A Framework for Conceptually Modelling the Domain Knowledge of an Instructional System

Emilia Pecheanu, Luminita Dumitriu, Diana Stefanescu, and Cristina Segal

University 'Dunarea de Jos' of Galati, Department of Computer Science and Engineering, 800008 Galati, Romania
{Pecheanu.Emilia, Luminita.Dumitriu, Diana.Stefanescu, Cristina.Segal}@ugal.ro

Abstract. This paper presents a solution for conceptually modelling the teaching domain knowledge for computer-assisted instructional systems. The model consists of a theoretical framework and a knowledge representation approach. This model might be adopted in order to build a well-defined conceptual structure for the domain knowledge of a CAI system. The paper also presents an authoring system which implements the features of the modelling methods. This approach can offer a better solution to the problem of knowledge structuring and organising for computer-assisted instructional systems.

1 Knowledge Modelling in Computer Assisted Instruction

Increasing the assistance capability of instructional environments has lately become an important research subject in the area of computer–assisted instruction. A direction pursued by many systems designers is to build flexible, well defined models of the domain knowledge (the knowledge belonging to the training area), which is presented to users as instructional content.

In many instructional theories [3] [4] [6] it is considered that learning occurs as a result of accumulating and reorganising the knowledge within the human cognitive–mental structures. This vision of human knowledge acquisition, corroborated with representational properties of network-like architectural structures, leads to the idea that the domain knowledge might be organised in the form of "conceptual networks".

A conceptual network's node has to contain a description of some of the domain's concepts (main ideas), and the links between the network's nodes should be kept for the multiple relationships between these concepts.

The organisation model of the instructional content, which is tightly connected to the structure of the domain knowledge, has to provide the user with multiple views or presentation patterns of the concepts in the training domain.

Studies and assessments of computer-assisted instructional systems (CAI systems) have shown that if they contain a structural model, they facilitate learning. A well-structured architecture of the instructional content can also improve the efficiency of any guidance method the instructional system might use.

V.N. Alexandrov et al. (Eds.): ICCS 2006, Part II, LNCS 3992, pp. 199 – 206, 2006.

2 A Theoretical Framework for Knowledge Modelling in Computer Assisted Instruction

2.1 Main Design Principles

The modelling methods described herein might be adopted in order to build and to represent the knowledge space of an instructional system. The resulting models integrate the domain knowledge pertaining to a CAI system in a single structure. The modelling methods have been developed and fully described in [8]. The modelling methods have been mainly derived from Formal Concepts Analysis, [2] and Logical Concept Analysis, [1].

The theoretical framework and the knowledge modelling methods presented in this paper comply with the following principles, taken from [5]:

1. Domain knowledge modelling is based on the didactic experience of educators, who know how a discipline should be taught within institutional frameworks
2. Knowledge modelling is realised in preparation for the act of instruction
3. Knowledge modelling is a way of organising the resources that are necessary in the instructional process, in this approach.

Another design principle should be highlighted. Most CAI systems belong to one of the following categories, depending on the main features they possess [5]:

1. Pedagogy-oriented systems, focused on implementing well-defined (explicit) teaching and organising/sequencing methods for the instructional content.
2. Performance-oriented instructional systems, focused on assessing/diagnosing the knowledge acquired by the learners.

The modelling framework described herein is aimed at developing conceptual structures for pedagogy-oriented instructional systems.

2.2 Key Elements in Modelling the Domain Knowledge

The key elements in modelling the domain knowledge of a CAI system are the following: the Conceptual Unit, the Conceptual Structure, and the Conceptual Transition Path. These three elements are defined below.

Definition 1: A **Conceptual Unit C_i** is a group of related notions (concepts, basic ideas) belonging to the domain knowledge of an instructional system. Conceptual Units are obtained by applying a mathematical relation (or a set of relations) over a set of notions belonging to that domain knowledge.

Definition 2: A **Conceptual Structure** $\mathcal{S}$ is a set (N, R_N, C_S, L_S), where N is a set of notions belonging to the domain knowledge of a course or teaching discipline, R_N is an order relation over N, C_S is the set of Conceptual Units obtained by applying the relation R_N over the set N, and L_S is the set of traversing paths of the structure $\mathcal{S}$.

Definition 3: A **Conceptual Transition Path** $\mathcal{T}_s \in L_S$ is a set of conceptual units (C_{1s}, C_{2s}, ..., C_{Ns}) connected one after the other within a Conceptual Structure $\mathcal{S}$. A conceptual transition path $\mathcal{T}_s$ has as origin a the Conceptual Unit C_{1s}, considered initial, and as destination a Conceptual Unit C_{Ns}, which encapsulates a set of notions comprising a study goal within a teaching course.

A Conceptual Structure is a model meant to represent the domain knowledge of a teaching course. A Conceptual Structure should map the cognitive structure of the domain knowledge, and should also reflect the pedagogical vision of the teacher-author of that course. The model has to allow for flexibility, i.e. to provide as many transition paths as possible in order to learn the domain's main concepts.

A Conceptual Transition Structure should offer various solutions to traverse a group of notions in the interactive course. This requirement comes from the fact that the users of an instructional system have various learning styles, as well as various behavioural attitudes in learning. As such, an educational software-system should offer its users a flexible support for the learning act.

In computer-assisted instruction, a teaching-domain's notion or group of notions is transmitted to users through a presentational software-object, which can be: an explanatory text, an image, an animated image or a combination thereof. The presentational object can be as well a full-fledged application or software tool, intended to mediate learning through more complex methods of instructing: tutoring, exploring, practising, and so on. One can observe that a presentational object belonging to an instructional system could constitute the medium to learn/assimilate several related domains' notions.

The present approach assumes that for each main notion or group of notions belonging to the teaching-domain, there is a corresponding presentational object within the instructional software system. Presentational objects are supposed to possess a consistent internal description, permitting storage and retrieval.

The previous introductory ideas have led to the following developments, setting up several significant elements in computer-assisted instructional activity.

2.3 Notions and Relationships Within the Instructional Content

The notions (concepts, basic ideas) existing in the knowledge space of a domain are interconnected through multiple relationships. Within the frame of an instructional activity, a sense of knowledge in a domain might be acquired only if one considers the domain's notions and relationships as a whole. The most common relationships between notions within a teaching content can be considered as following:

1. **the relationship of precedence between notions**, i.e. the order or the sequence of these notions in the teaching process;
2. **the relationship of contribution of a notion in teaching another notion,** this relationship, which can also be expressed using phrases such as "significant contribution" or "reduced contribution", can then be quantified within a well stated range of values.

Applying any of the above relationships over the notions within an instructional content is always generating a specific structure. This structure might be a simple sequence of notions, a hierarchy or - most commonly - a network of interconnected notions. In this approach, the process of learning/assimilating a course's notions can be considered as a way of traversing the network of notions. This way of traversing should be carried out by respecting a specific order: it should begin with the "basic" notions and end up with the notion (or group of notions) considered final, or the instructional target.

2.4 The Modelling Approaches

A lattice-like model has been developed in [8] in order to be the Conceptual Transition Structure previously defined. This model is named **COUL-M** (COnceptual Units' Lattice Model). It might be adopted in order to represent the relationships between notions within the space of knowledge of a teaching domain in a comprehendsible way.

The COUL-M model has been built upon the mathematical formalisation of the precedence and contribution relationships. The mathematical equivalents of precedence and contribution relationships are the Precedence Relation and the Contribution Relation. These relations have several properties permitting to transform them by applying sequences of aggregation and decomposition operations into one-to-one incidence relations.

The **formal contexts** and **the formal concepts' complete lattice** can further be built for these relations [2]. Then, the pairs of sets (intent, extent) which compose **formal concepts** can be extracted from these formal contexts and **the formal concepts' complete lattice** can further be built [2]. The formal concepts are mathematical, abstract representations for sets of related notions within the teaching material of the course. The formal concepts' intent part (or the intent sets of the formal concepts) will stand as the "Conceptual Units" related to the teaching material. Several lattices of Conceptual Units, standing as Conceptual Structures for the domain knowledge can eventually be derived.

Thus, the COUL-M model for the domain's knowledge of a CAI system is defined as a complete lattice C_S . The latter is generated by rewriting the relations between the course notions as relationships of incidence, where the formal concepts are an abstract representation for sets of related notions within the teaching content.

The subposition operation [2] has been applied in order to compose the formal contexts of the Precedence and Contribution Relations. The results of this operation were integrated into the model named **COUL-RM**, defined as a set (N, R_N , C_S, L_S), where:

1. N is a set of notions from teaching-domain of a course,
2. C_S is the complete lattice,
3. R_N is the relation of order within lattice C_S, established by set inclusion,
4. L_S is the set of elementary chains to traverse the lattice C_S , by respecting the order subconcept –concept.

The COUL-M model has the following properties:

1. it is a complete lattice - a well defined mathematical structure - able to represent the interactive course's notions, as specified by the teacher-author of the course
2. it is a set of Conceptual Units
3. each Conceptual Unit in the COUL-M model constitutes a formally expressed description of several course's notions
4. within the COUL-M model the "new" notions to assimilate within each instructional sequence depend on what has been learned in the previous sequence (they depend on the selected instructional path) and cannot be listed beforehand.

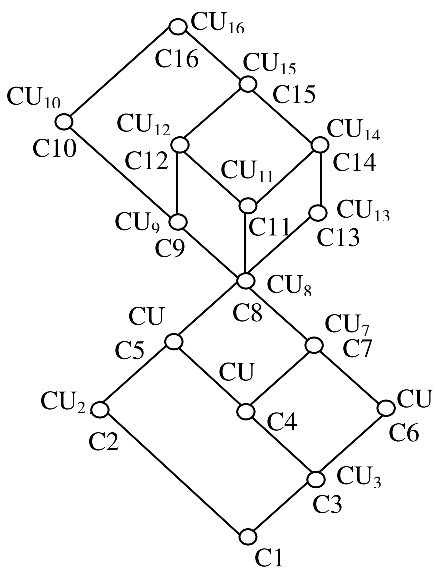
The COUL-RM model attempts to capture the semantic links between the notions of an instruction subject. It also attempts to corroborate them with the way in which the instruction can be carried out, taking into account the resources that are available at a given time within the interactive course.

2.5 The Knowledge Compiler

Let's consider the domain knowledge of a system meant for assisted-instruction in an IT course. This course might be "Operating Systems" for example. Furthermore, let's consider a set of basic notions belonging to a specific chapter in this course, i.e. "Mutual Exclusion". Starting from the pedagogical relationships, which are connecting this chapter's notions, the COUL-M model can be built.

In the teacher's pedagogical approach, the most important concepts of the "Mutual Exclusion" chapter are considered to be the following ones: mex, cr, cs, sem, sap, wap, pca, lra, where:

A = mex = mutual exclusion
C = cs = critical section
E = sap = strict-alternation of processes
F = pca = producer-consumer algorithms
B = cr = critical resource
D = sem = semaphore
F = wap = wait-activation of processes
G = lra = lecturer-redactor algorithms



Conceptual Units

CU_{16} = ABCDEFGH
CU_{15} = ABCDEFG
CU_{14} = ABCDEF
CU_{13} = ABCDE
CU_{12} = ABCDFG
CU_{11} = ABCDF
CU_{10} = ABCDGH
CU_{9} = ABCDG
CU_{8} = ABCD
CU_{7} = ABC
CU_{6} = AB
CU_{5} = BCD
CU_{4} = BC
CU_{3} = B
CU_{2} = D
CU_{1} = Φ

Fig. 1. The COUL-M model for the teaching-knowledge within "Mutual Exclusion" chapter

It is assumed that the teacher who has authored the interactive course has specified the pedagogical precedence relationships between these notions (depending on their meaning and on this teacher's pedagogical approach) as an ordered list:

cr, mex, cs, sem, sap, wap, pca, lra

The contribution relationship between notions has been specified by the following expressions:

cr → mex and wap → (pca, lra) and sap → wap
(cr, mex, cs, sem) → (sap, wap, pca, lra) and (cr, sem) → cs

The resulting COUL-M lattice-like structure for these notions and relationships - as described by the teacher - is depicted in Figure 1.

The COUL-M model has been implemented by means of a software tool - a knowledge compiler - named COUL-COMP (COnceptual Units' Lattice – knowledge COMPiler). The compiler has been developed so as to test the validity of the theoretical model and of the modelling methods carried out by the authors.

Based-upon the relationships specifications the compiler will produce the Conceptual Transition Path in the following form:

1. a list of the various learning paths the user can adopt in order to assimilate the notions of the interactive course
2. the list of the corresponding learning stages for each learning path.

In this approach, the learning paths calculated by the compiler are ordered sets of learning stages. The learning stages are standing for collections of interconnected notions that succeed in a well-defined order.

Various tests have been deployed and thus the authors have validated the modeling methods.

3 The Authoring System

The COUL-M model has been implemented within the frame of a software tool able to realise the teaching-content's structure design for an interactive course. The software-system has been named InStructGen, an acronym for Instructional-content Structure Generation and Deployment. InstructGen is destined to play the role of an authoring system and might be used in order to build and deploy a computer-assisted instructional environment.

Through a user-friendly interface (Figure 2) the InStructGen system helps the user (the teacher-author of the course) to specify the names of the course's notions and the location of software-objects corresponding to each notion. The relationships of contribution between these notions, i.e. the fact that some notions "contribute" to teach other notions must also be specified.

After the course notions and relationships specifications the conceptual structure can be generated. This structure is built according to the design principles previously described in this paper and can be presented to the teacher author in the shape of a lattice-like diagram (Figure 3).

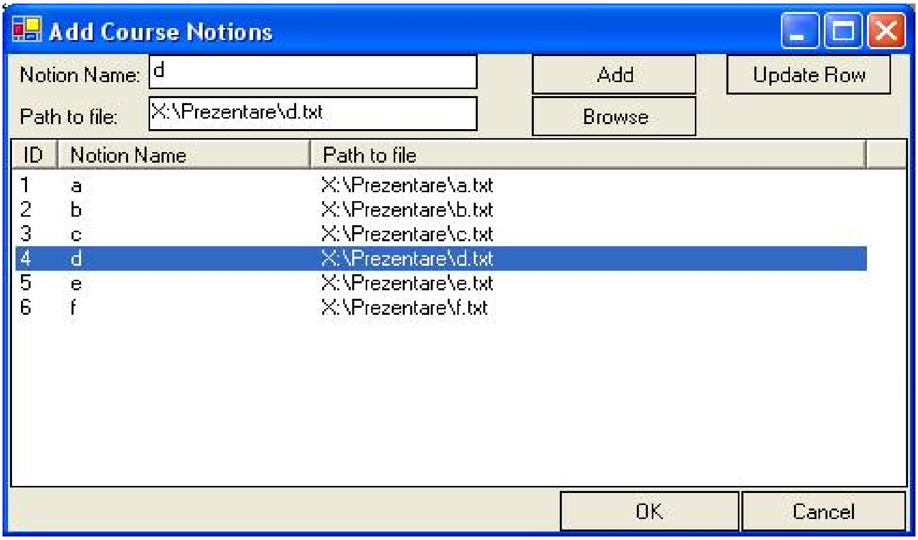


Fig. 2. An InsStructGen user interface screen-shot: specifications of software objects corresponding to the course's notions

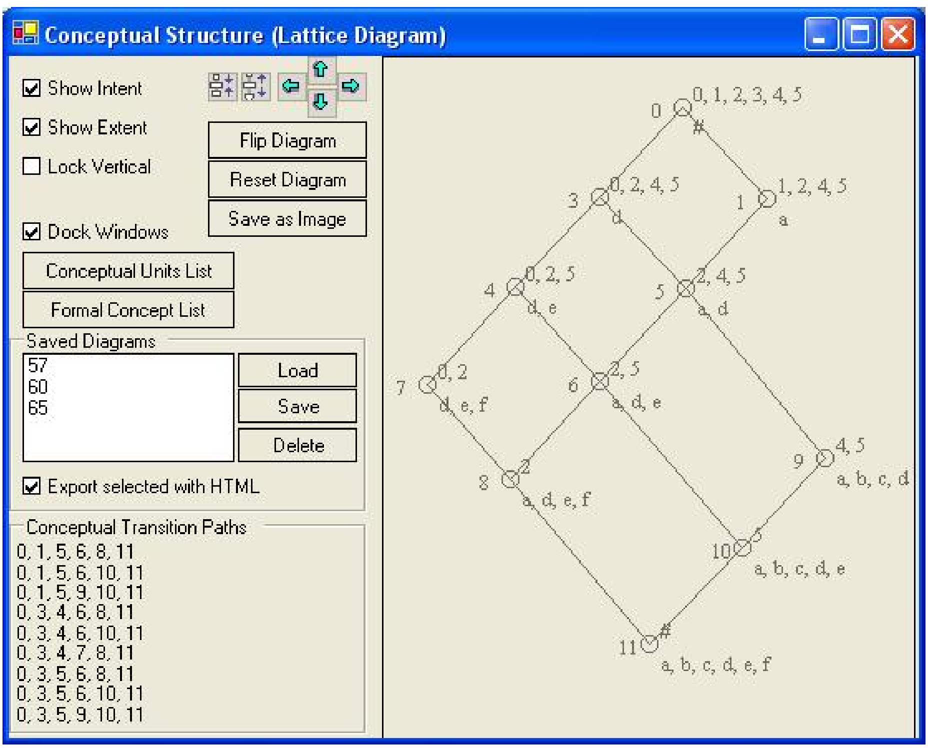


Fig. 3. A lattice-like model for the teaching content as generated by the InsStructGen system

Furthermore, the InStructGen software tool is able to generate a web-site including in its linked web pages all the software presentational-objects specified by the user.

This web-site's map is actually a network that strictly follows the lattice-like model previously generated by the InStructGen system.

4 Conclusions

The teaching-content design approach previously presented is based upon the lattice-like mathematical model COUL-M. This model is considered to be able to represent the relationships between notions within a training domain in a coherent, easily understandable way. The model has been implemented in a software tool – InStruct-Gen- standing as an authoring system that can generate an instructional environment having a well-defined conceptual structure.

The InStructGen authoring tool can be integrated in any environment including a Pool of Pedagogical resources or other collections of pieces of instructional content.

Further developments are planned for this project in the future: a client server architecture for the instructional environment generated by InStructGen and more extensive live-testing sessions carried out by users enrolled in instructional activities.

References

1. Ferré, S.: Systèmes d'Information Logiques. Rapport de recherche. IRISA, France, (1999)
2. Ganter, B., Wille, R.: Formal Concept Analysis – Mathematical Foundations. Springer-Verlag, Berlin Heidelberg New York (1999)
3. Larmat, J.: Intelligence's Genetics, Scientific Printing House, Bucarest, Romania (1997)
4. Messing, J.: The Use of Teaching Strategy Control Features in Computer Assisted Learning Courseware. MEd Dissertation. Charles Sturt University, Australia (1990)
5. Murray, T.: Authoring Intelligent Tutoring Systems-An Analysis of the State of the Art. International Journal of Artificial Intelligence in Education, Vol. 10, (1999) 98-129
6. Nicola, I.: Pedagogy. Didactical and Pedagogical Printing House, Bucarest, Romania (1992)
7. Pecheanu, E., Stefanescu D., Segal C.: Content Modeling in Intelligent Instructional Environments. In Proceedings of KES'2003 Conference, Oxford, (2003) 1229–12351
8. Pecheanu, E.: Conceptually Modeling the Domain Knowledge of an Computer Assisted Instructional System. PhD Thesis. University of Galati, Romania (2004)