Ontological Model of the Standardized Secondary School Curriculum in Informatics

Milinko Mandić*, Zora Konjović**, Mirjana Ivanović***
* Faculty of Education, University of Novi Sad, Sombor, Serbia
** Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Serbia
*** Faculty of Science, University of Novi Sad, Novi Sad, Serbia
milinmand@gmail.com, ftn_zora@uns.ac.rs, mira@dmi.uns.ac.rs

***Abstract*— The paper proposes ontological model of standardized secondary school curriculum in informatics. The model was created based on the ACM K12 CS curriculum proposal using competencies designed for the secondary level of education. The base class of ontological model is *Competence* with two direct subclasses (*Knowledge* and *Skills*). Skills are represented by classes corresponding to the categories of the cognitive process dimension of the revised Bloom's taxonomy. In addition to standardization of curriculum relying on ACM K12 CS curriculum model, a machine-readable representation that facilitates manipulation of the curriculum through applications intended for specific users (teachers, experts, administrative bodies, etc.) is proposed.**

I. Introduction

Ontologies can describe learning domains from different perspectives, allowing for a richer description and retrieval of learning contents [1]. Due to its ability to represent curriculum in a machine understandable manner, and the features of reuse and share, ontological approach has become widely used for representing some of the curriculum forms [1] [2] [3] [4] [5].

In [6] the authors state that the ontology offers an „objective base on which to build a curriculum recommendation". They use the ontology to represent computing curriculum and propose several potential uses of the ontology in curricula representation (for the purpose of distinguishing among computing programs and for highlighting the corresponding concepts in accordance with the selected outcome). In the paper [7] ontology has been created that allows the sharing of digital content for teaching mathematics and represents the mathematical domain of topics and skills. Reference [8] describes the development of a ontology-based curriculum knowledgebase which addresses the complexity of the interrelationships between the component parts of undergraduate enquiry based learning in medicine and other structured curricula and provides an approach to their collaborative maintenance. The same authors emphasize the importance of involving academics, students, teachers in the maintenance and improvement of a curriculum. Ref. [1] presents a system that has been developed with secondary school teachers that uses ontologies to support the development and management of the educative curriculum. In [3] the Bologna Ontology is created to model an academic environment as proposed by the Bologna reform. In [5], the model for representing ACM CS curricula based on the IEEE RCD standard is shown.

II. The Ontological Model of Curriculum

A. The class Competence

Informatics curricula have to be compliant at all the levels of education [5]. For example, higher education curricula for informatics teachers should be designed in the way that the graduated informatics teachers' *competencies* satisfy the needs of the current elementary and secondary school curricula [5]. After graduating from secondary school a student has to be educated enough in order to attend informatics courses of an adequate study program at higher education level. It is also important that secondary schools and faculties provide students with informatics *competencies* in order to satisfy companies' requirements for specific work posts.

Therefore, the ontological model of a secondary informatics curriculum is based on competencies, and the main class of the ontology is *Competence*.

There are numerous definitions of the term competence:

- A mixture of knowledge, skills, abilities, motivations, beliefs, values, and interests [9]
- A knowledge, skill, ability, or characteristic associated with high performance [10]
- It can be used to capture information about a skill, knowledge, ability, attitude, or learning outcome [11].

In [12], the authors list some examples of experts' efforts to define the term: "The knowledge, skills, and attributes that differentiate high performers from average performers", "It is a construct that helps define level of skill and knowledge", etc.

The same source concludes that "the term competence defines successful performance of a certain task or activity, or adequate knowledge of a certain domain of *knowledge* or *skill*".

Therefore, in this paper, the knowledge and skills mapped to specific classes of an ontological model of the curriculum (*Knowledge* and *Skills*), are represented as direct subclasses of *Competence*. (Figure 1). Classes *Skills* and *Knowledge* are related via object property *hasKnowledge*, that is its inverse property *hasSkill*.

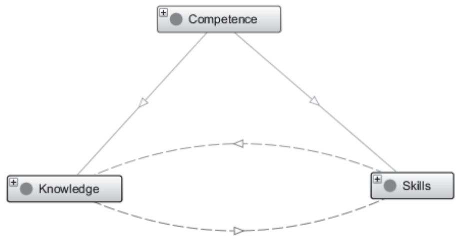

Figure 1. Structure of the class *Competence*

To ensure interoperability with learning management systems that provide information about competence, upper ontology classes are modelled in accordance with the IEEE RCD standard. The ontological representation of competencies based on the IEEE RCD standard (Figure 2) relies on the competences representation shown in [5] and [13].

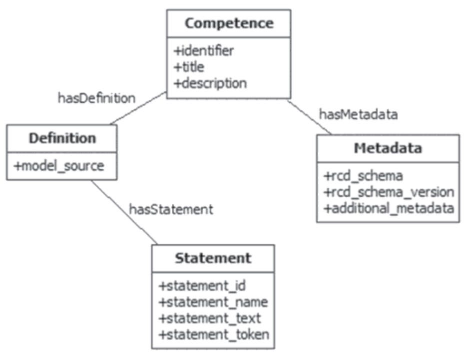

Figure 2. Ontological representation of the IEEE RCD standard [5]

The basic class is *Competence* with the properties *identifier, title and description*. Given that in the standard, these fields are defined as non-structured, they are represented as datatype properties in the ontological model. The field *definition*, although optional, is a structured part of the competence and can contain one or more *statements*, such as assessment criteria, outcomes, etc. It is modelled by the class *Definition* associated with the class *Statement* via the *hasStatement* object property. The *metadata* field is modelled by the *Metadata* class.

According to [11] the *Metadata* class enables the mapping of learning objects defined, for example, according to the IEEE LOM standard, while the classes *Definition* and *Statement* can be used for mapping data based on the principles for the assessment of student achievement, applied instructional methods, certification of competence, criteria, etc. Considering that the main goal of the ontological model curriculum presented in this paper is the defining of competencies from the perspective of acquired knowledge and skills, the classes *Metadata*, *Definition* and *Statement* are not structured in the ontology representing the secondary school curriculum.

Learning taxonomy is a way for describing the different behaviours in the learning process and the characteristics that students should develop throughout this process [14]. It provides a structure for the classification of educational goals and outcomes; in this paper it is used to define the class *Skills*. Skills are represented by classes corresponding to the categories of the cognitive process dimension of the revised Bloom's taxonomy [15], which is the dominant taxonomy in the area of Computer Science (CS) and in general [15] [16]. Exceptions are 'remember' and 'understand' categories, which are represented by a single class *Remember-understand,* due to the nature of having a CS domain in which the learning outcome involves only a recognition/memory without understanding is unlikely. Thus, the *Skills* subclasses are:

- *Remember-understand*,
- *Apply*,
- *Analyse*,
- *Evaluate* and
- *Create.*

B. *Creating the ontology of the secondary school informatics curriculum*

Secondary school informatics education clearly recognizes the need for and even provides the outcomes of curriculum standardization. The two most frequently cited models introducing curricula standardization in secondary school informatics education are the ACM K12 [17] and UNESCO/IFIP [18] models. ACM's K12 proposal is considered to be more modern and more comprehensive, because it proposes a curriculum based on computer science defined to represent a wider, more adequate and modern scientific discipline than ICT, as defined in [18].

For the above reasons, the ontological model of a secondary informatics curriculum in this paper was designed based on the ACM K12 CS curriculum proposal using only competencies designed for the secondary level of education (K8 or higher levels of standard).

Three general levels (L1, L2, L3) of the ACM K12 CS standard are separately described in detail in [19], [20] and [21] and consist of 12, 10 and 14 topics, respectively. Each topic contains a general description, a brief statement of support equipment for teaching, an assessment recommendations, detailed learning objectives, the focuses of each area and the proposal for the implementation of each of the focuses. Figure 3 shows part of the topic "Programming Languages" of the L2 level of ACM's proposal.

The ontological model of the secondary school informatics curriculum was created in two phases. Protégé tool [22] was used for the creation of the ontology. HermiT reasoner [23] was used for semantic verification of the model.

In the first phase, each of 36 topics of all three levels was modelled as a subclass of the *Knowledge* class. Focuses of the topics were modelled as the topic's subclasses. Learning objectives defined in [19], [20], [21] were mapped to the corresponding subclasses of the *Skills* class in accordance with the cognitive processes dimension of the revised Bloom's taxonomy.

For determining which general subclass of the *Skills* class certain objective belonged to, the synonyms of Bloom's taxonomy categories, as well as the detailed descriptions of the revised Bloom's taxonomy [24] [25] and digital taxonomy [26] were used. Learning objectives were related by the object property *hasKnowledge* with the appropriate topic that according to ACM references/documents they belonged to.

Other fields appearing in the curricula (teaching methods, knowledge assessment) are not currently covered by the ontological model. However, the model nonetheless enables mapping of these fields using the *Definition* and *Statement* subclasses. Additionally, learning objects can be easily incorporated into the model through the *Metadata* subclasses and connected via the object property with the appropriate thematic area.

Topic 11: Programming Languages

Topic Description:
Programming Languages will introduce the student to some basic issues associated with program design and development. The focus of this unit is to establish an appreciation of the work being done by software.

Textbooks and Supplies:
A programming language; interactive development environment recommended.

Time to Complete: 2-4 weeks

Student Learning Objectives	Assessment Measures
The student will be able to:	
1. Code, test, and execute a program that corresponds to a set of specifications.	Lab activity
2. Convert a word problem into code using top-down design.	Written activity Lab activity
3. Select appropriate data types.	Written activity Lab activity
4. Write structured program code.	Lab activity
5. Draw a series of diagrams showing the scope and values of variables during execution of a simple program.	Written activity

Assessment Recommendations: An average of 60% from combined assessment measures is required to demonstrate proficiency in course material.	
Lab activities	50%
Written activities, including tests, quizzes, and written assignments	50%

Detailed Outline	
Focus	**Sample Lab / Hands-on Activity**
1. Terminology	Identify and define key terms associated with programming.
2. Representation of text inside the computer	Each student writes a sentence in binary and exchanges it with a neighbor. The neighbor translates the sentence into text. Students stand or sit to mime a secret word in binary. Flashlights can also be used to represent binary code.
3. Representation of numbers inside the computer, including the largest and smallest values which can be represented in each of several types	Numbers are placed into imaginary bytes in a grid, each imaginary byte having a unique address. (A spreadsheet can be used for this purpose.) Instructions are provided to add and subtract values by address. Some of the resulting numbers should be too large to store in the imaginary byte and will overflow.

Figure 3. Topic 11: 'Programming Languages'

Thus, for example, the topic 'Programming Languages' (Figure 3) was mapped in the subclass of *Knowledge* class, while the topic's focuses were mapped in the subclasses of *Programming_Languages* class. The objective 'Write structured program code', in accordance with the revised Bloom's taxonomy, was modelled as a subclass of *Create* class and was associated with the class *Programming_languages*. The objective 'Select appropriate data types' was modelled by the subclass of *Analyze* class and was associated with the class *Programming_Languages*.

In the second phase, 36 topics of all three levels were placed into 13 areas that were determined based on overlapping of the focuses and goals in different topics. Topics and focuses with overlapping were labelled as 'related'. Then, related topics were mapped to one class and/or represented as subclasses of a common superclass. Characteristic examples are three topics: 'Problem solving', 'Algorithms', 'Problem solving and algorithms'. The parent classes *Problem_solving* and *Algorithms* became subclasses of the class *Problem_solving_and_algorithms* and the focuses belonging to the topics 'Problem solving' and 'Algorithms' were mapped to the corresponding subclasses of the class *Problem_solving_and_Algorithms*.

Related focuses of different topics were mapped to subclasses of a single superclass or to the superclass to which the related topic was mapped in the case where focus is represented by some special related topics (it appeared that some focuses of the lower-level ACM K12 curriculum (L1 or L2) were represented by special related topics in L2 or L3). An example is the focus 'Computing careers' of L1 level, which has been mapped to the *Careers_in_computing* superclass, as in L2 and L3 there is a special topic with the same name (Figure 4).

- Computers_in_Society
  - Careers_in_Computing
    - Careers_related_to_computers
    - Job-Career_Titles_and_Descriptions
    - Skills_and_academic_background
    - Sources_providing_information_about_computer_careers
    - Use_of_computers_in_other_career_choices
  - Future_changes
  - History_of_computer_science
  - Interdisciplinary_Utility_of_Computers_and_Problem_Solving_in_the_Modern_World
  - Social_impacts_of_computerization

Figure 4. Structure of the class *Careers_in_computing*

Additionally, if some focuses have been repeated in several topics (as has, for example, arrays), they are mapped to a single class. Thus, the resulting final list of subclasses of the *Knowledge* class is shown in Figure 5.

- Knowledge
  - Algorithms_and_data_structures
  - Applications
  - Computer_networks
  - Computers_in_Society
  - Connections_Between_Mathematics_and_Computer_science
  - Design_for_usability
  - Ethical_and_security_Issues
  - Internet_concepts
  - Models_of_Intelligent_Behavior
  - Multimedia
  - Operating_Systems
  - Principles_of_computer_organization
  - Programming_and_software_engineering

Figure 5. Subclasses of the class *Knowledge*

In order to include the latest topics in the curriculum, all the objectives defined in the latest current integrated version of the standard [17] were analysed and if some of them were not included in [19], [20] or [21], then the objective was added to the model and associated with the appropriate thematic area (subclass of *Knowledge* class). The reason is that in [17] learning objectives of students upon completion of specific levels of K12 curriculum are primarily defined, without sufficient explicit information about the required topics and knowledge that would enable consistent mapping into *Knowledg*e subclasses of ontological model.

Examples of the learning objectives from [17], added to the ontological model, are:

- Evaluate what kinds of problems can be solved using modeling and simulation,
- Evaluate algorithms by their efficiency, correctness, and clarity,
- Use mobile devices/emulators to design, develop, and implement mobile computing applications,
- Use Application Program Interfaces (APIs) and libraries to facilitate programming solutions,

- Demonstrate concurrency by separating processes into threads and dividing data into parallel streams.

Listing 1 presents a part of the owl code of the proposed ontological model.

```
<owl:Class
rdf:about="#Code_a_program_that_corresponds_to_a
_set_of_specifications">
      <rdfs:subClassOf>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty
rdf:about="#hasKnowledge"/>
          </owl:onProperty>
          <owl:someValuesFrom>
            <owl:Class
rdf:about="#Programming_Languages"/>
          </owl:someValuesFrom>
        </owl:Restriction>
      </rdfs:subClassOf>
      <rdfs:subClassOf rdf:resource="#Create"/>
    </owl:Class>
  <owl:Class
rdf:ID="Convert_between_decimal_binary_and_hexad
ecimal_numbers">
      <rdfs:subClassOf>
        <owl:Restriction>
          <owl:someValuesFrom>
            <owl:Class
rdf:about="#Connections_Between_Mathematics_and_
Computer_science"/>
          </owl:someValuesFrom>
          <owl:onProperty>
            <owl:ObjectProperty
rdf:about="#hasKnowledge"/>
          </owl:onProperty>
        </owl:Restriction>
      </rdfs:subClassOf>
      <rdfs:subClassOf rdf:resource="#Apply"/>
    </owl:Class>
```

Listing 1. A part of curriculum ontology

The ontological model is at the following address: www.pef.uns.ac.rs/SecondaryInformaticsCurriculum/index.html.

III. Conclusion

The main contribution of this paper is the ontological representation of a standardized secondary school informatics curriculum.

The advantages herein are the standardization of curriculum relying on ACM K12 CS, and machine-readable representation of the curriculum that facilitates curriculum manipulation through applications intended for specific user groups (teachers, experts in the domain field, the administrative bodies responsible for education management, etc.).

The simplicity of the model, which limits representation of the outcomes, representation of courses/topics prerequisites and aspects of the curriculum that are not closely related to the competencies, such as instructional method, methods of assessment, learning objects and the like, could be considered a model deficiency.

However, this deficiency may be mitigated by enriching the ontological model, which is one of the suggested future research directions.

For example, the model can be extended by alternatives for the outcomes' representations, which means adding new taxonomies in addition to the revised Bloom's taxonomy.

Introduction of the transitive object relation "prerequisite" would allow mapping information on the prerequisites for the study of a specific topic or course. In the existing curricula, this relationship is usually implicitly defined through the year/level of study, which makes it possible to establish preliminary links that can later be refined by manual intervention and/or machine learning methods applied to the courses/topics content.

References

[1] .J.T. Fernández-Breis, D. Castellanos-Nieves, J. Hernández-Franco, C. Soler-Segovia, M. C. Robles-Redondo, R. González-Martínez, and M. P. Prendes-Espinosa, „A semantic platform for the management of the educative curriculum," *Expert Systems with Applications*, vol. 39, no. 5, 2012, pp. 6011-6019.

[2] Y.L. Chi, "Ontology-based curriculum content sequencing system with semantic rules," *Expert Systems with Applications*, vol. 36, no. 4, 2009, pp. 7838–7847.

[3] G. Demartini, I. Enchev, J. Gapany, and P. Cudré-Mauroux,"The Bowlogna Ontology: Fostering Open Curricula and Agile Knowledge Bases for Europe's Higher Education Landscape," *Semantic Web - Interoperability, Usability, Applicability*, vol. 4, no. 1, 2013, pp. 53-63.

[4] A. Elsayed, "Interaction with Content through the Curriculum Lifecycle," *Advanced Learning Technologies, ICALT 2009*, 2009, pp. 730 – 731.

[5] M. Mandić, M. Segedinac, G. Savić, and Z. Konjović,"IEEE RCD standard based ontological modeling of Computer Science curriculum," *Proceedings of the 3th International Conference on Information Society and Techology*, 2013, pp. 189-285.

[6] L. Cassel, G. Davies, R. LeBlank, L. Snyder, and H. Topi, "Using Computing Ontology as a Foundation for Curriculum Development," *Proc. SWEL@ITS '08: The Sixth International Workshop on Ontologies and Semantic Web for E-Learning*, 2008, pp. 21-30.

[7] P. Libbrecht, C. Desmoulins, C. Mercat, C. Laborde, M. Dietrich, and M. Hendriks, "Cross-curriculum search for intergeo.," In: S. Autexier, J. Campbell, J. Rubio, V. Sorge, M. Suzuki, & F. Wiedijk (Eds.), Intelligent computer mathematics, *Lecture Notes in Computer Science*, Vol. 5144/2008, 2008, pp. 520–535.

[8] H. Dexter and I. Davies, "An ontology-based curriculum knowledgebase for managing complexity and change," *Ninth IEEE International Conference on Advanced Learning Technologies*, *ICALT*, 2009, pp.136-140.

[9] E. A. Fleishman, L. I. Wetrogan, C. E. Uhlman, and J. C. Marshall-Mies, „Abilities," In N. G. Peterson, M. D.,Mumford, W. C. Borman, P. R. Jeanneret, and E. A. Fleishman (Eds.), *Development of prototype occupational information network content model* , Salt Lake City, UT: Utah Department of Employment Security, vol. 1, 1995, p. 1086.

[10] R. Mirabile, "Everything you wanted to know about competency modeling," *Training and Development*, vol. 51, no. 8, 1997, pp. 73–77.

[11] *IEEE Standard for Learning Technology—Data Model for Reusable Competency Definitions*, Learning Technology Standards Committee of the IEEE Computer Society, 2008. http://www.doleta.gov/usworkforce/pdf/2007-ieeecomp.pdf.

[12] J. Shippmann, R. Ash, M. Battista, L. Carr, L. Eyde, B. Hesketh, J. Kehoe, K. Pearlman, E. Prien, and J. Sanchez, "The practice of competency modeling," *Personnel Psychology*, vol. 53, no. 3, 2000, pp. 703-740.

[13] J. De Coi, E. Herder, A. Koesling, C. Lofi, D. Olmedilla, O. Papatreou, and W. Siberski, "A Model for Competence Gap Analysis," *Proceedings of 3rd International Conference on Web Information Systems and Technologies (WEBIST)*, 2007.

[14] G. O'Neill and F. Murphy, "Guide to Taxonomies of Learning," *UCD Teaching and Learning/ Resources*, 2010. http://www.ucd.ie/t4cms/ucdtla0034.pdf

[15] U. Fuller, et al., "Developing a Computer Science-Specific Learning Taxonomy," *ACM SIGCSE Bulletin*, vol. 39, no. 4, pp. 152-170, 2007.

[16] L.W. Anderson, D.R. Krathwohl, P.W. Airasian, K.A. Cruikshank, R.E. Mayer, P.R. Pintrich, J. Raths, and M.C. Wittrock, *A taxonomy for Learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives,* New York: Addison Wesley Longman, 2001.

[17] CSTA K–12 Computer Science Standards, The CSTA Standards Task Force, 2011. http://csta.acm.org/Curriculum/sub/CurrFiles/CSTA_K-12_CSS.pdf

[18] *INFORMATION AND COMMUNICATION TECHNOLOGY IN SECONDARY EDUCATION, A Curriculum for Schools,* UNESCO, 2000. http://wwwedu.ge.ch/cptic/prospective/projets/unesco/en/curriculum2000.pdf

[19] A. Verno, D. Carter, R. Cutler, M. Hutton, and L. Pitt, "A Model Curriculum for K-12 Computer Science: Level II Objectives and Outlines," 2004. http://csta.acm.org/Curriculum/sub/CurrFiles/L2-Objectives-and-Outlines.pdf

[20] B. Madden, A. Verno, D. Carter, S. Cooper, T. Cortina, R. Cudworth , B. Ericson, and E. Parys, "A Model Curriculum for K-12 Computer Science: Level III Objectives and Outlines," 2007. http://csta.acm.org/Curriculum/sub/CurrFiles/L3-Objectives-and-Outlines.pdf

[21] D. Frost, A. Verno, D. Buckhart, M. Hutton, and K. North, "A model curriculum for K-12 Computer Science: Level I Objectives and Outlines," 2009. http://csta.acm.org/Curriculum/sub/CurrFiles/L1-Objectives-and-Outlines.pdf.

[22] N. F. Noy, R. W. Fergerson, and M.A. Musen, "The knowledge model of Protege-2000: Combining interoperability and flexibility," *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management (EKAW'00)*, 2000, pp. 17-32.

[23] R. Shearer, B. Motik, and I. Horrocks, "HermiT: A Highly -Efficient OWL Reasoner," *Proceedings of the OWL Experiences and Directions Workshop*, 2008.

[24] D. R. Krathwohl, „A revision of bloom's taxonomy: An overview," *Theory into Practice*, vol. 41, no. 4, 2002, pp. 212-218.

[25] R.Heer, "A Model of Learning Objectives–based on A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives," Center for Excellence in Learning and Teaching, Iowa State University, 2012. http://www.celt.iastate.edu/teaching-resources/effective-practice/revised-blooms-taxonomy/

[26] A. Churches, "Bloom's Digital Taxonomy," 2007. http://www.techlearning.com/techlearning/archives/2008/04/andrewchurches.pdf