DOMAIN MODELING TO SUPPORT ANTI-CYBER CRIME EDUCATION

Yordan Penev, Svetlozar Markov

Abstract: *This paper describes an approach to a computer-based learning of educational material. We define a model for the class of subjects of our interest - teaching of investigation and prevention of computer crimes, (those including both theoretical and practical issues). From this model, specific content outlines can be derived as subclasses and then instanced into actual domains. The last step consists in generating interactive documents, which use the instanced domain. Students can explore these documents through a web browser. Thus, an interactive learning scenario is created. This approach allows reusing and adapting the contents to a variety of situations, students and teaching purposes.*

Keywords: *e-Learning, Cyber Crime.*

ACM Classification Keywords*: K.3.2 Computer and Information Science Education.*

Introduction

Our main purpose is to develop educational activities through Computer based learning. That means creating a series of interactive documents that convey the subject matter in a proper way to our learners. The technology can help to make this work easier in two ways.

On the one hand, it allows a broader, faster and more effective spreading of this material which, in turn, enhances its usefulness for teaching to more students. On the other hand, it facilitates adapting and reusing the resulting material to a variety of situations and for a range of didactic purposes. Therefore, we have set the emphasis in developing a richer and more interactive material rather than copying the usual model based on lecturing. Technology alone can not cope with these goals. It is to be laid upon a solid ground to get the most out of it. Two well-known principles that we have taken into account are the level of abstraction and reusability. They have been widely proposed, for instance, in the work of Merril, Wenger and Mayorga from the Instructional Design perspective, in the works of Maglajlic and Maurer from the educational multimedia viewpoint. Following this line of thought, our idea is to have a domain model separated from the media technologies that will be used to navigate through it.

In fact, we set up taxonomy of domain models from the most generic description to the one that has the actual contents to be learnt. The rationale is to have a description for a class of subjects (a metamodel) which can be refined into a number of subclasses that represent specific domains - domain models, instructional models, cooperative models and structural models. Each subclass can be instanced to hold the actual and usable contents. Structured documents addressing particular educational purposes can be created from these domain instances. These documents include text and a variety of references to objects within the domain model (such as topics, problems, descriptions, hints or common misconceptions among others) which hold the appropriate information. We claim this approach to be both flexible and suited for developing educational activities through computer based scenarios.

The Meta-Model as a Generic Description of Information

A meta-model is a high-level generic description of information that draws a class of subjects. This representation can be refined into any number of domain outlines and is shaped in terms of entities and relationships between them. Thus, it defines a class of models that set up the entities and relationships needed to specify a particular knowledge field. Meta-models contain a number of domains that are also types of models that have their own specific entities and relationships. For each domain, the metamodel has its list of entities and their attributes, which stand for their properties. A domain can incorporate entities that belong to other ones as well as adding some extra attributes to them.

Hence, defining a domain means stating its entities and the relationships that can be held between them. The first ones characterize the objects which are relevant for the subject matter to specify. Any two entities will differ in the values of their respective attributes. The second ones hold the connections between those objects and can be divided into two classes.

Structural relationships allow setting up taxonomies between entities (such as class-subclass or metonymy). They hold the transitive property, which will allow making inferences through the object taxonomy. As a common rule, there will be, at least, two structural relationships: **type-of** and **part-of**. Domain-specific relationships have a meaning associated only to a particular subject and hence they carry their own semantics. As a metamodel gathers the commonalities of its derived models, it sets the pattern from which all of them inherit their shape. Thus, its usage is, mainly, to be instanced to create a domain model. Nevertheless, it could be useful also for spreading any necessary modifications among its subclasses. The possibility of changing the type of objects and relations at the meta-level makes building up and maintaining the domain easier.

As an example and study case we have created a metamodel for defining subjects in the combating of computer crime field. Those subjects share the feature of having a formal corpus of theoretical topics which should be applied to solve practical problems. We have used three domains to describe that metamodel: *the conceptual domain* holds the contents to be learnt.

The instructional domain includes the objects which will be used to perform the actual teaching and learning processes. The didactic model allows relating the other ones. Its elements are actually references to objects that belong to the other domains to which some new attributes are added. These extra properties permit assessing if the objects suit a particular didactic purpose. In the instructional domain, for example, we can see that a problem can have sub-problems through the structural relationship part-of or that a solution can be linked to a multiple-choice-question or a problem.

Models

A model is an instance of the metamodel and therefore a class derived from it (which is its super-class). It defines a knowledge structure which will be filled with actual data as the model is instanced. It includes a description of a subject matter in terms of elementary units and their relationships. Those constituents have their own meaning and purpose as objects within the context of a specific domain.

The domain model supplies an explicit characterization and a flexible access to the contents which could be retrieved in a variety of ways. They are also defined to be reusable and therefore to ease the creation of new domain models and didactic materials. A library of generic models providing a range of ontology - which could be adapted to a wide variety of subjects-would help authors and is being created.

An Instance of a Model: Network Evidence of Computer Fraud

As an application example of the approach described before we have developed a course on Network evidence of computer fraud. The goal for the students of this subject is to be able to identify locations and collect evidence on network servers where they can see and correctly identify evidence of computer crimes and hostile activity and to prove their correctness. The course includes a theoretical corpus of topics and techniques that are to be applied to investigate those crimes. Hence, theoretical issues are the grounds on which the applied knowledge is based in this practical course. In order to cope with these two perspectives, we have defined two different domain models (conceptual and instructional ones) and a didactic layer which stays upon them, following the meta-models shown in section 2.

The Conceptual Domain Model

This model collects the theoretical corpus stated before. It covers a number of topics dealing with both, declarative and procedural knowledge. There are two kinds of objects belonging to this domain: concepts and activities. The concepts represent the topics stated as declarative knowledge. They describe the basic vocabulary of the field providing accurate definitions for every relevant term. Examples of concepts are specification, precondition, post-condition, predicate or variable.

The activities represent the procedures that are to be learned and then performed by the students. They take a number of concepts and produce a result, which will be another concept. Examples of activities are specify, derive, verify or prove. Examples of how activities and concepts relate to each other could be “To specify means taking a sentence by enunciating the requisites and desired results for a network computer crime investigation and to produce a formal result - collect evidence”. The attributes of both entities are their name (a unique identifier) and their definition (a rather formal text that describes the topic). The information model for the conceptual domain also includes a variety of relationships.

Structural relationships (part-of, sub-activity) make up taxonomies for the concepts and the activities. There is a range of domain-specific relationships which define the semantics of this subject: Belong-To, Induce, Produce or Apply-To. For instance, a Precondition Belongs-To a Specification, a Predicate Induces a Set-of-States, (To) Specify Produces a Specification (as its result) or (To) Verify Applies-To computer crime.

The Instructional Domain Model

The actual learning of the topics that belong to the conceptual domain model requires a range of objects which are the contents of the instructional domain model. These objects include those ones used for illustrating concepts, practicing procedures or evaluating the student knowledge.

The information model for the instructional domain uses three kinds of objects: explanatory, exploratory and evaluative ones. The explanatory objects contain complementary information to help explaining a given topic. If there was such a type of information available for a topic, one of these objects could be associated to that particular concept or activity. Those explanations can play a number of different roles depending on their instructional purpose. Some explanatory objects can be associated to conceptual domain objects (as in the case of the examples or the common mistakes). Some others relate to other instructional objects (for instance, the hints, which could be associated to problems, or the explanations, which can be associated to any instructional object). Furthermore, there are instructional objects that can accompany any object within the domain (as the descriptions).

The exploratory objects allow the student to navigate through the domain and to practice the procedures that he should learn to apply. For the sake of convenience, we have just one type of such objects: the problem. A problem has as attributes: name, question, which enunciates the exercise to be solved, and location, which refers to a printed collection made available for our students.

Finally, the evaluative objects allow the student to self-assess her proficiency on the domain. A problem can also be an evaluative object; in that case, the student would not be offered the answer in advance. Besides, there are multiple-choice questions which represent small problems with a number of possible answers. The student has to choose just one option as the correct solution to the question. Those objects attributes are: name, question, number of options, list of options and correct answer.

The relationships that connect the instructional entities are Part-Of (which links, for instance, a problem to its sub-problems) and Is-Solution (which relates a problem to one solution). An author can ask the instructional domain about the different attributes of its objects or it is possible to retrieve objects that hold given properties (for instance, “obtain a set of n multiple- choice questions having m options each one”). The relationships allow retrieving the sub-problems or the solutions of a given problem.

The Didactic Domain Model

This model holds the information about the didactic usage and quality of the entities belonging to the other domains. Therefore, it represents a meta-layer over those ones. Its elements are entities which include a reference to conceptual or instructional objects. These units add a number of didactic attributes to those entities. There are two kinds of didactic entities, related to either the conceptual or the instructional entities. The former ones add, as new didactic attributes, the difficulty of the entity and the acquisition level (a measure of the importance of the entity, as a part of the learning process, to the student). The latter ones add the difficulty to the instructional attributes that the entity already had. The didactic relationships can be split into three categories depending on the domains that they connect.

The conceptual entities can be connected by means of the prerequisite relationship which shows conceptual dependencies between its subject and object. If a concept is prerequisite of another one, the former is to be studied before the latter one. The instructional entities can be related by means of two didactic relationships: *Describe* and *Explain*. Describe links a description or example to a problem. Explain connects a hint or an explanation to a problem. This relationship has an attribute called role which shows the intended purpose of the explanatory object. The allowed roles are focus-on, clarify, choose, discard, illustrate and reformulate. Finally, Evaluate connects an evaluative entity to a conceptual one. It allows assessing the knowledge of the student concerning that concept or activity.

The model for the didactic domain allows retrieving data such as “the prerequisites of a concept”, “the available hints to solve a problem”, “is there any available reformulation for a given problem?” or “what is the reason for using that particular solving method”).

We created student profiles by taking into account their performance while they carry out different learning activities. Examples of such documents are self-evaluation tests, programming projects or study guides. The tests help the students to monitor their learning process. The programming projects are structured as questionnaires that the students have to work out and then submit. Study guides propose a tour through the subject matter. They are organized into sections following the topics of the regular course design. They include, for each and every of their sections, a list of the most relevant topics with their descriptions and recommended readings, related exercises, common misconceptions, questions to think about, or some directions to organize their study.

Conclusion

Our environment is a traditional University environment where the high rate of students per teacher prevents a highly individual tuition. Our goal is improving the support to the students. Hence, we bear in mind the introduction of the new information technologies into our set of basic teaching tools. In order to achieve it, we are developing ways of providing interactive learning support for our students. The technology provides the students with the accessibility and the adequate learning material.

Our approach supports the educational authoring process. It provides the author with a wide amount of teaching material and a flexible way to use it. The domain description is based on a conceptualization work that has allowed creating taxonomy of domain models. Modeling means defining the domains in terms of relevant and useful objects that can be retrieved in a number of ways. We first created a metamodel from which domain models can be instanced. Those domain models can be filled with the actual subject matter contents. Document types addressing particular teaching purposes can be created. Those documents define structured templates and learning paths. What we achieved is that now it is possible to create many learning scenarios sharing the same objects but having different teaching purposes and didactic perspectives. Facilitating reusability is a way of reducing the cost of developing applications to teach. It also helps the authors get the most out of the materials they create. These documents have a fixed part of text and a variety of references to objects (topics, problems, descriptions, hints or common misconceptions among others) which hold the appropriate information.

Bibliography

[1] Maglajlic, S., Maurer, H. and Scherbackov, N. (1998), Separating Structure and Content: Authoring Educational Web Applications. In Proceedings of the ED-Media & ED-Telecom 98, pages 880-884.

[2] Mayorga, J.I. and Verdejo, M.F. (1996), Authoring Systems Revisited: The Software Design Cycle Metaphor. In Paiva, A., Brna, P. and Self, J., eds., European Conference on Artificial Intelligence in Education.

[3] Merril, M.D. (1987), The New Component Design Theory: Instructional Design for Courseware Authoring. Instructional Science, 16:19-34.

[4] Schwabe, D. and Rossi, G. (1995), Object-Oriented Hypermedia Design Method. Communications of the ACM, 8 (8):45-46.

[5] Wenger, E. (1987), Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge. Morgan-Kaufman.

Authors' Information

Yordan Penev – Associate Professor, Academy of the Ministry of the Interior, Chief of Department for Combating Crime, Sofia, Mladost-4, Bul. A. Malinov, N 1, ypenev@dir.bg

Svetlozar Markov – Assistant Professor, Academy of the Ministry of the Interior, Sofia, Mladost-4, Bul. A.Malinov N 1, smarkonius@gmail.com