[13] A. Syropoulos, E.G. Mamatas, P.C. Allilomes, K.T. Sotiriades, *A distributed simulation of P systems* , Artiom Alhazov, Carlos Martín-Vide and Gheorghe Paun (eds): Preproceedings of the Workshop on Membrane Computing; Tarragona, July 17-22 2003, 455-460

[14] http://psystems.disco.unimib.it/

Authors' Information

Fernando Arroyo – Dpto. Lenguajes, Proyectos y Sistemas Informáticos- Escuela Universitaria de Informática; Universidad Politécnica de Madrid - Crta. de Valencia km. 7 - 28031 Madrid SPAIN - farroyo@eui.upm.es

Carmen Luengo – Dpto. Lenguajes, Proyectos y Sistemas Informáticos- Escuela Universitaria de Informática; Universidad Politécnica de Madrid - Crta. de Valencia km. 7 - 28031 Madrid SPAIN - cluengo@eui.upm.es

Luis Fernandez – Dpto. Lenguajes, Proyectos y Sistemas Informáticos- Escuela Universitaria de Informática; Universidad Politécnica de Madrid - Crta. de Valencia km. 7 - 28031 Madrid SPAIN - setillo@eui.upm.es

Luis F. de Mingo – Dpto. Organización y Estructura de la Información - Escuela Universitaria de Informática; Universidad Politécnica de Madrid - Crta. de Valencia km. 7 - 28031 Madrid SPAIN - lfmingo@eui.upm.es

Juan Castellanos – Dpto. Inteligencia Artificial - Facultad de Informática - Universidad Politécnica de Madrid; 28660 Boadilla del Monte - Madrid SPAIN - jcastellanos@fi.upm.es

TEACHING STRATEGIES AND ONTOLOGIES FOR E-LEARNING [1]

Tatiana Gavrilova, Michael Kurochkin, and Victor Veremiev

Abstract. The paper presents one approach aimed at developing teaching strategies based on the principles of ontological engineering. The research framework is targeted on development of methodology and technology that will scaffold the process of knowledge structuring for e-learning. The structuring procedure is the kernel of ontology development. Ontologies that describe the main concepts of the domains are used both for teaching and assessment techniques. Special stress is put on visual design as a powerful learning mindtool. The examples are taken from the courses on the foundations of artificial intelligence and intelligent systems development. These courses are delivered by the authors in St.Petersburg State Polytechnical University at School of Computer Science and in Poland in the First Independent University.

Keywords: E-learning, Ontologies, Visual Knowledge Engineering, Expert Systems Building Tools, Knowledge Acquisition, Knowledge Sharing and Reuse.

1. Introduction

The drawback of e-learning is lack of feedback from the teacher or tutor. That is why the courseware should be more precisely structured that in face-to-face teaching.

The idea of using visual structuring of teaching information for better understanding is not new. Concept mapping [Sowa, 1994; Jonassen, 1998, Conlon, 1997] is scaffolding the process of teaching and learning for more than 20 years. Visual representation of the general domain concepts is facilitative and helps both learning and teaching. A teacher now has to work as knowledge analyst or knowledge engineer making the skeleton of the studied discipline visible and showing the domain's conceptual structure. This structure is now called "ontology". However, ontology-based approach is rather young. It was born in knowledge engineering [Boose, 1990; Wielinga, Schreiber, Breuker, 1992], then it was transferred to knowledge management [Fensel, 2001].

[1] The work is partially suported by Russian Foundation for Basic Studies (grant 02-01-00466)

The short prehistory of knowledge engineering (KE) techniques and tools (including knowledge acquisition, conceptual structuring and representation models), the overall overview of which is presented in [Adeli, 1994; Scott, Clayton, Gibson, 1994], is an ascending way to the development of the methodology that can bridge a gap between the remarkable capacity of human brain as a knowledge store and the efforts of knowledge engineers to materialise this compiled experience of specialists in their domain of skill.

Beginning from the first steps to nowadays knowledge analysts have been slightly guided by cognitive science. So major part of KE methodology suffer of fragmentation, incoherence and shallowness.

The last years the main interest of the researchers in this field is concerned with the special tools that help knowledge capture and structuring. This generation of tools is concerned with visual knowledge mapping to facilitate knowledge sharing and reuse [Eisenstadt, Domingue, Rajan, Motta, 1990; Tu, Eriksson,. Gennari, Shahar,. Musen, 1995; Johnassen, 1998]. The problem has been partially solved by developing of knowledge repositories called ontology servers where reusable static domain knowledge is stored (e.g. projects as Ontolingva, Ontobroker, KA2, etc.)

In tutoring systems teachers are supposed to reuse the domain ontologies in order to support the description of the discipline they taught and the problem-solving methods of their domain. The idea is to allow teachers to model both domain and problem-solving knowledge using the same visual language. Ontology design also may be used as an assessment procedure. Students show their knowledge and understanding while creating ontologies.

Knowledge entities that represent static knowledge of the domain are stored in the hierarchical order in the knowledge repository and can be reused by other teachers. At the same time those knowledge entities can be also reused in description of the properties or arguments of methods of another knowledge entity. Concept maps modelling language that is designed in the framework of the described project is based on a class-based object-oriented language which is aimed to support typing and parameterisation of knowledge entities. Due to the class subsumption and polymorphism of classes the reasoning process becomes extremely flexible. This non-formal system allows to reason on a large set of knowledge and to apply problem-solving rules described for the higher level knowledge entities to the lower level knowledge entities based on the class inheritance. In contradistinction to ontology server approach where static knowledge described is very specific to the domain, the approach which is taken in the paper simplifies reusability of the dynamic knowledge and as a consequence building of large-scale knowledge bases with a flexible reasoning capability.

The proposed ideas and methods may be applied to those tutoring systems where general understanding is more important that factual details. We used such approach in teaching Artificial Intelligence, Neuroscience and Computer Graphics.

2. Ontological Engineering

An ontology is a set of distinctions we make in understanding and viewing the world. There are a lot of definitions of this milestone term [Neches et al,1991; Gruber, 1993; Guarino et al, 1995; Gomez-Peres, 1999]:

1. Ontology defines the basic terms and relations comprising the vocabulary of a topic area, as well as the rules for combining terms and relations to define extensions to the vocabulary.
2. Ontology is an explicit specification of a conceptualization.
3. Ontology as a specification of a conceptualization.
4. Ontology as an informal conceptual system.
5. Ontology as a formal semantic account.
6. Ontology as the structured vocabulary.
7. Ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base.

All these definitions together clarify the ontological approach to knowledge structuring on one hand, on the other hand give enough freedom to the open thinking. So ontological engineering gives the intuitively clear representation of company structure, staff, products and relationship among them.

Many researchers and practitioners argue about distinctions between ontology and user's conceptual model. We supposed that ontology corresponds to the analyst's view of the conceptual model, but is not the model itself.

Ontologies are useful structuring tools, in that they provide an organising axis along which every student can mentally mark his vision in the information hyper-space of domain knowledge. Rather often we can't express all the information in one ontology, so subject knowledge storage includes a set of ontologies. Some problem may occur when jumping from one ontological space to another. But constructing of meta-ontologies may help.

Ontology development also faces the knowledge acquisition bottleneck problem. The ontology developer comes up against the additional problem of not having any sufficiently tested and generalised methodologies recommending what activities to perform and at what stage of the ontology development process these activities should be performed. That is, each development team usually follows their own set of principles, design criteria and steps in the ontology development process. The absence of structured guidelines and methods hinders the development of shared and consensual ontologies within and between teams, the extension of a given ontology by others and its reuse in other ontologies and final applications [Guarino, Giaretta, 1998].

Till now, few domain-independent methodological approaches have been reported for building ontologies [Swartout, Patil, Knight, Russ, 1997; Fensel, 2000]. These methodologies have in common that they start from the identification of the purpose of the ontology and the need for domain knowledge acquisition. However, having acquired a significant amount of knowledge, major researchers propose a formal language expressing the idea as a set of intermediate representations and then generating the ontology using translators. These representations bridge the gap between how people see a domain and the languages in which ontologies are formalised. The conceptual models are implicit in the implementation codes. A reengineering process is usually required to make the conceptual models explicit. Ontological commitments and design criteria are implicit in the ontology code.

This paper proposes the most explicit way for ontology design - to use the visual representation in a form of a tree or set of trees.

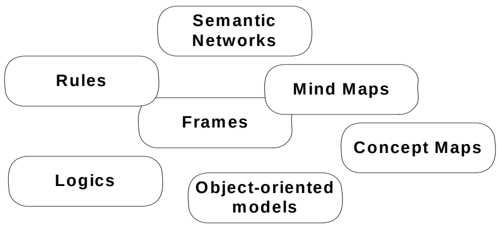

Figure 1. Unstructured set of Knowledge Representation Models

Figures 1 and 2 illustrate the idea how ontology bridge the gap between chaos of unstructured data (names of different models and techniques for knowledge representation) and clear knowledge of modern classification. Our approach shows that ontology development process needs some creative efforts of meta-concepts definition that helps to name the groups and stucture the chaos.

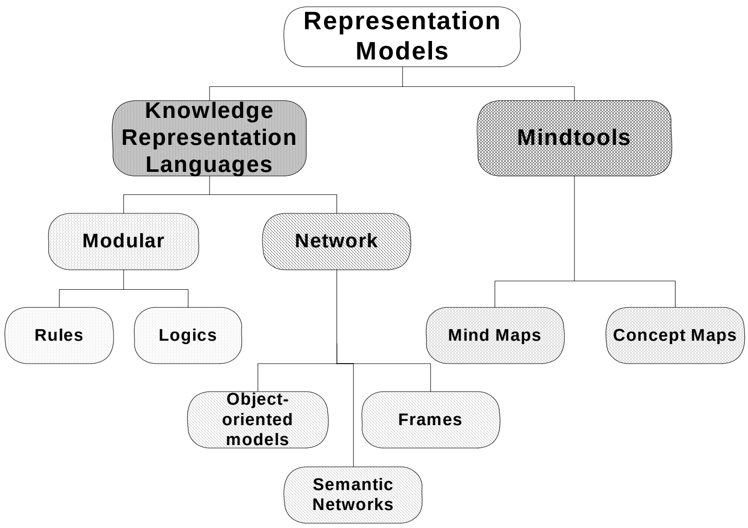

Figure 2. Ontology "Knowledge Representation Models"

Ontology developers (who are unfamiliar with or simply inexperienced in the languages in which ontologies are coded, e.g. DAML, OIL, RDF) may find it difficult to understand implemented ontologies or even to build a new ontology.

It is easier for any educationalist simply to draw the ontology using well-known to everybody “pen and pencil” technique then to study these languages.

3. Object-Structured Approach

Although there are some methods that are rather powerful and versatile [Kremer, 1998], the teacher as knowledge analyst is still weakly supported while working for a set of ontologies to describe the main subject knowledge. This process is the most important and critical stage in the courseware preparation life cycle - transition from elicitation to conceptualisation by understanding and realisation of the subject information structure and main reasoning way. The teacher may do this sophisticated g procedure alone or to ask for help of professional analysts.

In this way, a special methodology Object-Structered Analysis (OSA) has been developed [Gavrilova and Voinov, 1992-2000], which is intended to help knowledge analyst to perform the most informal step of knowledge acquisition, concluding in prior conceptual structuring of the subject domain. The approach presents the enhancement of classical structured analysis methodology [Sowa, 1994; Yourdon, 1990] to knowledge engineering.

OSA is based on decomposition of subject domain into several (3-8) strata (Tab.1). The number of strata is considered by the analyst. This multi-step and time-consuming procedure is methodological base for effective constructing of subject ontologies.

s1	*WHAT FOR* Knowledge	*Strategic* Analysis: Targets, Aims, Requirements, Constraints.
s2	*WHO* Knowledge	*Organisational or Historical* Analysis: Main Researchers, Human Resources, Actors.
s3	*WHAT* Knowledge	*Conceptual* Analysis: Main Concepts, Processes, Entities and Relationships between them.
s4	*HOW TO* Knowledge	*Functional Analysis*: Main Algorithms, Decision Procedures, Business Processes Modelling, Decision Making Models.
s5	*WHERE* Knowledge	*Spatial* Analysis: Geography, Environment, Communications, etc.
s6	*WHEN* Knowledge	*Temporal* Analysis: Historical Dates, Schedules, Time Constraints, etc.
s7	*WHY* Knowledge	*Causal* Analysis: Explanations to Decision Making Models.
s8	*HOW MUCH* Knowledge	*Economical* Analysis: Resources, Losses, Incomes, Revenues, SWAT, etc.

Filling that matrix is performed into two steps:

Step 1. Global (vertical) analysis, i.e. decomposition of the heterogeneous domain information into the groups related to mentioned above methodological strata.

Step 2. Local analysis of each individual stratum (horizontal), concluding in maintenance of gradually detailed structures. The number of levels depends on peculiarities of the subject domain and could vary dramatically for different strata. From the point of view of methodology the number of levels $n<3$ indicates ill-structured domain knowledge.

The first level (or column 2 in the table) corresponds to the discipline information as a whole. The second one corresponds to the problem that is studied now. The others may correspond to particular sub-problems, depending on the required reasonable deepness of detailing. The procedure of the described analysis may be performed both in top-down and bottom-up strategies, including their possible mixture.

The formation of strata with more of less definite meaning as described in Tab.1 allows to avoid many traditional didactic mistakes in teaching and learning. The minimal obligatory set of strata for the course structuring development is:

s3: Conceptual Structure or subject ontology.

s4: Functional Structure or main problem solving procedures.

Other strata are designed and developed if needed by subject peculiarities, e.g. spatial and temporal analysis strata (s5 and s6) may be formed in those disciplines which study construction or management where the issues of scheduling, real-time operations, real object manipulation are substantial.

Step 1 algorithms may be sketched in such form:

1.1: Gather all the data and knowledge of discipline identification

1.2: Select a set of N strata to be formed ($N \geq 3$).

1.3: For each i-th stratum select a subset of all available information, relevant to that stratum and represent it in way appropriate to that stratum (see below).

1.4: If there remains unused bulk of information, increase number of strata and repeat step 1.3. Otherwise, begin the horizontal analysis of each declared stratum.

Step 2 is horizontal analysis of strata that depends on the number of columns in OSA matrix and may be performed in two ways: deductive (top-down) and/or inductive(bottom-up). As the most essential stratum is s3 (WHAT-analysis), the horizontal analysis for it is concluded by resulting conceptual structure or a set of the domain ontologies.

Analogous algorithms were developed and practically tested and evaluated by the authors during developing of distance learning courses for different branches of computational science and for artificial intelligence (AI).

Table 1. Matrix for OSA

Level → / Stratum↓	Domain Level in general (u_1)	Problem Level (u_2)	Sub-Problem Level (u_3)	……	(u_n)
Strategic Analysis s_1	E_{11}	E_{21}	E_{31}	E_{i1}	E_{n1}
Organisational Analysis s_2	E_{21}				
Conceptual Analysis s_3	E_{31}				
Functional Analysis s_4	E_{41}				
Spatial Analysis s_5	E_{51}				
Temporal Analysis s_6	E_{61}				
Causal Analysis s_7	E_{71}				
Economical Analysis s_8	E_{81}				
…..				E_{ij}	
s_m	E_{m1}				E_{mn}

4. Teaching Ontologies in Artificial Intelligence

We have developed more than 20 teaching ontologies (What-knowledge conceptual structures s3) helping to understand and to remember main concepts of AI. Fig.3 shows one of them (it includes a part of Fig.2).

We worked out several tips to add expressiveness to the ontology on the design stage.

1. Use different font sizes for different strata
2. Use different colours to distinct the subset or branch
3. Use vertical layout of the tree
4. If needed use different shapes of nodes

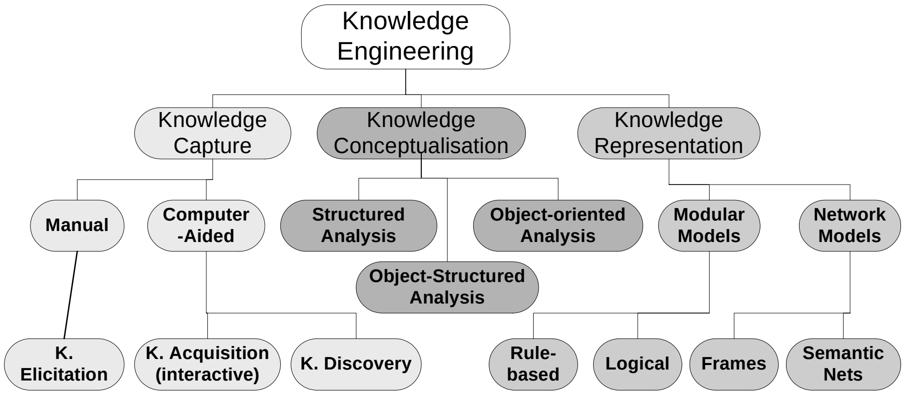

Figure 3. Ontology "Knowledge Engineering"

It is possible to use any of graphical editors to design ontology, e.g. PaintBrush, Visio, Inspiration. But computer program, which could be really useful for a knowledge engineer on the described stages of structuring of the subject domain, should necessarily follow the phenomenological nature of the knowledge elicitation and described above algorithms. This program must not frustrate the knowledge engineer with any "game rules" which were not evident for him/her. Ideally, it should adjust itself for particular cognitive features of the knowledge engineer. Moreover, each of the stages of analysis described above may be represented visually in its proper terms, as is already approached in some commercial expert system shells.

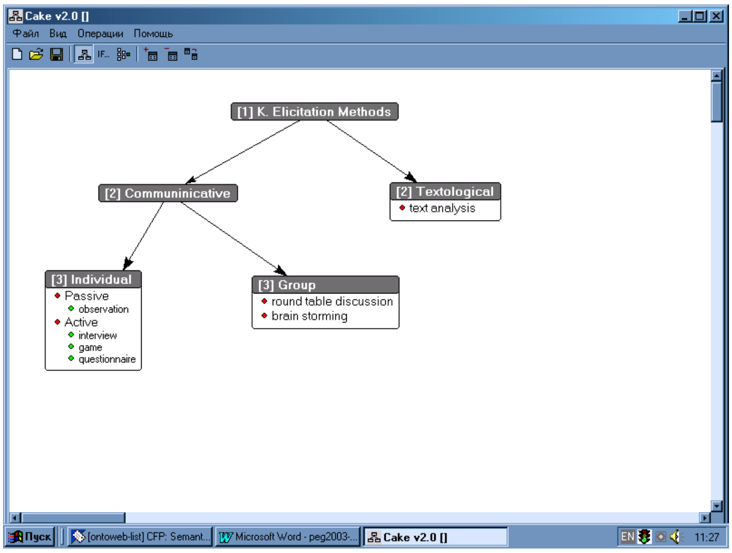

Figure 4. Screenshot with a part of ontology "Knowledge elicitation techniques".

A special visual tool was developed and named CAKE-2 (Computer Aided Knowledge Engineering by leading programmer Tim Geleverya, previous release by Alex Voinov). CAKE illustrates the idea of knowledge mappability, that find another application in the data mining and structuring for heterogeneous data base design. Its first prototype is described briefly in [Gavrilova, Voinov, 1996]. CAKE-2 proposes a kind of a visual knowledge representation language, which analogues may be found in a wide range of visual software construction tools – from large CASE's to Visual Basic. In particular, it supports the principle of a bi-directional mutually unambiguous correspondence between the two-dimensional visual object description syntax with the traditional one-dimensional one.

CAKE-2 is based on classical structured analysis methodology [Yourdon, 1989] enriched by new results that gives a teacher the opportunity to use special graphical interface to create ontology, to save it and to compile into the knowledge base (if needed).

Fig.4 presents CAKE's screenshot with fragment of the ontology of knowledge elicitation methods.

5. Discussion

Our approach puts stress on the knowledge structuring for better understanding of main course ideas in e-learning. The use of visual paradigm to represent and support the teaching process not only helps a professional tutor to concentrate on the problem rather than on details, but also enables pupils and students to process and understand great volume of information.

A better apprehension of teaching information might be achieved by imposing a knowledge structure on it. This may improve later usage of this information, comparing, generalisation, and so on. Therefore, a visual knowledge structure editor plays here a role of a two-dimensional, pictorial conspectus of the regarded piece of information.

Acknowledgements

The authors are very grateful to Tim Geleverya who assisted in the described research.

References

Adeli H. (1994) *Knowledge Engineering.* McGraw-Hill. New-York.

Boose, J.H.(1990) Knowledge Acquisition Tools, Methods and Mediating Representations. In: Motoda, H., Mizogochi, R., Boose, J., Gaines, B. (Eds.): *Knowledge Acquisition for Knowledge-Based Systems,* IOS Press, Ohinsha Ltd., Tokyo,123-168.

Conlon T. (1997) Towards Diversity: Advancing Knowledge-based Modelling with Knowledge Acquisition. In *Proceedings of 8th International PEG Conference.* Sozopol, Bulgaria.,379-386.

Eisenstadt M., Domingue J., Rajan T., Motta E. (1990) Visual Knowledge Engineering. *In IEEE Transactions on Software Engineering*, Vol.16, No.10, 1164-1177.

Fensel D. (2001) *Ontologies: A Silver Bullet foe Knowledge Management and Electronic Commerce*. Springer.

Gavrilova T.A., Voinov A. (1996) Visualized Conceptual Structuring for Heterogeneous Knowledge Acquisition // Proceedings of *Int. Conf. on Educational Multimedia and Hypermedia EDMEDIA'96, MIT, Boston, USA*, 258-264.

Gomez-Perez A. Tutorial on Ontological Engineering. *Int. Conference on artificial Intelligence IJCAI'99*.

Gruber, T. (1993) A translation Approach to portable ontology specifications. *Knowledge Acquisition*. Vol. 5, 199- 220.

Guarino, N., Giaretta, P. (1998) Ontologies and Knowledge Bases: Towards a Terminological Clarification. In *Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing*. IOS Press, 25- 32.

Jonassen, D.H. (1998). Designing constructivist learning environments. In C.M. Reigeluth (Ed.), *Instructional design models and strategies*, 2nd ed. Mahwah, NJ: Lawrence Eribaum.

Kremer R. (1998) Visual Languages for Knowledge Representation. In Proceedings of KAW'98 *Eleventh Workshop on Knowledge Acquisition, Modeling and Management*, Banff, Alberta, Canada, 124-129.

Neches et al, (1991) Enabling Technology for Knowledge Sharing. *AI Magazin,*. Winter, 36- 56.

Sowa, J. F. (1984) *Conceptual Structures: Information Processing in Mind and Machine*. Reading, Massachusetts, Addison-Wesley.

Scott A., Clayton J.E., Gibson E.L. (1994) *A practical Guide to Knowledge Acquisition*, Addison-Wesley Publ.

Swartout B., Patil R., Knight K., Russ T. (1997) Toward Distributed Use of Large- Scale Ontologies Ontological Engineering. *AAAI- 97 Spring Symposium Series,* 138- 148.

Tu S., Eriksson H., Gennari J., Shahar Y., Musen M. (1995). Ontology-Based Configuration of Problem-Solving Methods and Generation of Knowledge-Acquisition Tools: Application of PROTEGE-II to Protocol-Based Decision Support. *Artificial Intelligence in Medicine,* N7, 257-289.

Wielinga B., Schreiber G., Breuker J. (1992) A Modelling Approach to Knowledge Engineering. *Knowledge Acquisition*, 4 (1). Special Issue, 23-39.

Yourdon, E. (1989) Modern Structured Analysis, Prentice-Hall Bristol, Computer Science Department.

Authors' Information

Tatiana Gavrilova, Michael Kurochkin, Victor Veremiev –

Saint-Petersburg State Polytechnical University; Intelligent Computer Technologies Dpt.

Politechnicheskaya 29/9, St. Petersburg - 195251, Russia, Tel: + 07 812 371 82 72, Fax: + 07 812 373 06 89; gavr@limtu.spb.su