2014 International Symposium on Technology Management and Emerging Technologies (ISTMET 2014), May 27 - 29, 2014, Bandung, Indonesia

Ontology Development of Metacognitive Support System for Novice Programmers (MSSNP)

Siti Nurulain Mohd Rum
Department of Information System,
Faculty of Computer Science and Information Technology,
University of Malaya
Kuala Lumpur, Malaysia
snurulain@siswa.um.edu.my

Dr. Maizatul Akmar Ismail
Department of Information System,
Faculty of Computer Science and Information Technology,
University of Malaya
Kuala Lumpur, Malaysia
maizatul@um.edu.my

Abstract— **Learning and teaching computer programming have been acknowledged as being difficult and challenging. The metacognitive learning environment is needed for learning success in Computer Programming problem solving. In designing a support learning tool, the main principal of the instructional approach is to support metacognitive ability. Ontology has gained popularity in building E-learning, in which the formalism of ontology-based description can be achieved, thus giving a specification domain scope and offering an effective reuse of software patterns and systems of patterns. As there is no exact methodology for developing ontologies, here, in this paper, we present an approach for the development of MSSNP Ontology based on the general criteria and issues to be considered.**

Keywords- Ontology, E-learning, Metacognitive, Computer Programming, Ontology Development

I. INTRODUCTION

E-learning is a flexible learning that is not just about providing convenient learning resources through a learning resources repository at anytime and anywhere, but is also associated with supporting personal learning and the instructor-led and self-paced learning interchange of information, as well as the cooperation between learners and instructors [1, 2]. To achieve the context of e-learning, control should be given to the learner in planning their learning pace, assessing their learning, knowledge, and making the cognitive process of reaching a decision on various aspects in their learning process. These skills are also known as metacognitive skills that are required throughout the learning process. As proposed by [6], supporting metacognitive ability is one of the key E-learning instructional design principles. Having better support in regulating metacognitive skills in the learning environment can help students learn better, and, furthermore, can help them to be self-regulated learners across contexts and domains. In fact, research has revealed the positive impact on metacognitive behavior of learner through a well-designed instruction in support system. The expansion of educational technologies in the last decade has compelled interest in new methods for bringing the learning content to learners. In recent years, the semantic web is one of the 'hottest' among the E-learning technologies in which the major component is called an ontology. It is about making the content of E-learning more understandable by machines [7], and, furthermore, adaptively in the context of user input and experience. It is also about developing intelligent agents to perform complex actions on the web [5]. Moreover, the semantic web is about to bring together information in an intelligent way by fully and clearly declaring the knowledge encapsulated in many web-based applications, equipped with Internet access semantically [3]. In this paper, we have developed the ontology as the backbone of the metacognitive support system for novice programmers (MSSNP). Ontologies are used as the knowledge representation mechanism of learning content in order to support metacognitive skills. In addition, ontologies are used as a mechanism for doing inference in order to provide learners with a suitable educational resource for their intelligent query.

II. CONCEPTUAL FRAMEWORK OF MSSNP AND ONTOLOGY DEVELOPMENT

The simple rule in designing ontology is that there is no exact or specific way of modeling a domain as there are always practicable alternatives [8]. The ontology designs almost always rely on the application. The general rule of thumb in ontology development is that it requires an iterative process and that the relationships in the domain of interest, as well as the objects that describe the logical or physical, are something that ontology concepts should take into consideration. These are the most probable objects, also called nouns, or relationships, also called verbs, in the sentences expressing the domain. In the MSSNP ontology development process, there are seven steps and criteria, as listed below:

1. Scope and boundary definition
2. Reuse consideration
3. Class design/enumerate terms
4. Taxonomic identification
5. Property identification
6. Data property identification
7. Anomaly validity check

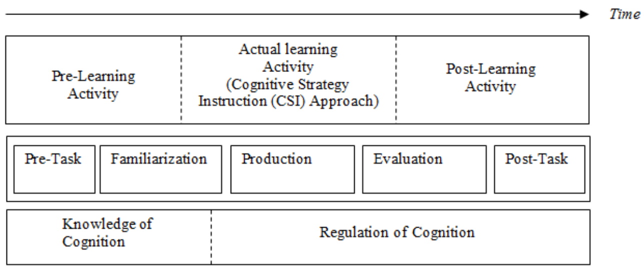

Figure 1. MSSNP Conceptual Framework Design

The development of MSSNP ontology is based on the framework design of the MSSNP, as shown in Figure 1. The MSSNP framework consists of timing and instructional approach elements, named as 'pre-learning', the tasks that take place before the learning session; the 'cognitive strategy instruction', the activity that takes place during the learning session; and, finally, the 'post-learning' activity, which takes place after the learning session. The conceptual stages for the MSSNP are divided into five; namely, 'pre-task', 'familiarization', 'production', 'evaluation' and 'post-task'. There are two main MSSNP activities involved at this stage, the 'post-task' and part of the 'familiarization' activities. The 'during' learning stage is where the cognitive strategy instruction (CSI) process takes place. Part of the 'familiarization', 'production' and partial 'evaluation' activities are also involved in this stage. The 'post-task takes place after the completion of the learning task with the objective of reflecting the student's learning process and performance. The activities involved in this stage are 'post-task' and part of the 'evaluation' activities. As illustrated in Figure 1, there are five main stages in the MSSNP framework, the post-task, familiarization, production, evaluation and post-task. Each of the stages has its own sub-activities and objectives as described in the next section.

A. Scope definition

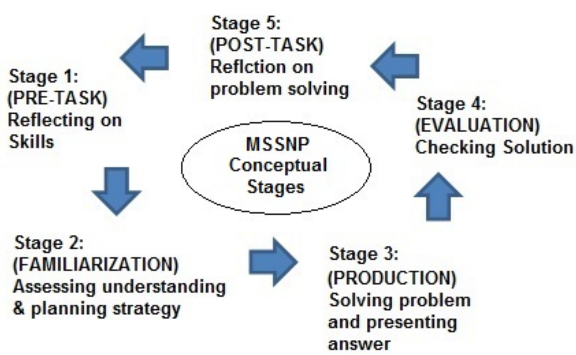

Figure 2. The MSSNP Conceptual Stages

One of the ways of determining the scope of the ontology is by sketching a list of questions, competency questions [4] concerning how to achieve its objectives. Later, the litmus test of using these questions will be used to determine whether the information depicted in the ontology is sufficient to correct the response to these questions. Is a very specific representation of an answer needed? These competency questions do not need to be comprehensive as they are just a sketch. Following the MSSNP framework, as shown in Figure 2, the following is the definition of each state and its objective and the possible competency questions at each stage

Stage 1: Reflecting on skills (PRE-TASK) – At this stage, the necessary aspect of metacognitive ability is covered to start the new problem. This stage is always considered to be self-reflection, which happens before the learning process. Students will be provided with suitable conditions for making them realize the benefits of general strategies, available resources as well as the degree of attention that is necessary to succeed in the problem-solving process and the activity. The objective of this stage is to trigger reflection on the monitoring progress of the student's knowledge. It focuses on the past problems and performance (low, average or high) of the student and comparing their estimation and judgment to their actual knowledge and understanding. At this stage, the student will be exposed to the following activities:

i. Monitoring knowledge and performance comparison
ii. Analysis of knowledge monitoring state, this is where the KMM is applied

Possible competency questions at this stage:

- What is the student's performance in learning?
- What strategies should be used?

Stage 2: Assessing understanding and planning strategies (FAMILIARIZATION) – At this stage, students are required to select strategies that are presented to them, or, alternatively, they can select a strategy by composing a new one. The objective of this phase is to make them reflect on the strategies as this activity helps them think of the strategies that are relevant and apply them appropriately. This phase focuses on metacognitive strategies related to the process during problem-solving. The student will be exposed to the following activities:

i. Understanding the components of problems and self-assessment
ii. Metacognitive selection strategies

Possible competency at this stage

- What is the appropriate strategy to implement in solving one particular problem?
- What is the resource that I can refer to?

Stage 3: Problem comprehension and self-assessment (PRODUCTION) – At this stage the student is required to solve a given task and problem and present an answer. The objective of this stage is to reflect the student's understanding concerning the concept as well as their confidence to solve the problem correctly. This activity is related to the actual assessment of student performance. The activities concentrate on translating the problem into the algorithm and monitoring the application of their planned strategies. The activities involved in this stage comprise:

i. Problem solving
ii. Quiz
iii. Checking answer

Possible Competency questions at this stage

- What is the correct answer for a particular problem?
- How do you solve a problem?

Stage 4: Problem solving, evaluation and experience (EVALUATION) – This stage only involves the activity of checking the solution provided by the lecturer, which is used as a comparison in studying the student's solution.

Stage 5: Reflection on problem solving (POST-TASK) – The activities designed at this stage provide the opportunity to the student to review their most recent experience, and explore what happened during the problem-solving activity. The aim is to assist the student in identifying the 'cause of the mistake' that relates to the problem, the recourses used and the issues relating to time management. The goal is to trigger self-questioning about the learning experience, such as:

- Did I use the available resources in solving the problem?
- How did I spend the time in solving the problem?
- Did I use the appropriate plan of action?

By doing this, students can build a better insight of problem-solving practice and experience.

B. Reuse Consideration

In this paper, the ontology is created from scratch since there is a lack of ontology in e-learning supporting the metacognitive learning environment. Considering what someone else has contributed can be worthwhile, in that we can extend the existing sources of particular tasks and domains by refining them to suit a particular scenario. Reusing existing sources may be a requirement for a system that needs to communicate with other applications that have already committed to a specific vocabulary that is controlled by the organization, field or domain, etc.

C. Enumerate Terms

It is beneficial to list down all the terms that we intend to use to explain things to a user. Taking into consideration the properties of the terms; for example, in this study, important metacognitive learning related terms will include student, syllabus, performance, knowledge monitoring, strategy and so on. First and foremost, it is important to comprehensively list down all the related terms, concepts or classes. The next step is to develop the association, class hierarchy between classes and defined properties. Based on the conceptual stages of MSSNP, the identification of possible terms for each stage is listed as follows:

- PRE-TASK stage – Metacognitive Strategies, Student
- FAMILIARIZATION stage – Problem, Programming Strategy
- PRODUCTION stage – Solution, Problem Attempt, Problem relation, Problem Solution
- EVALUATION stage – Lecturer, Solution
- POST-TASK stage – Performance

Metacognitive strategies relate to the plan or strategies used by the student in solving a problem; for example, 'read the problem question more than once', 'apply a similar solution as used in the past for a similar problem'. There are a number of programming strategies in solving problems (i.e. Figure 3, Figure 4) that can be applied, such as array, object oriented and control structure (i.e. loop, if condition). The student will attempt to solve a given problem. Each problem has a solution in which the student can compare their solution with the solution provided by the lecturer and evaluate their performance as well.

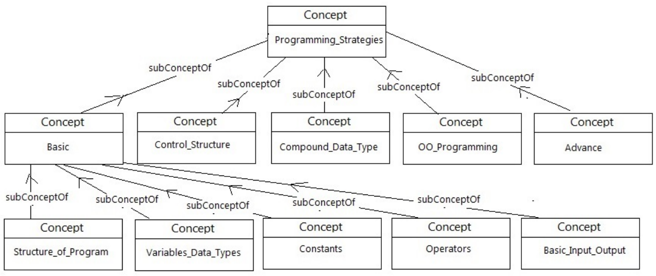

Figure 3. Basic programming strategy

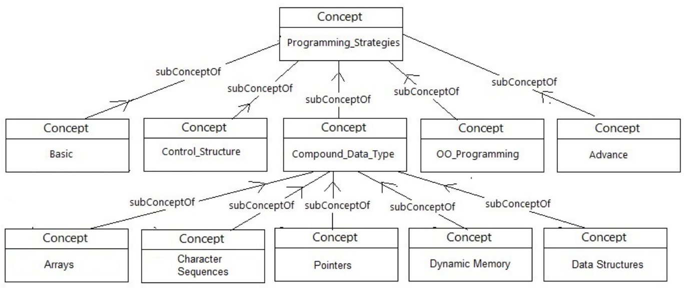

Figure 4. Compound Data Type Programming strategy

D. Taxonomic Identification

Figure 3 and Figure 4 show the programming strategy syllabus broken down into possible tutorials or chapters. Figure 5 presents the three levels of generalization, the top level, middle level and bottom level. In this study, we have employed the combination approach of top-down and bottom-up for the development of the MSSNP ontology. We generalized and specialized them appropriately after specifying the key concepts. We started with the top-level approach, such as programming strategies, before connecting them to a middle-level concept, such as object oriented programming and drill down to polymorphism as the specialization chapters. According to [8], none of these approaches (A top-down, bottom-up, combination) is superior to any of the others. The technique to choose strongly depends on the personal view of a particular domain [8]. The easiest technique for many ontology developers is using the combination approach top-down and bottom-up, since the "in the middle" concept tends to be the more descriptive concept in the domain [9].

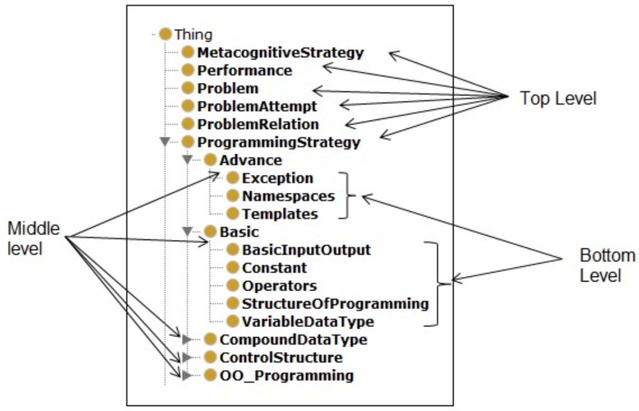

Figure 5. The Different level of MSSNP ontology

E. Property Identification

In order to answer the competency questions, classes alone will not be sufficient to provide the information. Once the classes have been identified, the concept's internal structure must be described. In step 3, we have created classes selected from the list of terms. Most of the training terms are likely to be properties of these classes. These terms include, for example, a property in OWL describes the relationship among the classes. There are two main types of property: object and data type property. Datatype properties link individuals to data value, while object properties are associated with the relationship between individuals. The third property is the annotation property, which can be utilized to add information.

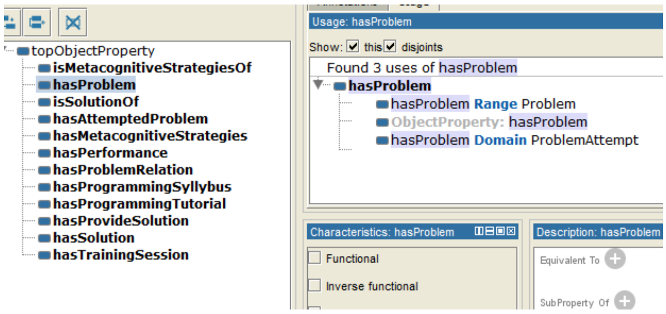

Figure 6. Object Property in MSSNP Ontology

There are two types of object properties involved in MSSNP ontology – the functional properties and inverse properties,

Functional Properties – As illustrated in Figure 7, this property is limited to one unique relationship to another individual for a specified individual, i.e. there cannot be two distinct values y1 and y2 such that the pairs (x,y1) and (x,y2) are both instances of this property. In this study, the functional properties are applied in which a student and a lecturer can have one account to access the MSSNP application

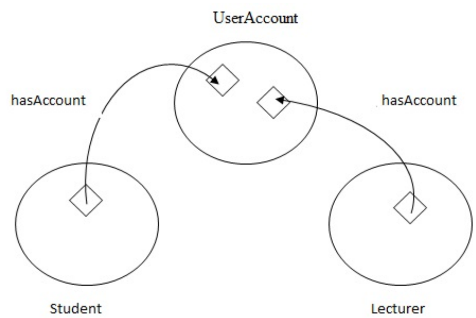

Figure 7. hasAccount functional property

```
<owl:ObjectProperty
rdf:about="&Ontology1373265476454;hasAccount">
        <rdf:type
rdf:resource="&owl;FunctionalProperty"/>
        <rdfs:domain
rdf:resource="&Ontology1373265476454;Lecture"/>
        <rdfs:domain
rdf:resource="&Ontology1373265476454;Student"/>
        <rdfs:range
rdf:resource="&Ontology1373265476454;UserAccount"/>
    </owl:ObjectProperty>
```

Inverse Functional Properties – Properties that describe the individual/domain are the inverse of another individual/ domain. In MSSNP ontology, the domain and the range for the hasSolution property and its inverse property isSolution. The domain of hasSolution is the problem and the range of hasSolution is ProblemSolution, the domain and range for isSolutionOf are the domain and range for hasSolution swapped over.

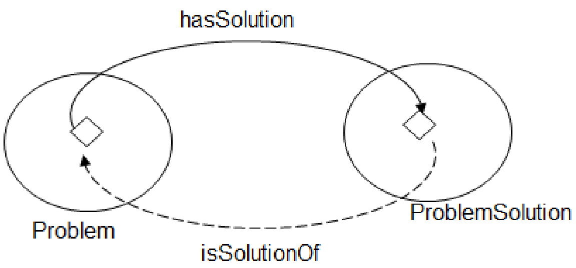

Figure 8. The domain and range for the hasSolution property and its inverse property isSolutionOf

```
<owl:ObjectProperty
rdf:about="&Ontology1373265476454;hasSolution">
        <rdf:type
rdf:resource="&owl;InverseFunctionalProperty"/>
        <rdfs:domain
rdf:resource="&Ontology1373265476454;Problem"/>
        <rdfs:range
rdf:resource="&Ontology1373265476454;ProblemSolution"/>
        <owl:inverseOf
rdf:resource="&Ontology1373265476454;isSolutionOf"/>
 </owl:ObjectProperty
```

F. Data Property Identification

Data properties connecting the individual to rdf or XML schema datatype. They describe relationships between an individual and data value. The ontology structure of MSSNP in this study has several data properties, as shown in Figure 9.

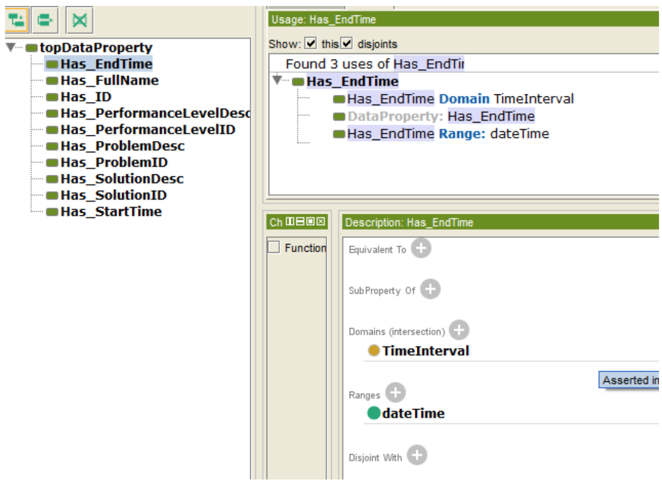

Figure 9. Data property in MSSNP

Has_Endtime – Properties of domain TimeInterval with range dateTime

```
<owl:DatatypeProperty
rdf:about="&Ontology1373265476454;Has_EndTime">
        <rdfs:domain
rdf:resource="&Ontology1373265476454;TimeInterval"/>
        <rdfs:range rdf:resource="&xsd;dateTime"/>
    </owl:DatatypeProperty>
```

Has_Fullname – Properties of domain Student with range String

```
<owl:DatatypeProperty
rdf:about="&Ontology1373265476454;Has_FullName">
        <rdfs:domain
rdf:resource="&Ontology1373265476454;Student"/>
        <rdfs:range rdf:resource="&xsd;int"/>
        <rdfs:subPropertyOf
rdf:resource="&owl;topDataProperty"/>
    </owl:DatatypeProperty>
```

Has_ID – Properties of domain Student with range Int

```
<owl:DatatypeProperty
rdf:about="&Ontology1373265476454;Has_ID">
        <rdfs:domain
rdf:resource="&Ontology1373265476454;ProblemAttempt"/>
        <rdfs:domain
rdf:resource="&Ontology1373265476454;Student"/>
        <rdfs:range rdf:resource="&xsd;string"/>
    </owl:DatatypeProperty>
```

Has_SolutionID – Properties of domain ProblemSolution with range Int

```
<owl:DatatypeProperty
rdf:about="&Ontology1373265476454;Has_SolutionID">
        <rdfs:domain
rdf:resource="&Ontology1373265476454;ProblemSolution"/>
        <rdfs:range rdf:resource="&xsd;int"/>
    </owl:DatatypeProperty>
```

G. Anomaly Validity Check

The anomalies check is the final step of the construction process to check the inconsistency of the ontology design. Reasoner is the tool used to check consistency. Using a reasoner Pellet, the process took 0.756 seconds to check the consistency concept. The overall MSSNP ontology is illustrated in Figure 10 using OWLViz.

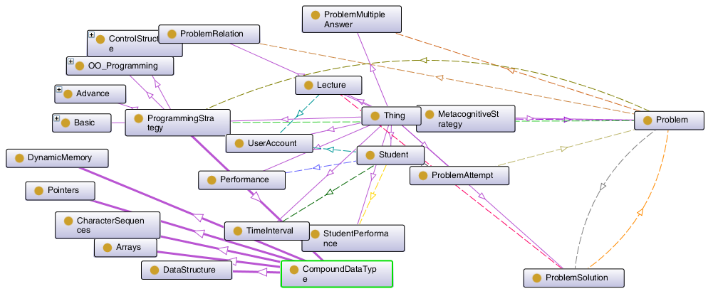

Figure 10. Final look of MSSNP Ontology

III. CONCLUSION

In this research work, an approach for constructing the MSSNP ontology structure is presented. Seven steps and criteria have been considered in developing the MSSNP ontology. This involved the scope and boundary definition, reuse consideration, class design, taxonomic identification, property identification, data property identification and anomaly validity check. In the early stage of the construction phase, we defined the scope of MSSNP that covered the five stages – pre-task, familiarization, production, evaluation and post-task. In designing the MSSNP, we have identified the possible competency questions for each stage to achieve its objectives. The data type properties and object properties have also been constructed in the structure. The two object properties involved in MSSNP ontology are inverse and functional properties. To verify the consistency of constructing the ontology, the Pellet reasoner is employed. This task takes part in the final stage of the construction process. Next, through OWLViz, the structure of MSSNP ontology can be viewed graphically.

REFERENCES

[1] Barker, P.: 'Developing Teaching Webs: Advantages, Problems and Pitfalls', in Editor (Ed.)^(Eds.): 'Book Developing Teaching Webs: Advantages, Problems and Pitfalls' (2000, edn.), pp. 89-94

[2] Drucker, P.: 'Need to know: Integrating e-learning with high velocity value chains', A Delphi Group White Paper, 2000, pp. 1-12

[3] Gómez-Pérez, A., and Corcho, O.: 'Ontology languages for the semantic web', Intelligent Systems, IEEE, 2002, 17, (1), pp. 54-60

[4] Grüninger, M., and Fox, M.S.: 'Methodology for the Design and Evaluation of Ontologies', 1995

[5] Heflin, J., and Hendler, J.: 'A portrait of the Semantic Web in action', Intelligent Systems, IEEE, 2001, 16, (2), pp. 54-59

[6] Learn, H.P.: 'Brain, Mind, Experience, and School', Committee on Developments in the Science of Learning, 2000

[7] McIlraith, S.A., Son, T.C., and Zeng, H.: 'Semantic web services', Intelligent Systems, IEEE, 2001, 16, (2), pp. 46-53

[8] Noy, N.F., and McGuinness, D.L.: 'Ontology development 101: A guide to creating your first ontology', in Editor (Ed.)^(Eds.): 'Book Ontology development 101: A guide to creating your first ontology' (Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880, 2001, edn.), pp.

[9] Rosch, E.: 'Principles of categorization', Concepts: core readings, 1999, pp. 189-206