6. Conclusion

One of the most important issues students should dwell on when solving applied problems are whether the required quantity is physical or non-physical and if the unit of the quantity in question is known or not, etc. Any computation in a specific problem should commence only after all quantities are converted into the right units according to SI, if no other system of units is referred to explicitly. After solving a specific problem they should do a check through the physical dimensions of the output quantities. If all is correct, then this is a guarantee of a right solution. Solving applied problems enhances students' engineering thought, i.e. their ability to observe links between Maths and Physics, on the one side, and various technical applications of these sciences, on the other, also to envisage possibilities for applying this knowledge to practice and to realize scientific ideas on a practical level.

Bibliography

[1] "A Dictionary of Units of Measurement" http://www.unc.edu/~rowlett/units/index.html
[2] "Index to Units & Systems of Units" http://www.sizes.com/units/index.htm
[3] Engineering and scientific calculations http://www.megabook.ru/pc/encyclop.asp?topic=pc_542&rubr=pc_542
[4] Kovacheva Ts. "Mathematical packages for teaching and research in Internet applications and information support", "Information theories@Application", v.11,4,2004, 387-393.
[5] "Maple Application Center", "Waterloo Maple Software"www.maplesoft.com
[6] "New packets of the system Maple 7" http://ais.khstu.ru/Maple7/Glava%2016/Index40.htm
[7] "Products and Solution" www.maplesoft.com
[8] "Reference book for Units" http://www.bgnews.bg/media.html?media=47549806
[9] "The Units Package in Maple" http://www.adeptscience.co.uk/download/dlddsp/9304/0/All/A4+Maple+Units+Poster.html
[10] Units in Mathcad & Maple http://twt.mpei.ac.ru/ochkov/work2_eng.htm
[11] "SI Units, Conversion, Measurement skill" http://www.wacek.co.za/

Author's Information

d-r Tsvetanka Kovacheva – Lecturer at Department Mathematics of Technical University, Bulgaria, 9010 Varna, 1 Studentska str., e-mail: Tsveta_Kovacheva@tu-varna.acad.bg

ABOUT ONTOLOGY APPLICATION TO THE DESCRIPTION OF SYLLABUS

Evgeny Eremin

Abstract: Publication describes the experience in application of ontology technique to structuring of educational materials. Several topics of physics were formalized by means of Protégé *software tool. Some principal problems in building of knowledge structure were found, so the discussion may interest not only ontology users, but also the developers of ontology tools.*

Keywords: ontology, program, education, course, Protégé, knowledge structure, objects, inheritance, classes.

ACM Classification Keywords: K.3.1 Computer Uses in Education – Computer-assisted instruction; E.1 Data Structures – Trees; E.2 Data Storage Representations – Object representation; I.2.4 Knowledge Representation Formalisms and Methods – Frames and scripts; I.2.6 Learning – Knowledge acquisition.

Introduction

At present time **ontologies** – formal descriptions of terms and relations between them in some knowledge domain – are increasingly using for structuring of the comprehensive expertise, accumulated by mankind, and its transformation into computer representation. This method of structure storage already has many functioning complete applications in various areas. Perspective theoretic studies, aimed on support of the correct semantic search in electronic documents (including data selection by net agents) and automatic building of ontologies from found information, are also of great importance. Numerous examples of ontologies and their practical applications were more than once described in literature [1–4].

Ontologies essentially facilitate the mutual understanding between people jointly using information. Furthermore knowledge representation in a form of ontology not only makes possible its automatic processing, but permits people to formulate their experience in some domain in the clearest and the most demonstrative way.

The last fact is very interesting from educational point of view. Learning, being a process of purposeful knowledge transfer, belongs to the fields of human activities where ontologies are natural (see [3] for examples). In particular, creating of the effective automated learning systems strongly depends from the success in structuring of knowledge and its representation in the forms available to computer.

This paper describes an attempt to apply ontology for structuring of some topics from the educational course. Usually most examples of such ontologies are developed for programming languages (see publications [5, 6] for example), where all the results look clear and elegant because these languages are artificial design and so well-structured. Publication [5] even raises a question about the harmony of the built ontology. But other knowledge domains are not so clean, as it results form [7, 8] where the analyses of ontologies developed for various educational disciplines are described.

Several chapters from physics were considered in this paper for structuring of educational materials (to simplify the study, they were taken from the textbook for the last classes of secondary school). Although the choice of the topics was made according to the author's preferences, it has some reasons: physics is the thing for ontology application. From discussing point of view this discipline is very interesting because it mirrors objective complexity and interdependence of the natural phenomena; besides, the way, in which physic knowledge is structured, determines learning strategy in many respects [9, 10]. In consequence of the importance of physical basics conceptual systematization, paper [10] offers a special course in teachers training, aimed «not to teach more physics but to organise what have already been learnt».

Package Protégé v. 3.1.1 (Protégé-2005) was used for creating and keeping educational ontologies. Software choice was determined by the renown of this package and also by the existence of accessible detailed descriptions [4, 11–13].

Statement of the problem

The problem under consideration naturally arises from everyday pedagogical activities. Every teaching course always has a syllabus – some list of topics and subjects to learn (in Russia syllabus is an official document, although it's not dogma). When it is written for a course taught for a long time and supported by the set of time-proved textbooks of various authors (let's mention physics and geometry as examples), it's not difficult to realize such scheme. Furthermore, practical experience shows that existent changes in this case are not too often and usually not essential. Situation with rapidly developing computer disciplines is quite different: syllabuses are changing at every turn, so textbooks are often incomplete and can't timely catch all movements in science; the experience of teaching for nascent topics does not exist. As a result, the questions that were added to syllabus during regular renewal are unfamiliar to teachers and sometimes they can't neatly imagine where to find the material for these chapters of the course. In such cases teachers strongly lack for several phrases explaining the topic, together with reference list closely connected with this concrete question. So they naturally wish to get additional information on any syllabus item, in other words, using pedagogical terminology, every item must provide brief explanation of its subject.

Possible computer solution of the problem lies in building of educational course's ontology, which contains comments and references at every point. As an additional advantage of ontology approach (in comparison with the «paper» solution), we visualize interrelations between subjects of our course, that is very useful for organization of well-timed repetition and planning of exposition order.

As it was mentioned above, this work deals with ontologies for the course of physics. The stable Russian schoolbook [14] was used as a resource for building ontology. We'll not consider questions, connected with plural possibilities of material exposition here. Using the terms from publications about ontologies [6, 8], we plan to create the ontology of the educational course but not of the correspondent knowledge domain, at that in the simplified version – using the only textbook.

Principles of ontology building

Considering some ambiguity of existing terminology about ontology's components, let us name the main terms that will be used in the discussion: in fact it is the denomination system, on which Protégé software is based.

Ontology is built from classes, slots and instances. **Classes** describes individual concepts of the knowledge domain, and their **instances** are examples of concrete objects realization. **Slots** describe properties and attributes; they can be obtained both to classes and instances as well. For example we can name terms *concept* or *law* as typical classes of educational material. Representative slot of the *concept* class is *definition*, and the *law* class may contain specific slot *mathematical expression*. Instances of *concept* are *electric current* and *electric charge*; the *law* class specifically realizes as *Ohm Law* and *Joule-Lentz law*, or maybe texts about these objects.

It is worthy of note that only definite interrelations between mentioned above categories are admissible in Protégé software [12].

As a rule, new class may be reproduced from parent class. Concrete instance also descends from correspondent class (some classes, called **abstract**, deny this possibility), but it must be terminal node of any ontology, i.e. instance principally can't have inheritors. Protégé also supports **multiply inheritance**, when some class has several parents and inherits all their slots.

Slots are determined independently from any class or instance (this allows using the same slot in different branches of hierarchy). There are two types of slots in Protégé: **own slots** and **template slots**. First are attached to their class or instance and capable to store an individual value. Second type belongs to class only, and all such slots are inherited. Template slots in a class are unable to have value in substance, until will be passed on to concrete instance; here they become the own slots and hence get property to be filled with value. Slots may be added to the class only, and instances get slots inheriting them from the class.

Each class is associated rigidly with interface **form**, by means of which user fills required slot values. Form's view is easy to edit, so it may be composed in any usable for input look.

Later, describing difficulties in building of our ontology for educational material from physics, we'll need these data about ontology's components interrelations.

Ontology realization and its difficulties

Even preliminary analysis of the knowledge domain for our problem displays, that three kinds of classes as minimum are necessary:

- type of educational material (*description* or *law* as an example); it determines what data components are stored for every variety of syllabus questions;
- fundamental concepts from the whole physics course which are used in all of its parts (physical magnitude, unit, system of units etc.);
- categories, comprised in the concrete part of physics (electricity and magnetism were selected for the trial ontology).

Existence of several closely interrelated class levels in itself generates significant difficulties. Essential interdependence between fundamental physical concepts make situation much more complicated, because such complex relations not always keep within hierarchical structure. Let us consider an example from fundamental categories of physics. Some *system of units* is used for measuring a *physical magnitude*. This *system of units* consists from an aggregate of definite *units*, and every *system of units* contains its own set of *units*. Some *units* may be included into different *systems of units*. Withal many *physical magnitudes* can be measured by several *units*, and selected *unit* in its turn depends from *systems of units*. At last the process of measurement itself lies in the comparison of measuring *physical magnitude* with some reference *physical magnitude* which is defined as a *unit*. We must emphasize, that we can't go without enumerate above concepts, because every physical magnitude must be measured in some units.

We must agree with the authors of publication [7], who accentuate that classification of educational content in the form of hierarchical structure is one of the main difficulties. The reason is apparently principled and lies in the nature of a learning material. As it was noted in [3, 15], five types of relations between terms exist: «part–whole», e.g. bumper and a car; «collocation», e.g. words in the sentence; «paradigmatic relations», e.g. Sun and Solar system; synonyms and antonyms. G. Booch in his fundamental book [16] adduce somewhat different list of basic relations between classes: class/subclass («is–a») – rose is a flower; whole/part («part of») – petal is a part of rose; semantic relations, associations (rose and candles both can be used for table decoration). It is evident that in spite of some difference in classifications, the semantic relations in both are the worst for univocal tree formalism versus all the rest.

The analysis of the physics categories' tree also demonstrates that hierarchy of concepts not always determines the order of their learning. For example the topic Ohm law for the part of the circuit use to be learnt before the similar law for the whole circuit, although from hierarchical position the first one is a descendant from the second.

If we remember about the existing of two opposite ways in cognition – induction and deduction, then some limitation of univocal tree ontologies (at least as applied to educational process) become more distinct.

Another principled difficulty in building educational ontology springs out from «heterogeneity» of all real learning materials. In the same textbook chapter we often find closely a definition of some physical magnitude, some laws in which it is involved, and also the description of these laws' application to human activity. As a result it is not easy to build unified ontology for all various fragments of knowledge.

Let us consider one more example. Fig. 1 shows a small fragment of developing ontology for the topic «Electric current».

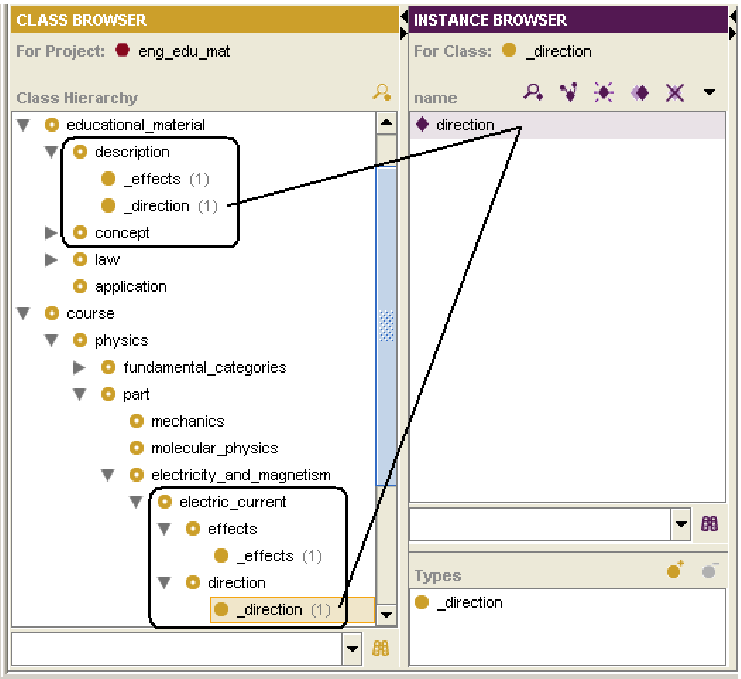

Fig. 1

In the top part of this ontology we see the class *educational_material*; its subclasses (*description*, *concept*, *law* and others) correspond to the pieces of the learning texts that form any concrete physical topic. Every type of the material has definite set of slots, which is inherited according to the basic ontology's principles. For example the specific slot for *concept* is *definition* (slots are not shown on fig.1).

Now let's advert to the class *electric_current*, located in the bottom part of the figure. In substance it is *concept*, so we ought to add the slots for this kind of educational material, using multiply inheritance. But at the same time two next subclasses – *effects* and *direction* – serve as *descriptions*, hence they do not need slot like *definition* from the superclass, which is the descendent of *concept*. Nevertheless, subclasses can not refuse from inheriting slots. Thus we see the main reason of this contradiction: class must have instrumentation not only for the extension of its own structure, but for its restriction as well [16].

The possible solution, which stays within Protégé knowledge model, is to add necessary slots from subclasses of *educational_material* on the very last stage, i.e. just before creating concrete instance of a class. As a result, we may create accessorial class (*_direction* for *direction* superclass on fig. 1), which doesn't take part in the common hierarchy, and add the second parent superclass *description* to it. In fine, *_direction* (peculiar «mixer» for the required slots) gets its own set of slots, specific for this material, without any influence on inheritance process because of absence of the descendant classes. But this mixed class allows producing an instance of educational material with all necessary properties.

Described above accessorial classes are marked on the hierarchical scheme on fig. 1 by different graphic symbol – fully fill circle: this sign is used in Protégé for concrete classes, which are allowed to have instances. Alternative classes that produce only classes are called abstract and marked by the circle with white center. From the point of view of the described model such difference is clear, because instances are created only from classes with preliminary prepared total set of slots.

Such method of the multiply inheritance was suggested in Flavors language (citing from [16]): small classes, unappropriated for producing instances, were mixed to other classes, providing them more complex structure.

Such method is called **creating admixture** (mixin). Subclasses, produced from *educational_material* class, play a role of admixture in our ontology.

In fact Protégé allows to manage without accessorial classes like *_direction* in the above example, mixing slots directly in the instance (see the bottom of the instance browser on fig. 1). This solution, quite suitable for practice, seems us slightly inconsecutive from the positions of the conventional building of full class hierarchy.

Seeing principled character of these difficulties with restrictions of class inheritance, we may think about improvement of the ontology keeping systems themselves. Several ways of the improvement may be offered.

First of all, from theoretical point of view it looks winning to introduce a role for slots similar it is done for abstract and concrete classes. Considering that Protégé already has two types of slots (template slot and own slot), it is easy to generalize formally inheritance model and define new type, called, say, private slot. As it is clear from the following table, this new type of slot, in contrast to the own one, will pass to the instances of the class, but will not be inherited by its subclasses.

Slot type	Class inheritance	Instance inheritance
private?	–	+
own	+	–
template	+	+

Maybe such generalization will conflict with Protégé knowledge model. As a variant, the mentioned above way of temporary mixing of another class without including its slots into inheriting template may be suggested.

It is worth to mention about prevalent in OOP **hidden** (private) properties, which are inherited, but «not visible» in the subclasses-descendants. Something similar may be done for slots, but great amount of hidden unused slots will make tables of slots boundless.

At last some variant template for slots may be offered for realization – an analog for variant record in Pascal, where an actual set of fields depends from the value of tag field (from *educational material* in our case).

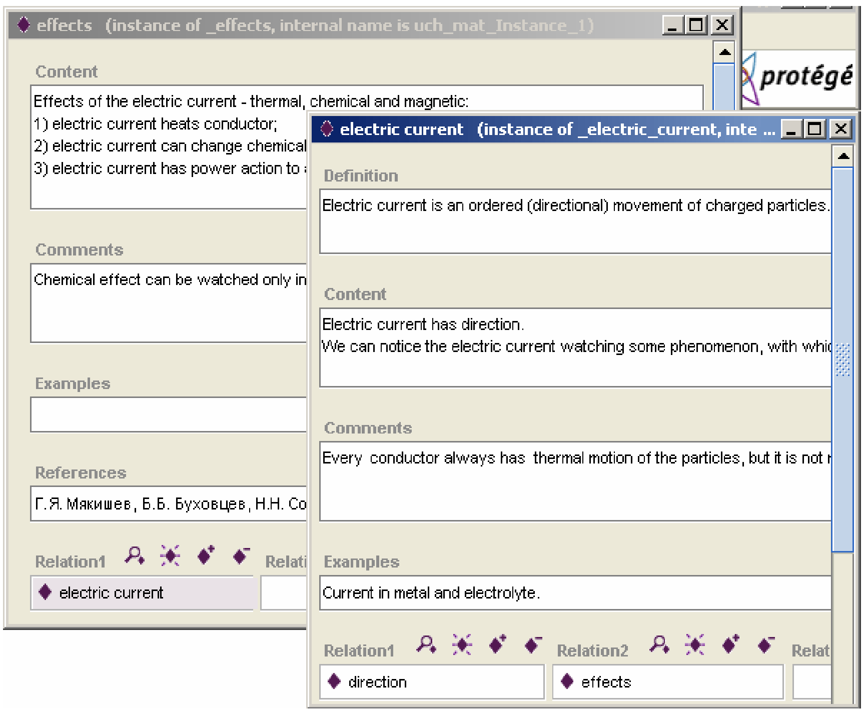

Fig. 2

Realization of links inside material

Very important advantage of educational material in electronic form is the possibility to see related topics while reviewing it. Protégé package provides a simple but usable mechanism for realization of such references – the possibility of slot to have an instance as its value. Consequently, slot of such type, placed on form with the material, becomes the reference to another material: remember that in the terms of our ontology concrete fragments of educational texts are instances, so it is a question of links between instances of different classes. Using such slots on material's form, we can review linked instance by means of mouse double click. The obtained realization is very close to hyperlink, which, being clicked, opens connected resource in a new browser window.

Fig. 2 demonstrates a form with question «Effects of the electric current». In the top window related fragment «Electric current» was opened; reference slot of the same name (see field *Relation1* in the bottom form) was the source of call.

Forms on fig. 2 also represent a possible way how explanation of the syllabus items' contents can look. Let us note the special importance of the form field called *References* – it can be found in the left bottom part of fig. 2 and contains the reference to Russian textbook [14].

Future work

At present time the common basis of the ontology is built and class hierarchy for several topics of physics course are created. Some additional theoretic interest may be found in continuing of work, aimed on realization of ontology for several different parts of physics, because their interrelations may arise some new specific problems.

Similar work for other disciplines should be done in perspective with the purpose of comparison the result ontologies. The hypotheses exist that such results may help to choose some objective elements for estimation of the complexity of learning materials, and also for the description of courses' organization and internal structure.

Bibliography

[1] A. Gómez-Pérez, M. Fernández-López, O. Corcho. Ontological Engineering with Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web. – Springer-Verlag, London, 2004.

[2] A. Maedche. Ontology Learning for Semantic Web. – Kluwer Academic Publishers, Boston, 2002.

[3] J. Milam. Ontologies in High Education. In: Knowledge Management and Higher Education: A Critical Analysis. Ed. Metcalfe A.S., Idea Group Publishing, Hershey, PA, 2005. – P. 34–62.

[4] N.F. Noy, D.L. McGuinnnes. 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. [Electronic resource] / Accessible at: http://protege.stanford.edu/publications/ontology_development/ontology101.html

[5] S. Sosnovsky, T. Gavrilova. Development of Educational Ontology for C-programming // Information Theories & Applications. – 2005. – 13 (4). – P. 303–308.

[6] K. Olsevicova. Topic Maps e-Learning Portal Development // The Electronic Journal of e-Learning. – 2006. – 4 (1). – P. 59–66; available online at www.ejel.org

[7] D. Dicheva, C. Dichev. Authoring Educational Topic Maps: Can We Make It Easier? // Proceedings of ICALT. – 2005. – P. 216–219.

[8] D. Dicheva, C. Dichev. Confronting Some Ontology-building Problems in Educational Topic Map Authoring // Proceedings of the Workshop on Applications of Semantic Web in E-Learning (SW-EL@AH'06), in conjunction with AH. – 2006. – P. 55–64.

[9] N.W. Brickhouse. Teachers' beliefs about the nature of science and their relationship to classroom practice // Journal of Teacher Education. – 1990. – 41 (3). – P. 53–62.

[10] I.T. Koponen, T. Mäntylä., J. Lavonen. Challenges of Web-based Education in Physic Teachers' Training // Proceedings of ICTE. – 2002. – P. 291–295.

[11] The Protégé Project [Electronic resource] / Accessible at: http://protege.stanford.edu

[12] N.F. Noy, R.W. Fergerson, M.A. Musen. The Knowledge Model of Protégé-2000: Combining Interoperability and Flexibility // Proceedings of EKAW 2000, Lecture Notes in Artificial Intelligence, no. 1937. – Springer-Verlag, Berlin. – P. 17–32.

[13] N.F. Noy, M. Sintek, S. Decker, M. Crubézy, R.W. Fergerson, M.A. Musen. Creating Semantic Web Contents with Protégé-2000 // IEEE Intelligent Systems. – 2001. – 16 (2). – P. 60–71.

[14] G.Ya. Myakishev, B.B. Buchovcev, N.N. Sotskiy. Physics: Textbook for the 10 form of secondary school. – Prosveshenie, Moscow, 2002 (in Russian).

[15] A. Papadopoullos. Meaningful Search: Why PET Scanners are not about Cats & Dogs. – Convera, Carlsbad, CA, 2003.

[16] G. Booch. Object-Oriented Analysis and Design with Applications. Addison-Wesley Publishing Company, 1993.

Author's Information

Evgeny Eremin – Perm State Pedagogical University; Russia, 614990, Perm, Sibirskaya str., 24; e-mail: eremin@pspu.ac.ru