employers list of requirements to engineering education level. With that, higher schools' teachers can consider the CDIO Syllabus plan to be a system of the main learning and competencies results, which shall own the graduates. List of students' knowledge, skills and competencies shall consist of sufficient general formulations, in order to apply it to many engineering areas. In the meantime, it shall be worked out thoroughly enough to be useful upon degree programs disciplines curricula planning and concrete specialty educational process assessing.

In our researches for the knowledge defining there is used the ontological engineering, the result of which is SWEBOK knowledge domain semantic imaging by means of support concepts ontologies, on the base of which there formed repeatedly used knowledge components, further applied for designing the degree program disciplines knowledge content and learning trajectory on the software engineering.

As the properties, endued to the CDIO Syllabus plan are quite critical for their certain use upon planning the learning outcomes, the properties thereof are considered to be an integral part of degree programs learning trajectories design process. In our researches, under the properties we understand the smart-contract prerequisites and postrequisites, where prerequisites are statements on the key competencies, indispensable for trainee's cognitive activity, and postrequisites are key competencies if the CDIO Syllabus plan has been acquired by the trainees upon successful mastering the knowledge components.

Smart-contract of the CDIO Syllabus plan initial level, has the following format:

$$\{P\} < \text{CDIO Syllabus} > \{Q\} \quad (1)$$

where $\{P\}$ – prerequisites, connected with the statements on competencies, necessary for successful learning, and $\{Q\}$ - postrequisites, which define the statements on competencies, which determined with success and quality of learning planned outcomes achievement.

Software engineering postrequisites are the statements, connected with trainee's fundamental skills and knowledge, having been formulated as following competencies:

- Possessing the software engineering knowledge and skills, necessary to start the practical work.
- Individual or team work on qualitative software creation.
- Designing on one or several subject areas, using software engineering approaches, uniting “ethic, social, legal and economic interests”.
- Demonstrating such skills, as interpersonal communication, work efficient methods, leadership and intercourse.
- Studying the new models, methods and technologies as far as they appear.

Further decomposition of the CDIO Syllabus plan and smart-contract properties is linked with the structure of the educational process within the learning courses framework or individual learning trajectories. For example, for individual learning trajectories case the smart-contract will include the set of knowledge components (KC, Knowledge Components) and learning scenario configuration parameters:

$$\{P\} < \{KC\}, Col, Lev > \{Q\} \quad (2)$$

where, a parameter Lev assigns required for a trainee level of professional, basic or additional knowledge components competencies, and a parameter Col – references to those knowledge components, which are indispensable for a trainee to master successfully the knowledge component from the set $\{KC\}$.

4. Formalisms and specifications of being used conceptions and mechanisms of their implementation

Ontology model is defined as multitudes trinary: $Om = \langle C, R, F \rangle$, where C – educational space concepts (terms) multitude; R –

multitude of ratios between the concepts; F - multitude of interpretation functions, the definitions of which are assigned at ratios between concepts in ontology.

Knowledge specifications language formalisms: dyadic relations: “composition - symbol ‘*’ ”- aggregation form with clearly-cut expressed relations of possession and coincidence of the life cycle between the concepts in ontology; “aggregation – symbol ‘+’ ” and “alternative selection– symbol ‘~’ ” - relations, by means of which there is imaged the semantic identity between concepts in the ontology [11-15].

Support concept is the basic abstraction of the educational milieu, reflecting the semantic generality and typical individual properties of the concepts of the given environment.

Identifying concepts are ontology concepts, by means of which the support concepts semantic and distinctive features, in reference to other educational space support concepts are defined.

Concrete concept is an ontology notion, which in the own identifying concept context configurate the support concept by means of typical, or different combination of child concepts, possessing clear and explicit description of semantic and distinctive properties of ontology support concept.

Knowledge expression is the specification of the support concept ontology by means of identifying and concrete concepts sequence with definite relations over the concepts: composition, aggregation and alternative selection. Knowledge expression implies the truth of conclusion on the support concept, that is, provided “parcel is the knowledge expression” is true, then the same is true for “conclusion is a support concept”.

For instance, specification of support concept ontology C_i is represented in the form of the following knowledge expression:

$$C_i \leq *C_{i,1} * C_{i,2} (*C_1 \sim C_2) * C_{i,3} + C_{i,4}; \quad (3)$$

From expression (3) it is seen, that the support concept ontology has a hierarchical structure, consisting of two levels, in which the first level concepts are identifying concepts - $C_{i,1}$, $C_{i,2}$, $C_{i,3}$ and $C_{i,4}$, each of which might have the level from identifying concepts. For example, an identifying concept $C_{i,2}$ has two concrete concepts from an obligatory concept C_1 and alternative non-obligatory concept C_2 .

Developed software editor allows forming the support concepts ontology in the form of knowledge expressions and visualizing them in the form of relational graph of support concept ontology C_i , as it is shown on the Figure 1.

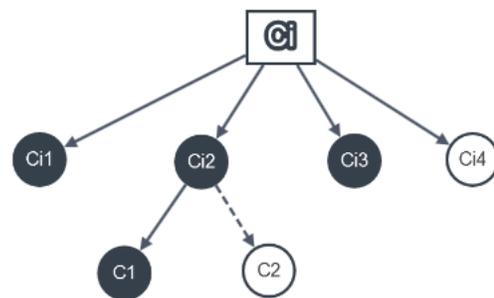


Figure 1: Relational graph of support concept ontology C_i

As it has been noted above, the CDIO Syllabus plan includes the employers' list of requirements to engineering education, which, in reference to the project-oriented teaching technology are interpreted in the form of competencies expression of the CDIO project execution stages.

Competencies expression (ec - expression of competences) is the specification of CDIO stage competency model, in the form of professional, basic and additional competencies signatures.

Let's give an example of the CDIO first stage competencies expressions – Conceive stage, connected with an analytical type of professional activity:

ec.PrC.Conceive \leq *PrC₁(*P₁*P₂)*PrC₂*PrC₃; - professional competencies

ec.BaC.Conceive \leq *BaC₁~+BaC₂*BaC₃(*B₁~+B₂)+BaC₄; - basic competencies

ec.AdC.Conceive \leq *AdC₁+AdC₂*AdC₃~+AdC₄; - additional competencies.

In the above examples under competencies we understand the dynamic knowledge aggregate, reflected in the support concepts, application of which is necessary to implement the project solutions in compliance with professional activity types of the CDIO stages.

Knowledge component (KC, Knowledge Component) is a composition of the knowledge expressions. Knowledge component possesses a clearly-cut definite smart-contract, by means of which the rules of its implementation and configuration management are represented upon the learning scenario designing. Absolutely all its dependencies on the environment have been described within the smart-contract framework.

Knowledge component is the one, from which there assembled the learning scenario family members. Constructively, the knowledge component is oriented to the knowledge duplexing minimization, raising the compatibility and, consequently, to the highest favor of repeated usage.

Knowledge trend is the knowledge model, being the composition of professional, basic and additional competencies of the CDIO stage competency model. Mastering each of the competency models assumes studying the support concepts set. Signatures of professional, basic and additional knowledge trend competencies represent learning planned outcomes - the CDIO Syllabus plan.

Knowledge component is the specification of the support concepts ontology of professional, basic and additional competencies in the knowledge expressions form. The knowledge content, therefore, defines the knowledge, the degree program discipline's working program or learning scenario shall possess [12-19].

5. Example of ontological engineering knowledge area “Software Configuration Management”

Proceeding from ontological engineering knowledge area “Software Configuration Management”, there have been selected six support concepts, representing the given knowledge area semantic context SWEBOK. As an example, we describe only one of the six support concepts of the field of knowledge «Software Configuration Management».

Support concept C₁ - SCM-process management.

SCM-activity, connected with product professional and integrity control via its elements identification, management and control over the changes, as well, checking and accountability according to configuration information.

Identifying concepts of the given support concept are:

C_{1.1} - Organizational context SCM.

C_{1.2} - SCM constraints and rules.

C_{1.3} – Planning in SCM. The given identifying concept is the second level ontology semantic context, including the following concrete concepts: C₁-Organization and duties; C₂-Resources and schedule; C₃ – Tools and implementation; C₄ - Suppliers/contractors control; C₅ – Interfaces control.

C_{1.4} - Configuration management plan.

C_{1.5} - SCM-process execution control. The given identifying concept is the second level ontology semantic context, including the following concrete concepts: C₁ - Metrix and quantitative assessment process in SCM; C₂- Audit in SCM framework.

Knowledge expression of the support concept “SCM-process management” is:

$$C_1 \leq *C_{1.1} * C_{1.2} * C_{1.3} (*C_1 * C_2 * C_3 + C_4 * C_5) * C_{1.4} * C_{1.5} (*C_1 * C_2); (4)$$

Thus, ontological engineering consists in displaying the SWEBOK knowledge areas semantic context, with necessary and enough support concepts set and with modeling the support concepts in ontology form-hierarchical structures from identifying and concrete concepts of the knowledge area herein.

To visualize the support concept ontology there has been used the software editor, allowing conduct the knowledge expression editing and fulfill the ontology assembly in relational graph view. For instance, a relational graph of the support concept ontology C₁ - SCM-process management, will have the view, as shown on the Figure 2.

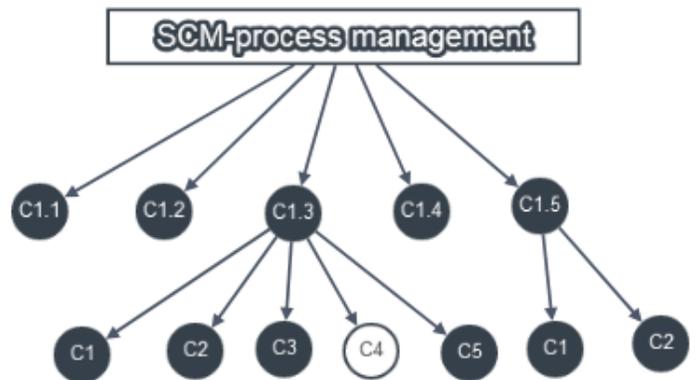


Figure 2: Relational graph of support concept ontology C₁ - SCM-process management

Let's consider the knowledge area – “Configuration management”, it is one of fifteen areas of SWEBOK, the semantic context of which is displayed with six support concepts, and

mastering of which secures vocational competencies, linked with configuration management processes.

6. Conclusion

The methodology, used concepts practical relevance is in: raising the training efficiency, based on schematic and symbolic teaching materials models; innovative solution on transfer to a new training information base, based on the project-competency model of knowledge organization and educational resources semantic context ontological engineering; development, based on the knowledge display ontological model, methodology of formulating and implementation of knowledge components, which might be recognized as an educational technology; consistency of representation and formalization of the knowledge components for designing the disciplines working programs knowledge content training guidelines, based on the smart-contract; providing the possibility for the knowledge components configuration in compliance with the demanded scenario and learning outcomes; habituation of students skills and rational techniques in structuring, compressing and visual representing the knowledge and, as a consequence, cognitive thinking development; possibility to form repeatedly used knowledge components, their design and adapting to a wide spectrum of degree programs and services.

Acknowledgment

The work has been executed with the support of the Ministry of education and science of the Republic of Kazakhstan, grant № AP05134973.

References

- [1] B. Kubekov, V. Naumenko, A. Utegenova, N. Zhaksybaeva, "Methodology of formation educational resources on basis of ontology", 12th IEEE International Conference "Application of Information and Communication Technologies – AICT2018", October 17-19, Almaty, Kazakhstan, 408-413, 2018.
- [2] K. Czarnecki, U. W. Eisenecker *Generative Programming. Methods, Tools, and Applications*/ Addison-Wesley, 731p, 2000.
- [3] E. Blomqvist, "Experiments on Pattern-Based Ontology Design/E. Blomqvist", In: Proc. of the 5th Int. Conf. on Knowledge Capture (K-CAP 2009) (September 1-4, Redondo Beach, California, USA). ACM, 41-48, 2009.
- [4] Software Engineering, Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. A Volume of the Computing Curricula Series. IEEE CS and ACM, 134 p, 2014.
- [5] F.Y. Crawlye, "CDIO program: Description of goals and tasks of bachelor degree engineering education", CDIO report. Ed. #1. MIT, 2001. <http://www.cdio.org>
- [6] E.F. Crawley, D.R. Brodeur, D.H. Soderholm, "The education of future aeronautical engineers: Conceiving, designing, implementing and operating", *Journal of Science Education and Technology*. 17(2). 138-151, 2008.
- [7] A.I. Chuchalin, M.G. Minin, S.Y. Kulyukina, "Experience of forming vocational and universal competencies of engineering programs graduates at foreign higher schools", *Higher education in Russia*. #10, p.p.105-116, 2010.
- [8] A.I. Chuchalin, "Bachelor program upgrade in methods and technology area, with account of engineering education international standards", *Higher education in Russia*. #10, 2011.
- [9] A.I. Chuchalin, A.V. Yepikhin, Y.A. Muratova, "Learning outcomes assessment planning upon designing degree programs", *Higher education in Russia*, #1, – 13-20, 2013.
- [10] A.I. Chuchalin, "Concerning application of CDIO approach for designing engineering education level programs", *Higher education in Russia*. #4(200), 2016.
- [11] U. Indira, K. Saule & Z. Gulnaz, "Developing a Model for Managing the Curriculum Model Competency Form", 46-51, 2019. DOI: 10.1145/3343485.3343503.
- [12] B. Kubekov, I. Utebergenov, "The use of multiparadigm approach in knowledge modeling", 7th International Conference on Education and New Learning Technologies". - Barcelona (Spain), 6th-8th of July. - 136-141, 2015.
- [13] B. Kubekov, J. Kuandykova, I. Utebergenov, A. Utegenova, "Application of the conceptual model of knowledge to formalization of concepts of educational content", 9th International Conference on Application of Information and Communication Technologies AICT2015.- Rostov-on-Don, 14-16 October. - 294-306, 2015.
- [14] B. Kubekov, "Educational components formation technology for the planned CDIO SYLLABUS education", 9th annual International Conference of Education, Research and Innovation - ICERI2016, Seville (Spain), 14-16 November, 6139-6145, 2016.
- [15] B. Kubekov, Beyer, Ditmar, A. Utegenova, N. Zhaksybaeva, "Innovative paradigm of education of knowledge - competency form based on ontology", *Journal of theoretical and applied information technology* 15th November 2017, 95 (21), -ongoing JATIT@LLS (E-ISSN 1817-3195 / ISSN 1992-8645). 5859-5868, 2005.
- [16] W. Wojcik, B. Kubekov, V. Naumenko, S. Narynov, S. Toibayeva, A. Utegenova, "Project-competency based approach and the ontological model of knowledge representation of the planned learning", - *INTL Journal of electronics and telecommunications*, 65(1), 75-80. 2018. doi:10.2478
- [17] B. Kubekov, L.K. Bobrov, E.A. Savelyeva, A. Utegenova, V. Naumenko, "Project-competent paradigm of Knowledge representation of the three-level engineering education System", 10th International scientific-practical conference on software engineering and computer systems, section "Ontology Models and Semantic Web", MICSECS 2018. <https://www.scopus.com/authid/detail.uri?authorId=56826094200>
- [18] B. Kubekov, L. Bobrov, U. Anar, N. Vitaly & A. Raigul, "Model of engineering education with the use of the competence-project approach ontological engineering and smart contracts of knowledge components", *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*. 3, pp 14-17, 2019. DOI: 10.35784/iapgos.243.
- [19] B.S. Kubekov, "Organization and presentation of the planned training knowledge, based on the ontology", *Monograph. – Almaty: "LP-Zhasulan"*– 336p, 2019.