

## Reengineering of Educational Content: an Experience in the Computer Networks Domain

Joice B. Machado<sup>1</sup>, Marco A. G. Silva<sup>1</sup>, José C. Maldonado<sup>1</sup>, Ellen F. Barbosa<sup>1</sup>

<sup>1</sup>University of São Paulo (ICMC/USP) – Sao Carlos (SP), Brazil

{joicebm, magsilva, jcmaldon, francine}@icmc.usp.br

**Abstract.** *The growing demand for high-quality educational products, capable of better engaging learners and instructors in an active learning process, has pointed out the need for more efficient and productive learning development mechanisms. The IMA–CID approach address such concerns, defining an integrated approach for modeling educational content. By means of a set of models, IMA–CID helps the content developers to determine the relevant parts of the knowledge domain, providing a systematic way to structure the concepts and related information. In this paper we summarize the key characteristics of IMA–CID, and define guidelines for its application, illustrating them in the reengineering of an educational content in the Computer Networks domain.*

### 1. Introduction

Education has been through enormous changes in the last decades. The need for a global education, capable of crossing international, cultural and social borders in order to prepare the learners for the global market, is rapidly changing the concept of learning. Additionally, the fast evolution of information and communication technologies has multiplied the possibilities of learning.

The idea of *educational modules*, concise units of study delivered to students by using technological and computational resources, has emerged in this context. Among the activities related to the development of educational modules, content modeling plays an essential role [Rodríguez-Artacho and Maíllo 2004]. It aids in the identification of the relevant parts of the knowledge domain, providing a systematic way to structure the concepts, promoting reusability and adaptability [Barbosa and Maldonado 2006].

There are several related works dealing with the content modeling activity. For instance, EMLs (Educational Modeling Languages) [Koper et al. 2000] have been proposed to support the description of instruction mechanisms and resources used in the learning. Learning Design specification (LD) [IMS 2003] provides a notation to support the description of instruction in computational environments. MISA (*Méthode d'Ingénierie des Systèmes d'Apprentissage*) is a method for instructional engineering based on a problem solving approach [Paquette et al. 2005].

Also, there are approaches for designing educational hypermedia applications. DAPHNE (*Definição de Aplicações Hipermedia na Educação*) [Kawasaki and Fernandes 1996] is based on the Concept Mapping Theory [Novak 1990] and on the Information Mapping Technique [Horn 1989]. EHDM (Educational Hyperdocuments Design Method) [Pansanato and Nunes 1999] is based on the Concept Mapping and on the Michener's work [Michener 1978]. MAPHE

(*Metodologia de Apoio a Projetos de Hipertextos Educacionais*) [Pimentel 1997] incorporates to the Concept Mapping some relations of Object-Oriented Modeling.

The existing content modeling approaches addresses different modeling issues and levels of abstraction, which can be suitable for a given learning scenario but inadequate for others. In this sense, we proposed the *IMA-CID* approach (Integrated Modeling Approach – Conceptual, Instructional and Didactic), aiming at providing a well-defined set of mechanisms to create, reuse and evolve educational modules [Barbosa and Maldonado 2006]. *IMA-CID* has already been applied into the development of educational modules for several domains, and a common usage scenario is the adaptation and evolution of current courses. Thus, educational content, such as slide sets, are usually available and can be used to bootstrap the development of educational modules. Considerations about this reengineering activity, such as guidelines on how the *IMA-CID* models can be built, constitute a valuable contribution to the adoption and effective use of the approach.

In this work we describe the use of *IMA-CID* in the reengineering process of an educational module about Computer Networks. The module is part of a course on Network Security. Theoretical and practical aspects of *IMA-CID* are described, focusing on the guidelines for its application. We consider a scenario in which the educational module is built from existing educational resources (slide sets). In the end of the reengineering process, a new set of slides is created. The new module is compared against the original one, providing preliminary evidences on the benefits of adopting *IMA-CID*.

This paper is organized as follows. In Section 2, we present an overview of *IMA-CID*. In Section 3, we illustrate the application of *IMA-CID* in the reengineering of the educational module. Our conclusions and further work are presented in Section 4.

## 2. An approach for modeling educational content

The *IMA-CID* models are defined to specify different perspectives of educational resources, aiding the identification and definition of concepts, which are later refined and augmented with information and instructional elements, also allowing the consistent development regarding pedagogic and didactic theories and principles [Barbosa and Maldonado 2006]. Considering the elements required by an educational module, three main perspectives can be identified according to *IMA-CID*:

- *Conceptual*: defines the main concepts and relationships in the knowledge domain represented by the educational module. Often, those concepts and relations can be interpreted as learning objectives.
- *Instructional*: defines and associates information items (facts, principles, and concepts) and instructional elements (examples, exercises, and evaluations) to the elements previously defined in the conceptual perspective.
- *Didactic*: establishes the sequencing that elements, defined in the instructional perspective, are to be delivered and used by learners.

The conceptual perspective is represented by the conceptual modeling and its concept maps. The instructional perspective is built using, as input, the concept map, information items and instructional elements, which are defined in a statechart-based model (*HMBIS/Instructional*). This model is refined and augmented with sequencing information in the didactic model, which is represented by a *HMBIS/Didactic* model.

## 2.1. Conceptual model

The conceptual model provides a high-level description of the knowledge domain to be taught. It defines the main concepts and the relationships between them. Such relationships represents either structural relations (hierarchy and composition/aggregation) or domain-specific relations [Barbosa and Maldonado 2006]. The model is built by using the Concept Mapping Theory [Novak 1990]. Besides the suitability for representing the concepts and relationships, it is intuitive, based on educational principles and it does have a good acceptance among educational professionals.

Although the construction of a concept map relies on the educational setting, learning objectives and domain experts, some guidelines can be provided for the current scenario, which has a set of educational content as groundwork. Assuming that the first slides provide an overview of the entire course, the main topics can be identified from them. For each topic, conceptual models can be created by using the following guidelines:

1. Identify the main concept of the module. Concepts are often defined as nouns.
2. Insert the main topic in the top of the concept map.
3. Identify the concepts directly related to the main concept. Such relationships are often represented by verbs (that links the related concepts).
4. Identify structural relationships between the concepts.
  - (a) Identify taxonomy relationships (usually identifiable by expressions such as “is a type of”, “is classified in”).
  - (b) Identify composition relationships (usually identifiable by expressions such as “is composed by”, “is part of”, “has”).
5. Insert all the related topics in the same line, right below the main topic.
6. Draw the relation between the main concept and its related concepts.
  - Label taxonomy relationships with *type-of*.
  - Label composition relationships with *part-of*.
7. Identify domain-specific relationships between the main concept and the related concepts, and between the related concepts (relationships that are not taxonomy or composition related are domain specific).
8. Draw the relations between the concepts, and label them with the verbs identified in the relation.
9. Considering the topics identified in 1 as the main topics, repeat steps 3 to 9 until all relevant concepts have been represented.

## 2.2. Instructional model

Besides concepts, information items and instructional elements should be considered as part of the knowledge domain. The instructional model captures such elements, associating them to the concepts previously identified, and defining a structure that provides a more significant and motivating educational module. Information items are modeled after the Component Display Theory [Merrill 1983], which classifies the information as concepts, facts, procedures and principles. Regarding to instructional elements, they can be classified into three categories: explanatory (complementary theoretical information about an information item), exploratory (elements that foster practical activities regarding information items), and evaluative (diagnostic, formative and summative evaluations).

Information items and instructional elements are modeled using the *HMBBS/Instructional* model, which is an extension of the *HMBBS* (Hypertext Model Based in Statecharts) [Turine et al. 1997]. Briefly speaking, the model is similar to a statechart, disregarding transitions, events and history mechanisms. It represents the hierarchical decomposition and knowledge categories: the hierarchy is realized by relationships between the states (parent-child, sibling and nested states). The model can be created by using the following guidelines:

1. Create a new state for every concept in the conceptual model.
  - The main state corresponds to the main concept of the educational module.
  - For concepts (child) which are related to the same concept (parent) using a *type-of* relationship: (1) create a state for the parent state, (2) create, within the parent state, an orthogonal (OR) state for each child-concept.
  - For concepts which are related by a *part-of* relation, create a plain state, for the minor concept, within the state that represents the major concept.
  - For all the other concepts, either create a new (plain) state at the same level of the related concept (if it is in the same level at the concept map) or represent it using orthogonal (OR) states.
2. For each concept identified in the conceptual model, information items should be extracted (*e.g.*, from the original text) or created.
  - Facts are often described as sentences that defines a concept or that relates a concept to another.
  - Procedures are a sequence of steps that describes how to accomplish a specific activity or operation regarding to or using one or more concepts.
  - Principles are axioms or general statements that explain why something happens in a particular way.
3. Associate the information items to a state. Information items related to the same concept are generally represented as AND states.
4. Identify instructional elements (examples, exercises, tools).
5. Create a state for each instructional element. If an instructional element is too complex, child-states can be created to properly define it.
6. Associate each state that represents an instructional element to a state that represents a concept by means of explicit transitions.
7. For each information item and instructional element, specify the media to be used (text, code, image, animation, video, simulation, etc).

### 2.3. Didactic model

The didactic model is responsible for the establishment of prerequisites and didactic relationships between the states identified in the instructional model, defining sequences of presentation for a specific educational context, and pedagogical approach/order [Barbosa and Maldonado 2006]. The model is specified using *HMBBS/Didactic*, which is similar to *HMBBS/Instructional*, but now using all the *HMBBS* elements (transition, events, history mechanism) and adding a new mechanism: dynamically defined (*DD*) states. The semantic of the model is as following:

- *DD* states do not have an initial inner-state (the initial state is defined at runtime).
- Orthogonal states, children of a *DD* state, are fully connected.
- The exit of a *DD* state can activate any orthogonal state of the hierarchy of *DD* states.

*DD* states support the notion of open specifications, which allows the representation of all possible navigation paths between states. Such characteristic allows the dynamic definition of sequences of navigation (based upon restrictions as timing, learner's profile, personalization, etc.), in contrast to closed specifications, that assume a navigation model for a predefined user profile. Considering such notion of open or closed specifications, the learner's profile and pedagogical approach, the *HMBSDidactic* can be created by using the following guidelines:

- Define precedence relationships (represented by transitions) between sibling states.
  - Analyze the information items defined within the state and their dependence to information items in other states.
  - If there is no precedence between sibling states, replace the parent state with a *DD* state.
- Considering the innermost parent-state, define the initial state (but only if the state is not a *DD* state).

#### **2.4. Content generation**

From the didactic model, it is possible to generate the educational content. Different formats can be considered (PDF slides and/or HTML pages). Let's consider, for instance, a set of slides. The following guidelines apply:

1. Each state corresponds to one or more slides (depending on the amount of information conveyed).
2. Hierarchical decomposition should be represented in the navigation menu.
  - For a given state, every reachable state must be shown in the menu.
3. Implicit and explicit transitions among states should be represented by means of links in the navigation menu.
4. Besides the main slides, some specific slides regarding title page, references, and summary should be created.

### **3. An educational module about Computer Networks**

To illustrate the application of *IMA-CID* and the aforementioned guidelines, an educational module about Computer Networks has been reengineered. The module is part of a course on Network Security, developed by professors and researchers on such domain. In short, the development consists of the reengineering of the original course to a new format. By using the guidelines (described in Section 2), the original set of slides is deeply analysed and the related *IMA-CID* models are created. So, based on such models, a new set of slides is created in conformity with the approach.

The entire course on Network Security is organized into 20 lessons, each of them having approximately 50 slides. The target audience is composed of undergraduate and graduate students as well as network administrators and security professionals. It is important to highlight that, in the end of the reengineering process, each lesson corresponds to an educational module.

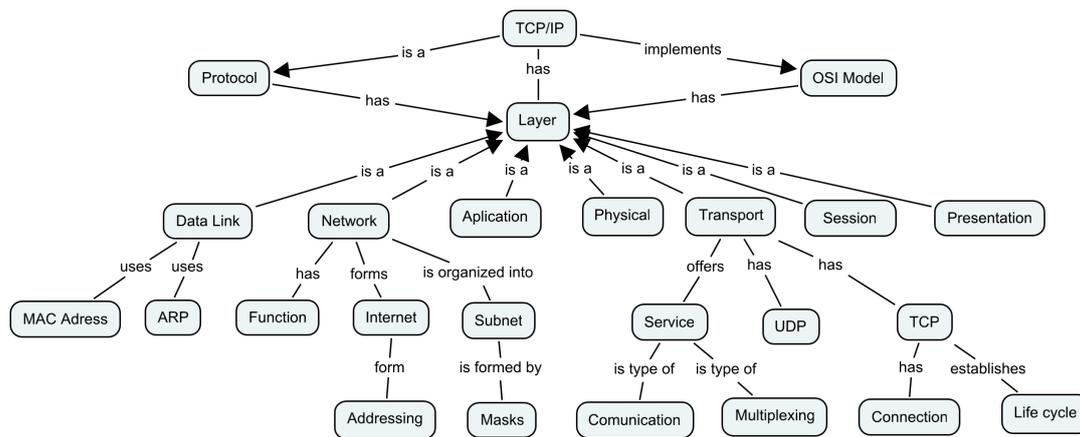


Figure 1. Conceptual model for TCP/IP Network.

### 3.1. Conceptual model

For the sake of illustration, we will consider a single topic of Computer Networks: the TCP/IP Network. As said before, the conceptual model is composed of the main concepts and their relationships. By applying the guidelines related to the conceptual modeling phase, the main concepts for TCP/IP networks were identified: *TCP/IP*, *Protocol*, *Layer* and *OSI Model*. They were organized in the conceptual model, as depicted in Figure 1.

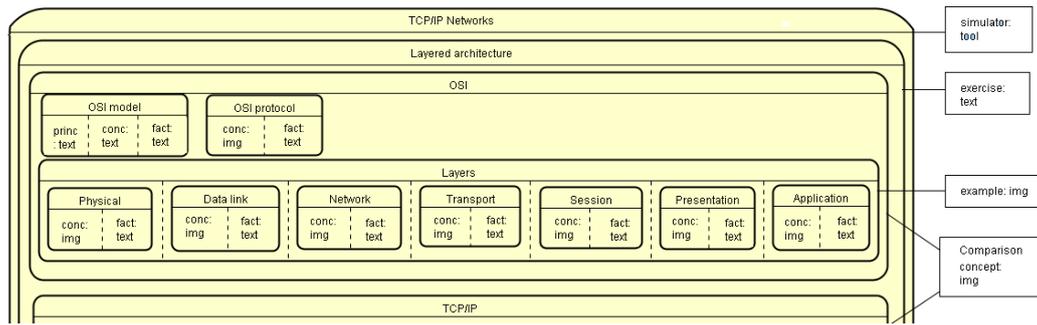
Notice that the concepts *Link*, *Network*, *Application*, *Physical*, *Transport*, *Session* and *Presentation* are in the same level and can be considered siblings, with a taxonomy/hierarchy relationship among them and the *Layer* concept (parent). Among *Layer* and *OSI Model* there is a composition relationship. The same type of relationship can be identified between *Service* and its parts: *Communication* and *Multiplexing*. Finally, as domain specific relationships we highlight the *implements* relation between *TCP/IP* and *OSI Model*, and the *uses* relation between *Link* and *ARP*.

Trying to “read the topic” by using the Figure 1 (*i.e.*, not considering any information in the original course), one can get a general view about TCP/IP Networks. For instance, we can infer that TCP/IP is a protocol that implements the OSI Model, organizing itself into layers.

Although the conceptual model can be used to provide an overview of a given knowledge domain, it must be pointed out the subjective characteristics of this phase. Thus, several different conceptual models can be defined, representing different conceptual views depending on the authors knowledge and experience. Therefore, the completeness of the concepts and relationships must be taken into account so that the next phases of modeling activity can be correctly done.

### 3.2. Instructional model

The instructional model is responsible for representing information items and instructional elements related to the concepts previously identified in the conceptual model. The diagram at Figure 2 represents a partial view of the instructional model addressing the *Layered architecture* of *TCP/IP Networks*, and the concepts directly related to it (*OSI* and *TCP/IP*). Some instructional elements (*Simulator*, *Exercise*, *Example* and *Comparison*) are also represented.



**Figure 2.**Part of the instructional model for TCP/IP Network.

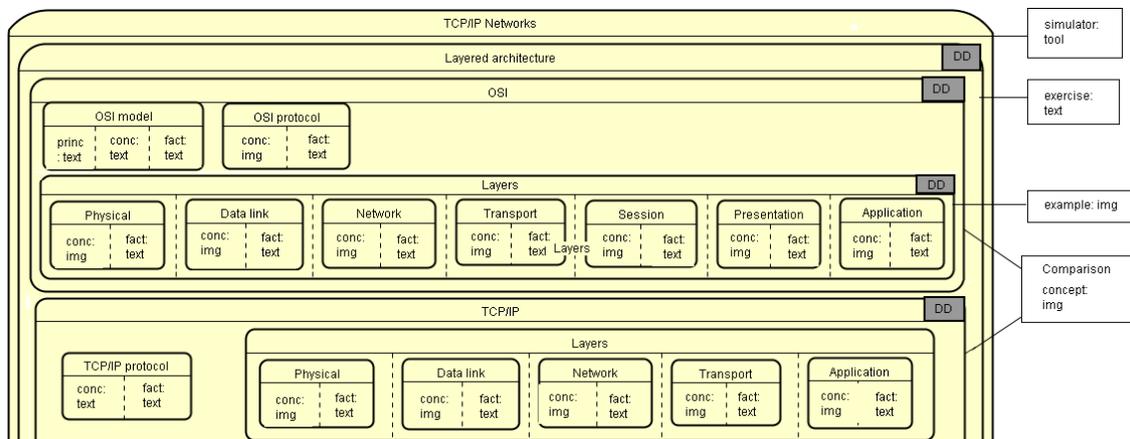
For instance, the concept *OSI* is composed of the concept *Layer*, which can be classified in seven other concepts: *Physical*, *Link*, *Network*, *Transport*, *Session*, *Presentation* and *Application*. For each concept, two information items were defined: a concept definition (represented by an image); and a related fact (represented by a text).

The hierarchy and placement of the states in the instructional model are in agreement with the hierarchy established in the conceptual model. For example, the state *OSI* is the parent state of *OSI Model*, *OSI Protocol* and *Layers* (which can also be seen in the conceptual model).

### 3.3. Didactic model

The didactic model is responsible for the sequencing of the content delivery. Learning paths can be established by defining the transitions among the states and the type of specification (open or closed). Correlated subjects that do not require a specific learning ordering can be represented as *DD* states, *i.e.*, the sequencing is dynamically defined by the user, in “execution time”.

For the Computer Networks module we chose to use an open specification, where all the possible presentation sequences among states (information items and instructional elements) are available. So, the learner can browse the concepts according to his/her own learning goals.



**Figure 3.**Part of the didactic model for TCP/IP Network.

Figure 3 represents part of the didactic model for TCP/IP Network. Since we are modeling an open specification, *DD* states were used for every state. Thus, from *OSI*: (1) any of its children states (*OSI Model*, *OSI Protocol* and *Layers*) can be accessed; and (2) from its children, any other child can be accessed too (e.g., it is possible to navigate from *Layers* to *OSI Protocol*).

### 3.4. Generation of slides

Considering the generation guidelines and its use against the instructional model, we produced a set of slides for the educational module. The first slide, as seen in Figure 4, whose goal is to provide an overview of the course, realizes the initially reachable states defined in the model (Figure 3). The state *TCP/IP Networks* is the main state: thus, it is in the center of the slide. As its a *DD* state, its child-states (*Layered architecture*, *Data link*, *Network*, *Transport*) must also be shown (and reachable from the slide). The same applies to the *Layered architecture* which is also a *DD*-state. Moreover, instructional elements, defined for each initially reachable state, are promptly identified in the right side of the slide: tools (network simulators), extra material (exercises, examples), and related content (related educational modules such as *Network security* and *Linux server administration*).



Figure 4. Comparison between the original *IMA-CID*-based overview slide.

Comparing the original (Figure 4a) and the new slide (Figure 4b), the generated version provides more information and navigational options. Thus, the instructor can employ different didactic approaches, such as to start an example or a simulation right at the beginning of the lesson, instead of showing and delivering instructional elements (usually plain text or static images).

Consider the state *OSI protocol* (Figure 5). It illustrates one of the original slides of the *OSI Protocol* in comparison with its new version, reengineered with the application of *IMA-CID*. The original slide, depicted in Figure 5a, was created *ad hoc*, based upon the original authors expertise. It describes the *OSI protocol*, providing some facts about how the protocol is organized (layers) and the rationale for using such architecture.

The generated slide is shown at Figure 5b. The title of the slide clearly identifies the current state (*OSI Protocol*). On the left, there is a navigation menu, that provides links to other states reachable from the current one, as specified in the didactic model (Figure 3): the parent state (*OSI*) and its sibling states (*OSI Model* and *Layers*). As *Layers* is a *DD state*, its children are also reachable, and links are also provided for them.

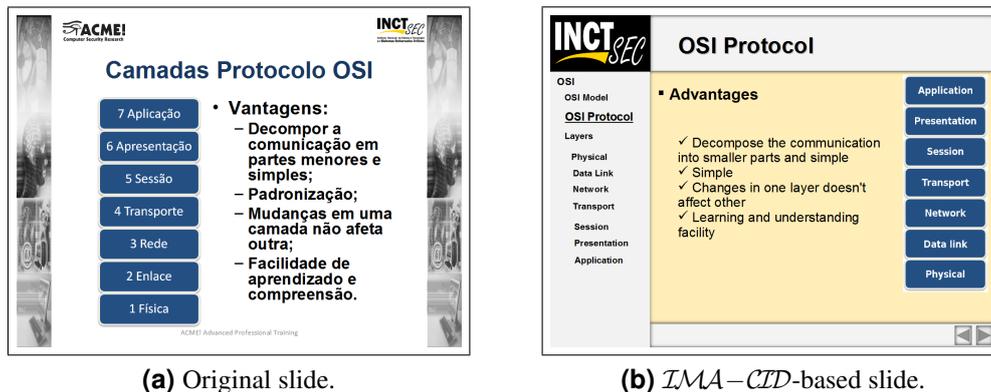


Figure 5. Comparison between the original and *IMA-CID*-based content slide.

The core of the slide has several facts, represented as text and images. The bottom right buttons (left and right arrow)s provides, respectively, links to the previous slide (definitions regarding the OSI protocol) and to the next slide (facts about drawbacks of the protocol). Both, the facts and the definitions, represents the information items defined in the didactic model. Notice that this slide/state is not associated to any instructional element (example or exercise) but, if it was the case, the link to the element would be provided by a button on the bottom-left of the slide.

An important point to be highlighted is regarding the navigation menu, defined according to an open specification. The user is able to dynamically decide which concepts of the module should be navigated and in which sequence, for instance, based on his/her profile and learning preferences.

Comparing the original and the *IMA-CID*-based slide, it can be observed that the amplitude and knowledge domain are better structured. Every concept directly related to the current concept is reachable, providing a set of effective navigational alternatives. The hierarchy among the concepts is also represented, as well as the domain specific relationship among them. Overall, the educational module generated using *IMA-CID* provides consistent and structured didactic content, that leverages the reusability and the dynamic issues of the educational process. Although it is possible to create educational content with such characteristics using an *ad hoc* approach, *IMA-CID* provides a systematic and effective solution in this regard.

#### 4. Concluding remarks

The existence of a well-defined approach to systematize the development and reengineering of educational content and, at the same time, flexible enough to be adaptable to different knowledge domains and development teams, is a significant advantage provided by using *IMA-CID*. On the other hand, the main concern regarding the *IMA-CID* approach is the need to be familiar with the structure and execution semantics of statecharts, what can lead to some additional costs to initially develop *IMA-CID*-based courses. However, the quality factors of the produced materials, such as customization, evolvability, maintainability and reusability, would increase the long-term benefits and decrease the overall costs. The use of proper tooling, such as *AIM-Tool* [Borges et al. 2011], could further reduce the overhead caused by the approach.

As future work, we intend to finish the reengineering process of the entire course on Network Security. Such experience will be helpful to the refinement of the *IMA-CTD* application guidelines and, also, for the improvement of the *IMA-CTD* features. Additionally, based on the results obtained so far, we are motivated to conduct more systematic and controlled experiments involving the *AIM-Tool* application in order to validate the ideas presented in the paper.

### Acknowledgments

The authors would like to thank FAPESP, CAPES and CNPq for their support.

### References

- Barbosa, E. F. and Maldonado, J. C. (2006). An integrated content modeling approach for educational modules. In *IFIP International Conference on Education for the 21st Century*, volume 210, pages 17–26, Santiago, Chile. Springer.
- Borges, V. A., Maldonado, J. C., and Barbosa, E. F. (2011). Towards the establishment of supporting mechanisms for modeling and generating educational content. In *ACM Symposium on Applied Computing*, pages 1202–1207. ACM.
- Horn, R. E. (1989). *Mapping Hypertext*. Lexington: Lexington Institute.
- IMS (2003). IMS learning design specification.
- Kawasaki, E. I. and Fernandes, C. T. (1996). Modelo para projeto de cursos hipermídia. In *SBIE 1996*, pages 227–240, Belo Horizonte, MG.
- Koper, R., Hermans, H., Vogten, H., Brouns, F., et al. (2000). Educational modeling language. Retrieved on March 15, 2011.
- Merrill, M. D. (1983). *Component Display Theory*. Lawrence Erlbaum, Hillsdale, NJ.
- Michener, E. R. (1978). Understanding understanding mathematics. *Cognitive Science*, 2(4):361–383.
- Novak, J. D. (1990). Concept mapping: A useful tool for science education. *Journal of Research in Science Teaching*, 27(10):937–949.
- Pansanato, L. T. E. and Nunes, M. G. V. (1999). Autoria de aplicações hipermídia para ensino. *Revista Brasileira de Informática na Educação*, 1(5):103–124.
- Paquette, G., Teja, I., Léonard, M., Lundgren-Cayrol, K., and Marino, O. (2005). An instructional engineering method and tool for the design of units of learning. In Koper, R. and Tattersall, C., editors, *Learning design: A Handbook on Modeling and Delivering Networked Education and Training*. Springer, Netherlands.
- Pimentel, M. G. (1997). MAPHE: Metodologia de apoio a projetos de hipertextos educacionais. In *SBIE 1997*, pages 351–368, São José dos Campos, SP, Brazil.
- Rodríguez-Artacho, M. and Maílo, M. F. V. (2004). Modeling educational content: the cognitive approach of the PALO language. *Educational Technology & Society*, 7(3):124–137.
- Turine, M. A. S., de Oliveira, M. C. F., and Masiero, P. C. (1997). A navigation-oriented hypertext model based on statecharts. In *ACM Conference on Hypertext*, pages 241–256. ACM.