A MODEL FOR KNOWLEDGE DOMAIN, PEDAGOGICAL ACTIVITIES, AND TEACHING STRATEGIES REPRESENTATION IN INTELLIGENT TUTORING SYSTEMS APPLIED TO ENGINEERING EDUCATION

Aida A. Fadel
University of Brasilia - Department of Mechanical Engineering – Asa Norte – CEP 70910-900.
aida@unb.br

Emiliano S. de Faria
University of Brasilia - Department of Computational Science – Asa Norte – CEP 70910-900.
esf@brturbo.com

Dianne M. Viana
University of Brasilia - Department of Mechanical Engineering – Asa Norte – CEP 70910-900.
diannemv@unb.br

Abstract. *This paper presents a formal model for the specification of knowledge domain structure and teaching strategies.This model allows automatic sequencing of engineering pedagogical activities in intelligent tutoring systems in computer networks (Internet or Intranet). The proposed model is expansible and able to deal with different pedagogical needs, aiming at application to complex themes and environments, such as those found in university courses. The present proposal is based on the expert overlay approach for modelling the student's model representation. In this approach the knowledge domain is represented by a hierarchical structure which will be visited through different paths for each student, depending on his or her own development and performance at the pedagogical activities. The choice of the paths is done after checking the teaching strategies on the knowledge domain, as defined by an engineering expert teacher . The proposal also incorporates concepts and practices of interoperability, interchangeability and reusability sponsored by international standardisation consortiums and entities.*

Keywords*: Intelligent Tutor for Engineering, ITS*

1. Introduction

Intelligent tutoring systems (ITS) are educational softwares that use some artificial intelligence techniques to aid the teaching-learning process. They are capable to simulate decisions, playing the role of the specialist teacher. More precisely, an intelligent tutoring system is a specialist system applied to an educational area which is enlarged with the Student Model and Teaching Strategies (Burns and Capps, 1998), notions that will be better described in the body of this work.

The ITS can be utilized on the WEB, in distance learning courses, in regular courses and in others learning environments as the main tool for support the teaching-learning process. Figure 1 shows the chronological evolution of modern tutoring systems.

A sprouting of ITS appeared in the 50's as the result of the evolution of the Computer Aided Instruction (CAI) systems, which were called Linear Programs. These programs considered knowledge in a linear form, so that no factor could change the teaching rule established by the programmer.

In the 60's, the programmers started to consider that students' responses could be used to control the study focus. So, the Branched Programs aroused because they were more suitable since they enabled a feedback, although they had a fixed number of instruction possibilities in the same way as the Linear Programs. But the focus was still the teacher: The student must comprehend the contents proposed by the teacher and answer the questions previously formulated to measure the level of understanding.

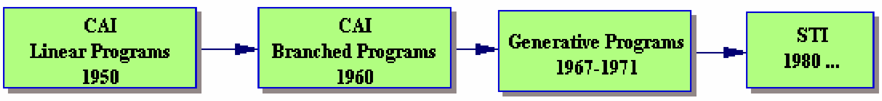

Figure 1. Historical evolution of ITS.

In the early 70's the Generative Programs appeared. They were capable to create pedagogical situations or questions that needed to be answered and corrected in an automatic manner. The proposal was to measure how much the student had learned. For example, the use of a random generator of numbers could produce two numbers to be added by the student and then, the result evaluated by the computer would be compared with the result provided by the student in order to give the answer. This approach was restricted to a little number of knowledge domain areas and the solution was the result of a procedure different from the human mind process logic.

In the 80's the ITS appeared intending to correct the mistakes of the old Generative Programs. They could be considered as intelligent as the CAI systems, but structured in a different manner to work educational domains such as the association of AI techniques and cognitive psychology techniques to guide the teaching-learning process. Sleeman and Brown (1982) revised the state of art about CAI systems and created the term "Intelligent Tutoring Systems" to distinguish ITS systems from CAI systems. This term has an implicit meaning about how to learn, focusing the learning experience in concrete activities. These systems aimed at making easy the teaching-learning process so that it could be effective, right and pleasant.

The traditional structure of an ITS (Barr and Feigenbaum, 1982; Yazdani, 1987; Self, 1988; Viccari, 1990; Oliveira, 1994) is shown in Fig. 2.

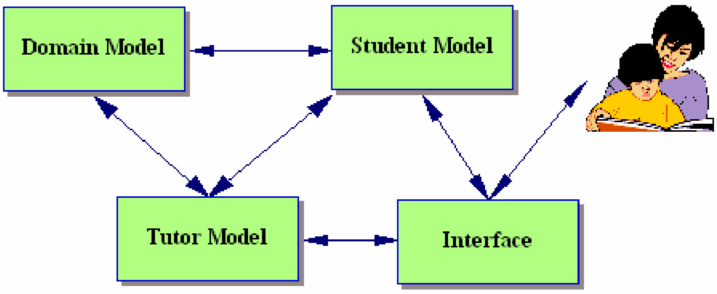

Figure 2. Traditional structure of an ITS (Viccari, 2003).

1.1. How can an ITS be used in engineering teaching?

In Brazil, the increasing necessity for universities vacancies is shown on the public policy to improve the capacity to supply these needs. This task is strongly limited by the real number of education professionals, once the growth rate of these professionals is not satisfactory. Moreover, the technologic evolution impels the knowledge expansion, increasing the amount of information and teacher qualification. How to consider this necessity without compromising the quality in engineering teaching?

The requirement for vacancies in Brazilian universities suggests the necessity of an instrument that uses the available technology and concepts of psychology and learning education, common to all knowledge areas.

The proposed model intends to modify the computer use, from a general hypermedia generator tool, into a resource generator tool for supporting the student privately during the learning process. The teacher will also have condition to review the material used in teaching activities or even the teaching process.

The tutoring system developed is capable to attend privately each student in agreement with his necessities. It offers different manners to treat engineering applications, as the valuation of presented results by the student in different unit systems. It could be also considered the possibility of using the same graphical representation introduced to the student in a question, by using image-editor software. For example, a student could easily draw the loads acting on a free body diagram, and then obtain the answer directly on the teaching environment.

Other characteristic of the system is the gain obtained by storing in specific libraries contents and pedagogical activities elaborated regarding different courses. The libraries can be catalogued according to information technology rules and the learning objects are formulated in agreement with standardization rules to enable their reusability.

2. Model Description

This model is based on the interaction between the Expert Teacher and the Student in the learning process using Artificial Intelligence techniques. It's composed by the following modules: *Knowledge Domain, Student Model, Diagnostic Module*, *Teaching Module.*

Knowledge Domain, which represents the Expert Teacher knowledge presented in a structured way and the knowledge transfer rules. In this module the sub modules of Specific Data Basis can be identified, which are the Knowledge Tree and the Teaching Strategies. *Student Model*, that denotes the actual level of the Student's knowledge during his interaction with the system, consisting of the sub modules: Student Background, Interaction History and Student Knowledge; the *Diagnostic Module* is the process responsible for applying the rules contained in the Teaching Strategies according to the actual level of the Student's knowledge, defined at the Student's History; *Teaching Module*, that refers to the interface between the Student and the system.

Authoring Tools are friendly interface programs to the user, in this case the Expert Teacher. These tools are used to design both, Knowledge Domain Module and *Learning Objects*, that are a set of Contents and Pedagogical Activities Descriptions presented according to International Specifications.

In Fig. 3, Carvalho, 2000, presents the overall scheme of the inter relationships between the main modules and other components of the ITS. More details on the components of the ITS will be presented later in this paper.

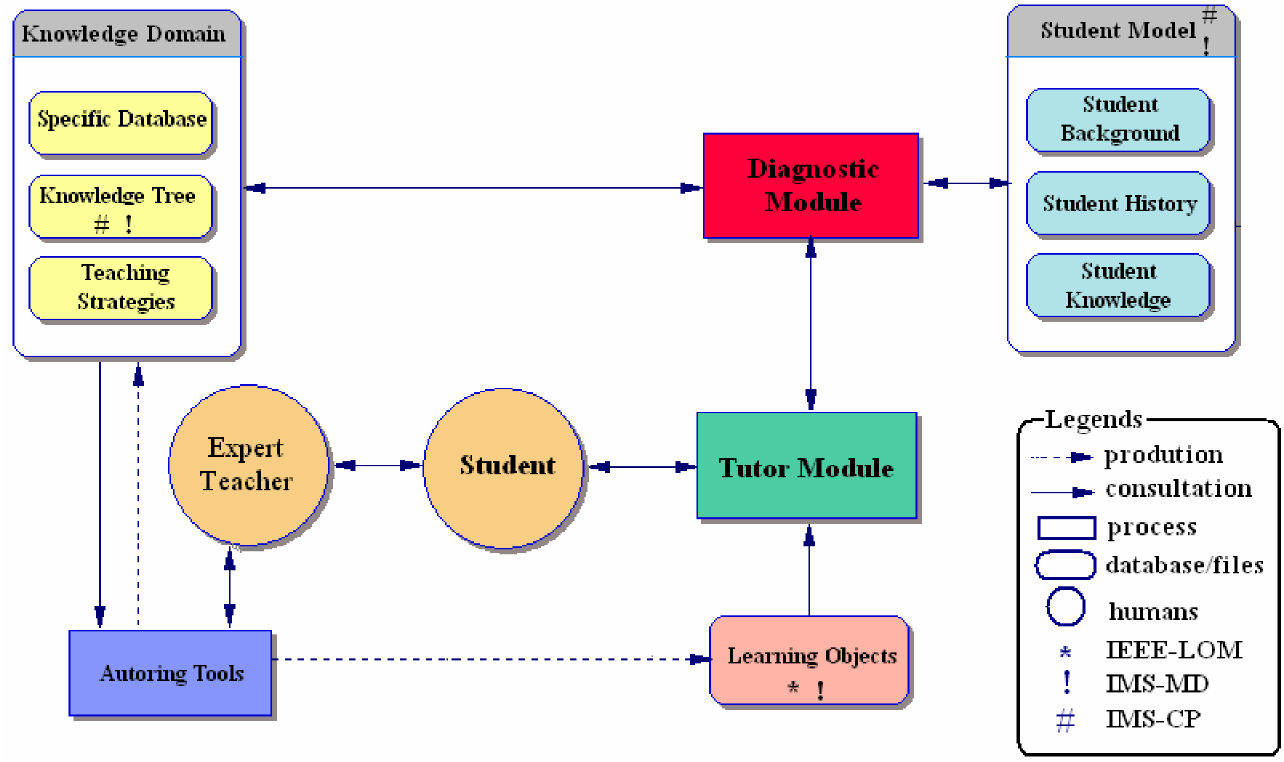

Figure 3. Intelligent Tutoring System architecture.

During the *design time*, the Expert Teacher will supply the basis of his expertise to the Knowledge Domain by using Authoring Tools. A Student Model, initially in the lowest level of knowledge of this expertise, will be updated in each system's iteration representing the Student's learning process. A register of the learning evolution process will, then, be recorded at the Student History and will be used as a parameter in the decision tree pre set by the Expert Teacher at the Teaching Strategies. Each decision will lead the Student into a new pedagogical scenario.

3. Implemented Standards

In order to guarantee the reusability of contents used in the proposed model, specifications established by IMS Consortium and LTSC-IEEE, main international standardization organisms, are adopted. Standard Specifications can be noted at contents organization as well as in the learning objects design, as shown in Fig. 3 legend. The contents are hierarchically organized, using XML technology, as determined by IMS Content Packaging (IMS, 2004), version 1.0, Final Specification; this will guarantee interchangeability, interoperability and reusability characteristics to the system.

Learning Objects have their classification defined by Learning Object Metadata (LOM) from LTSC-IEEE (IEEE, 2004) standard, being implemented according to IMS-Learning Resource Meta-Data, version 1.2.1, Final Specification.

Specific characteristics of Learning Objects' relation to the STI are defined in an external representation data model of Learning Objects as extensibility of LOM. This external model can define, for example, the processing directives of learning objects by the system as well as the class definitions properties of these learning objects and their applications instances.

4. How does the system capture and transmit the Teacher Expertise?

The mechanism of expertise capture is done through the Knowledge Tree definitions as well as the Teaching Strategies and the Learning Objects development. These resources are detailed in the following sections.

4.1. Knowledge Tree

The Knowledge Tree is the course syllabus presented in a hierarchal structure to organize the contents and activities in the respective curricula items. It is described in a common text file, using the Knowledge Tree Definition Language (KTDL), which is specifically established for this purpose (Carvalho, 2000), (Faria, 2004). In this file are described: 1) the curricula items, displayed in a sequence considered optimum by the Expert Teacher, and 2) the classes and instances of Learning Objects, which corresponds to the resources which will be used to present the curricula items, as classes, tests, examples...

4.2. Teaching Strategies

Through the Teaching Strategies Definition Language (TSDL) (Major, 1995), (Carvalho, 2000), (Faria, 2004) that uses natural language and also can be expressed in a common text file, a set of learning rules is declared. This file in addition to the Student and Expert Models are going to guide the teaching-leaning activities sequence. The rules are established in a condition-action structure. The TSDL uses the same terms and learning objects previously declared at Knowledge Tree.

The Expert Teacher anticipates an action to be taken by a particular student as the result of his performance in the teaching-learning process, in the sense of optimizing the learning process individually.

The set of rules will then offer several pedagogical scenarios to be chosen in each program interaction, based in the values that each Student shows in his grades' history. A student that didn't have a satisfactory valuation in a determined subject will be presented to the same subject in a different way. If the missing of some background knowledge is detected, the student can be guided to this content in order to correct the failure and return to the point where the original process was interrupted. If the grades rate of several students stays under a desired level, it can be an indication that the contents could be improved or the approach could be reformulated. This gives the Expert Teacher the possibility of continuous valuation and improvement of his material.

It can happen, for instance, that the Teaching Strategy defines a period of time in which the Student must reach a desired learning level. If the Student's grades are not satisfactory, he should contact the Expert Teacher or some assistant to get a personal help. Once his problems were solved, the Student can return to the ITS and continue the learning process. In this last case it can be observed that this model allows a strong human interaction at the process.

The definition of Teaching Strategies offers a level of flexibility that enables the Expert Teacher to adopt his preferred pedagogical strategy.

4.3. Learning Objects

The pedagogical activities are defined according to the proposed data model and labeled following the international standard specifications, being then referred as learning objects. The learning objects are related to both, the activities chosen to transmit the knowledge as classes, seminars, projects, and so on, but also to the contents and its approach. This means that the same subject can be exposed as movies, classes or using several hypermedia resources.

Learning objects represents a relevant component of ITS and must be planed, carefully chosen and previously evaluated before being incorporated to the system and applied, in order to guarantee a good performance of the learning tool.

5. Learning Diagnostics

The learning diagnostics are obtained by comparing the acceptable levels of learning, defined by the Teaching Strategies and the learning level showed by the Student, registered at the Student History, Fig. 4. For a student that just started the process, his knowledge level is the lowest possible while the Expert Teacher's is the highest. The system goal is to approach the Student's level to the Expert´s. This strategy is named Expert´s Overlay (Anderson, 1988).

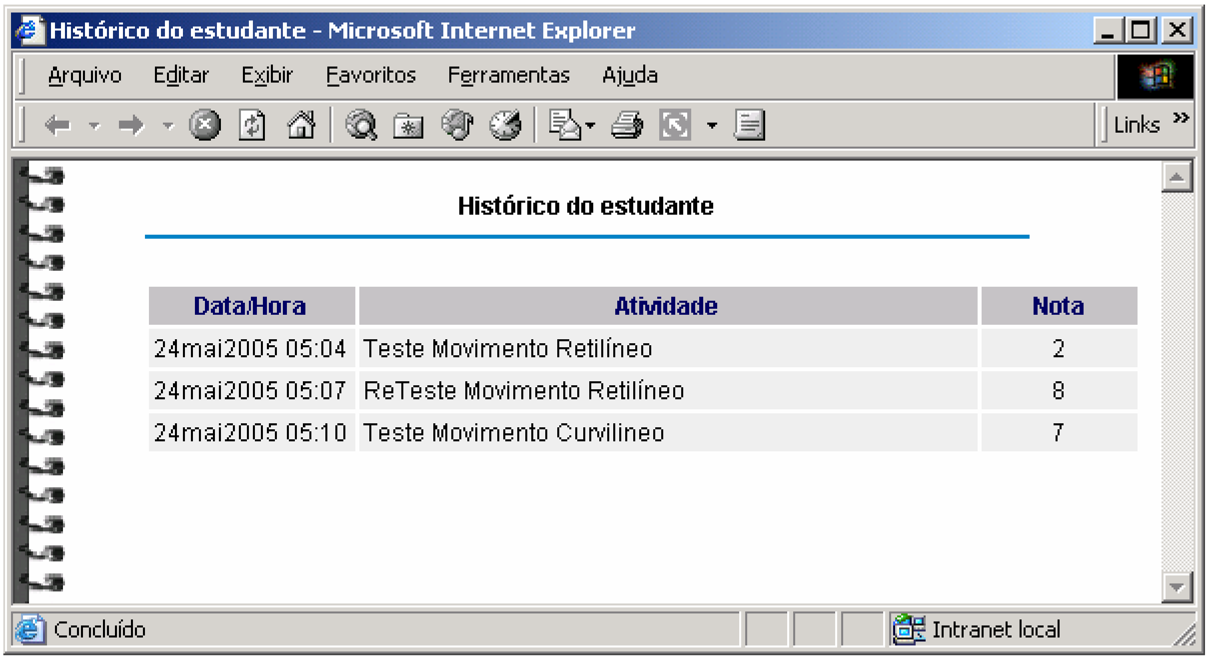

Figure 4. Student History.

6. Tutor Module

The Tutor Module represents the interface between the Student and the system itself. It's the gate that allows the student to reach the ITS and must present the necessary ergonomic characteristics to facilitate the teaching-learning process. For this purpose was used the Microsoft LRN tool kit V.3,0.

In this gate are presented the contents according to the level reached by the Student. A new set of activities are enabled by the application of learning rules. Among them, the selected activity is the one suggested by the system to be executed, as shown in Fig. 6.

7. Authoring Tools

For the Knowledge Tree and the Teaching Strategies' definition and implementation are necessary programming rules rather than international standards knowledge. This will turn the work into a complex task. To overcome these difficulties, authoring tools are developed, not just to define that two documents but also to help the production of Learning Objects.

7.1. Learning Objects Production

The authoring tools used to define the Knowledge Tree and the Teaching Strategies are specifically developed for this task, unlike Learning Objects Authoring Tools. The Learning Objects Authoring Tools can use several resources largely available and also specific software that can be developed to satisfy different needs. Among the most common tools are text editors, presentation editors, HTML page editors, and other programming environments in accordance to IMS international standard.

7.2. Ontology and Taxonomy

In order to guarantee interoperability characteristics, reusability of contents and activities common to more than one course, taxonomy and ontology concepts are applied in the Learning Objects creation.

The possibility of sharing knowledge was the main reason for ontology utilization in AI field in the last years. More than a standard vocabulary, ontology assures that the terms chosen are enough to specify, define concepts and correct relationships from the adopted terminology. It gives more information than taxonomy does, once ontology is able to define what the specific knowledge domain the files belong to is. For the computer science community the term ontology refers to a semantic definition that includes a vocabulary of terms, organized in taxonomy, their definitions and a set of formal axioms which are used to establish new relations and restrict their interpretations according to a desired sense (Noy and Hafner, 2000).

In other words, the needs of understanding among the agents are assured by using a common vocabulary among the interested parts, which will be consistent with respect to what is specified at ontology. This agreement is called “Ontological Commitment” and is crucial to enable different work groups in several areas to communicate and share results.

An example in engineering field would be to classify activities and contents related to build Free Body Diagrams. This topic must appear in different courses such as Statics, Dynamics, Mechanics of Materials, Fluid Mechanics, Machine Design … These contents and activities could be shared at the topic “Free Body Diagrams”, by using the ontology concepts.

8. An Engineering Model Application

An application of this system was developed for two courses of the Mechanical Engineering Department of University of Brasília: Mecânica 1 and Mecânica 2, which are related respectively to Statics and Dynamics.

To implement the ITS Generator (GeSTI) (Faria, 2004), that includes the development of Tutor Module, Diagnostic Module, KTDL and TSDL, the Visual Studio 6.0, SGDB Access 2000, XML SPY 6.0, and some graphic editors for image and icon edition were used. The program was compiled to run in an Internet Information Server under the operational system Windows 2000. Contents were implemented using resources as text editors and PDF document generators, as well as automatic questions generators, among others.

To insert the model application the Knowledge Tree was initially created, which is the structured syllabus from the course, Fig. 5, in KTDL, Fig. 6. The next step was to create a file of Teaching Strategies using TSDL, as shown in Fig. 7. Once the program is compiled the Tutor Module, shown in Fig. 8, is able to apply the learning rules, expressed as condition-action, couching the Student according to his idiosyncratic characteristics.

Mecânica 2 Course Syllabus

1.0 Introduction
- **1.1 Mechanic System Models**
- **1.2 Mechanics Domain**

2.0 Basic Concepts
- **2.1 Coordinates Systems**
- **2.2 Classes of Movements**

...

Figure 5. Partial View of the Mecânica 2 course Syllabus.

Activity	**Class**
Modality	distance activity
Activity Class	Boolean
Activity	**Tutoring**
Modality	presential
Valuation type	manual
Valuator	tutor
Activity Class	Boolean
Activity	**Multiple Choice Test**
Modality	distance activity
Number of Students	individual
Rate of Grades	from 0 to 10
Valuation type	automatic

1. Introduction
   introduction.htm
2. Particle Kinematics
   cinem_part.htm

2.1. Types of Movements
   tiposmov.htm
2.1.1. Rectilinear Movement
   mov_ret.htm
2.1.1.1 Read The Message!!!!!
   Message
   Mes1a.htm
   Disable
2.1.1.2. Rectilinear Movement Class
   Class
   1_movim_ret_pdf
2.1.1.3. Rectilinear Movement Test
   Multiple Choice Test
   GERATEST

Figure 6. Partial View of the Knowledge Tree in KTDL.

If
Multiple Choice Test's Grade (Rectilinear Movement) ***graterthan*** MM
Then
Disable Multiple Choice Test (Rectilinear Movement)
Enable (Curvilinear Movement)
Select Class (Curvilinear Movement)
Enable Multiple Choice Test (Curvilinear Movement)
Select Multiple Choice Test (Curvilinear Movement)
Else
Enable Message (Rectilinear Movement)
Select Message (Rectilinear Movement)
Disable Multiple Choice Test (Rectilinear Movement)
Enable Multiple Choice ReTest (Rectilinear Movement)
If
Multiple Choice ReTest Grade (Rectilinear Movement) ***smallerthan*** MM
Then
Disable Message (Rectilinear Movement)
Enable Tutoring (Curvilinear Movement)
Select Tutoring (Curvilinear Movement)
Disable Multiple Choice ReTest (Rectilinear Movement)

Figure 7. Partial View of Teaching Strategy files in TSDL.

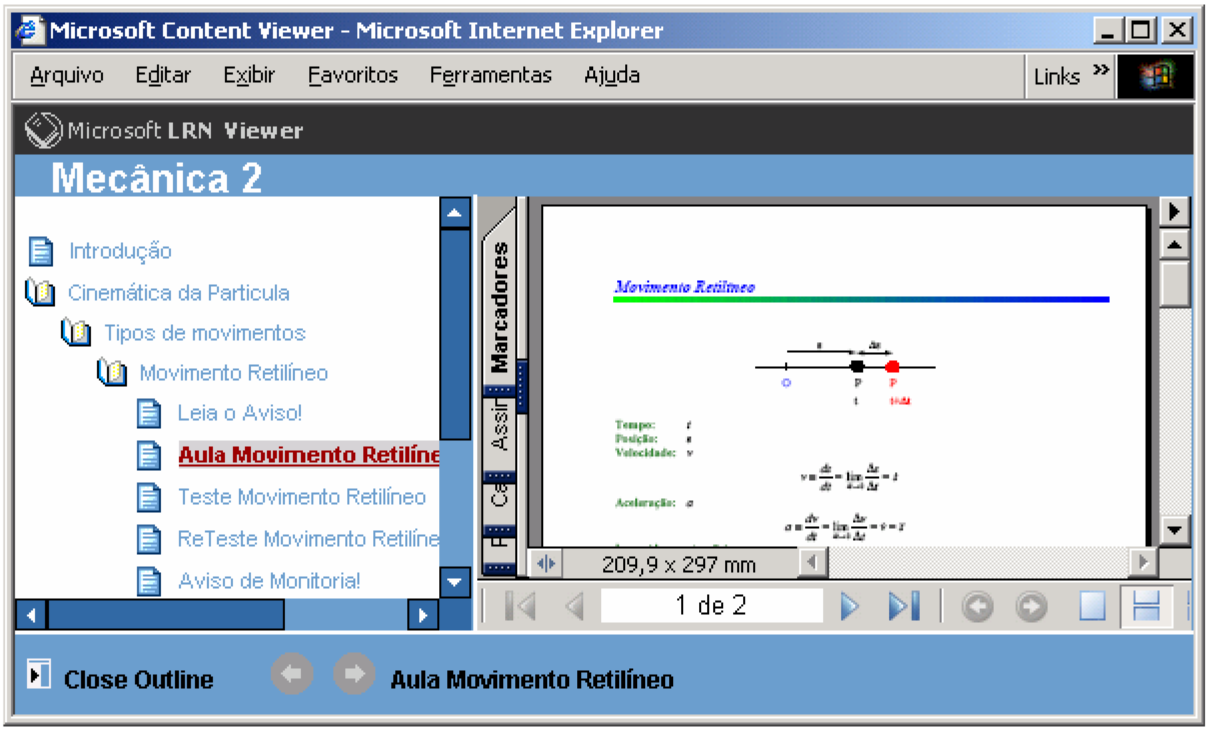

Figure 8. Tutor Module View.

9. Conclusions

The computational model is still being developed. Specific engineering authoring tools are being developed. Considering the computational point of view the learning tool combines the use of traditional compilers to international standards, allowing its sharing ability. The application of indexation concepts guarantee characteristics of reusability of material previously generated by any course.

The implemented model allows a major human interaction to the teaching learning process once it makes possible the straight contact between the Student and the Expert Teacher during the *run time* of the process.

The prototype is already working and tests will be performed with engineering courses to validate and improve the model.

10. References

Anderson, J.R., 1988, “*The Expert Module*”. In Polson , M.C.; Richardson , J.J., Editors, Foundations of Intelligent Tutoring Systems, chapter 2, pp. 21–52. Lawrence Erlbaum Associates Publishers, Hillsdale, New Jersey.

Barr, A., Feigenbaum, E.A., 1982, “*The Handbook of Artificial Intelligence*”. *Los Altos: California*, USA.

Burns, H.L.and Capps, C.G.,1988, “*Foundations of Intelligent tutoring systems: An Introduction*”. In Polson, M.C.; Richardson , J.J., editors, Foundations of Intelligent tutoring systems, chapter 1, pp 1–19. Lawrence Erlbaum Associates Publishers, Hillsdale, New Jersey

Carvalho, M., 2000, “*Generating intelligent tutoring systems for teaching reading: combining phonological awareness and thematic approaches*”.Tese de doutorado, University of Edinburgh, Edinburgh, U.K.

Faria, E.S., 2004,” *Um Modelo para Especificação de Domínio de Conhecimento e Estratégias de Ensino para Sequenciamento de Atividades Pedagógicas em Sistemas Tutoriais Inteligentes*”, Tese de Mestrado, Universidade de Brasília, Brasília.

IEEE, 2004 -Learning Technology Standards Committee. IEEE P1484.12 LOM Working Group. Available at http://www.ltsc.ieee.org/wgl2 , at june/2004.

IMS, 2004 - Instructional Management Systems,. Available at http://www.imsproject.org, at july 2004.

Major, N., 1995, “*Modelling Teaching Strategies*”, Journal of Artificial Intelligence in Education, 6(2/3), pp. 117-152.

Noy, N. and Hafner, C., 2000, “Ontological foundations for experimental science knowledge bases”. *Applied Artificial Intelligence*, 14(6):565–618.

Oliveira, F. M., 1994, “*Critérios de Equilibra*ção *para Sistemas Tutores Inteligentes*”, Tese de Doutorado, Porto Alegre: CPGCC/UFRGS, Brazil.

Sleeman, D.H. and Brown, J. S. ,1982, “Intelligent Tutoring Systems”, New York, Academic Press.

Self, J., 1988, “*Artificial Intelligence and Human Learning*”. London: Chapman Hall.

Viccari, R.,1990, “*Um Tutor Inteligente para a programação em Lógica -Idealização, Projeto e Desenvolvimento*”, Viccari, R.; Giraffa, M.M, 2003, “*Fundamentos de Sistemas Tutores Inteligentes*”. CPGG – Curso de Pós-Graduação em Computação- UFRGS: PPGCC – Programa de Pós Graduação da Faculdade de Informática - PUCRS.

Yazdani, M, 1987, “*Intelligent Tutoring Systems: an overview. Artificial Intelligence and Education*”. [S.I]: Ablex, Vol.1, pp. 183-201. Tese de Doutorado, Universidade de Coimbra.

11. Responsibility notice

The authors are the only responsible for the printed material included in this paper.