# An Instructional Component for Dynamic Course Generation and Delivery

Carsten Ullrich

cullrich@activemath.org

**Abstract:**

E-Learning offers the advantage of interactivity: an E-Learning system can adapt the learning materials to suit the learner's personality and his goals, and it can react to the students interaction with the learning materials. Since the advent of the Web, lots of research investigated the new possibilities to support learning offered by the means of hypertext and the Internet. However, even though important results have been achieved with respect to adaptive hypermedia, open student modeling, collaborative web-tools to name but a few, current features of adaptivity fall behind compared to pre-web instructional planning systems developed in the early nineties.

In this paper, I describe an instructional component for the E-Learning environment ACTIVEMATH that marries work in instructional planning, on how to generate learning materials that are best suited for an individual learner and his tasks, and recent work for E-Learning in the Web, especially in knowledge representation, of learning materials (IEEE LOM) and the Semantic Web, i.e., RDF and OWL.

## 1 Motivation

A good teacher reacts to the needs of his students. He possesses instructional knowledge: he knows how to adapt his selection of content with respect to his student's knowledge, needs, capabilities, and goals. E-Learning has the potential to offer at last some of these instructional capabilites.

In this article, I describe a component for E-Learning systems that possesses instructional knowledge similar to a teacher and uses this knowledge to generate and deliver courses of learning materials. With respect to the scope of this workshop, technology and applications of XML-technologies, I especially highlight those parts of my work that illustrate how XML-technologies can come into play in instructional planning. First, I briefly provide some of the relevant background information on E-Learning.

## 2 Background

This section contains a succinct description of relevant research on which my work is based. Due to page constraints, this section distinguishes itself mainly for its fragmentariness.

*Instructional Knowledge.* Instructional design theories [Re83] describe how to design teaching materials that are effective (how well is learned), efficient (effectivity divided by used time), and appealing with respect to the learning goal. Instructional design is relevant for adaptive E-Learning systems because it provides information about the general instructional knowledge that systems should possess and about how this knowledge is reflected in instructional strategies.

*Instructional Planning.* Numerous approaches exist to formalize instructional knowledge and to use it for automatic teaching Peachy and McCalla [PM86] realized one of the first approaches to combine instructional knowledge and artifical intelligence techniques by applying planning techniques to generate sequence of learning materials. Subsequent work adds instructional goals (such as Bloom's taxonomy of educational objectives [Bl56]), and reactivity in an one-to-one problem solving tutoring situation [Wa90] and a course setting [Va95]. Whereas these approaches mingle instructional and domain knowledge, van Marcke developed a generic tutoring environment (GTE) [vM98]. His instructional tasks (activities accomplished by an instructor during an instructional process) and methods (methods that can be used to perform an instructional task) are to a great extent domain-independent. However, GTE fails to have an explicit representation of pedagogical control knowledge and therefore can not represent different instructional strategies. It was never ported to the Web nor does it use hypertext, and does not provide an explicit representation of instructional goals.

*Adaptive Hypermedia.* The last ten years have seen an increasing research making usable the possibilities offered by the spread of the Internet and hypertext in order to support learning. Several techniques for adaptive hypermedia have been developed (for an exhaustive overview see [Br01]). Until recently, the adaptive generation of a complete courses and the content of their pages, fell behind the capabilites of systems like GTE (but see recent systems such as ACE [SO98] or WINDS [SKPK01]).

*Knowledge Representation.* Parallel to the research in adaptive hypermedia, considerable effort has been spend investigating the description of learning materials. The part of this research concerning the metadata of single learning objects was eventually integrated in IEEE LOM [IE02]. Descriptions of learning materials collections led to IMS Content Packaging [IM03a]. Educational modeling languages [IM03b, IM03c] add pedagogical information to these collections, for instance, the pedagogical role they fulfill in a course. All these knowledge representations have XML-bindings.

*The* ACTIVEMATH *learning environment.* ACTIVEMATH [MBA$^+$01], a learning environment developed at the Saarland University and the German Research Center for Artificial Intelligence (DFKI), integrates instructional planning, adaptive hypermedia, and semantic knowledge representation. It uses instructional knowledge to generate a course adapted to the individual learner and his goals from learning objects represented in the XML-language OMDOC [Kh01]. OMDOC allows the encoding of different categories of items at paragraph level (mathematical concepts such as definitions and theorems, and further items such as examples, exercises, or elaborative texts) and their annotation with metadata.

ACTIVEMATH uses pedagogical rules to dynamically assemble the learning objects to a course. These rules define what pages for concepts should look like depending on the knowledge and goals of the learner, e.g., what elements are to appear on a page and in which order. The learner can explicitly state his learning goals by choosing among different scenarios (e.g., overview, guided tour, or Polya-inspired proof presentation).

The current course generation suffers several drawbacks, some of which are interesting with respect to the scope of this article. An explicit representation of instructional task or methods is missing, thereby hampering the implementation of additional learning strategies and the integration of third-party content. Furthermore, the complete course is generated at forehand, and reactivity is limited only. In the following section, I present those parts of my thesis that alleviate these problems by marrying instructional planning techniques and todays sophisticated knowledge representations and descriptions.

## 3 An Instructional Component for Dynamic Course Generation and Delivery

A goal of my thesis is to realize a generic, domain independent instructional component that generates courses of learning materials and supports the student while interacting with the course. As already mentioned, I restrict myself to those parts that employ XML-technologies. Other aspects of my work are described briefly, if necessary.

I focus on the following questions: How can generic instructional planning be realized? How can the generated course be made usable for third parties? How can the student be supported while interacting with the course? How can an uncomplicated integration of third-party content be achieved?

### 3.1 Instructional Planning

The goal of an instructional session is that the student achieves his learning goals. The instructional planner selects which learning objects to present and in which order.

The course generation happens the following way: The learner (or a teacher) specifies his learning goal, such as *teachApplication(addition)*. The instructional planner treats this goal as an instructional task. Other examples of instructional tasks are *summarize*, *provideMotivation*. In general, an instructional task can be fulfilled by several instructional methods by decomposing the task in sub-tasks. For instance, the task to arouse the students interest can be fulfilled by the method *EasyExamplePresentation* or *RealWorldProblemPresentation*. Instructional methods decompose tasks into subtasks until primitive, non-decomposable tasks are reached. These primitive tasks correspond to actions that act on the course itself, for instance they choose that exercise number 15 is shown as the second element on page 3.

The choice of methods depends on the instructional strategy. A strategy comprises a selection of tasks and methods, and a rating. For instance, a learning-by-doing strategy prefers methods that foster active problem solving.

Figure 1 contains an instructional plan that teaches about the mathematical concept *group*. The task *teachTopic(GROUP)* is fulfilled using the instructional method *ProblemBasedTraining*. Different choices would have been possible, for instance *ClassicDidacticApproach*. In this case however, the planner opted for a problem-based learning strategy because Eva, the current user, learns best using concrete problems (a fact represented in her user model). The planner decides to present the necessary prerequisites, together with a test (not shown in the figure), followed by the definition of *group* (the learning object with the *id group_def*) and the problem (*group_problem1*).

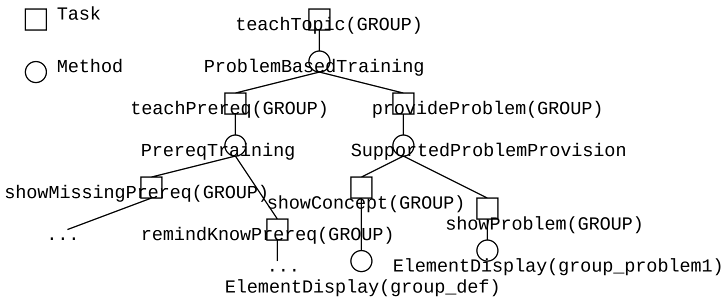


Figure 1: Instructional Plan

## 3.2 Course Data Structure

A generated course needs to be delivered to the learner. A course does not solely consist of a structured sequence of learning materials, but ideally it contains additional information, e.g., on the instructional goal of a step, on time constraints, on reactions when time constraints are violated or goals are met. Furthermore, the generated course should be usable in different learning environments, so that content can not only be imported as seen in the previous section, but also exported.

A recent standard meets these requirements. Using IMS Simple Sequencing [IM03c], an author specifies the structure (e.g., sections and subsections) of a collection of learning materials and additionally provides information on how to guide the learner through this structure. Conditional rules can make, for instance, a learner skip a section if her knowledge exceeds a given threshold.

Therefore, the result of the instructional planning will be an IMS SS structure generated from the instructional plan. Figure 2 provides the IMS SS structure from the plan of Figure 1. The first page to be presented contains the prerequisites, followed by an exercise that tests whether Eva indeed possesses the necessary knowledge, and finally the new concept and the problem.

However, using IMS SS leads to several problems. First, the complete course needs to be generated at forehand but some assumptions underlying the planning can turn out to be wrong. Second, reactions as specified in IMS SS depend solely on interactions of the learner with the content. Other properties of the user, such as his field of study, are not taken into account. Third, interactivity in IMS SS is limited. As it relies on a fixed IMS CP, it is not suited for dynamically adding or deleting content.

Whereas IMS SS seems an appropriate data structure for some dynamic aspects, we need a component that allows for real interactivity and reactivity that overcomes the limits of programmed instruction,

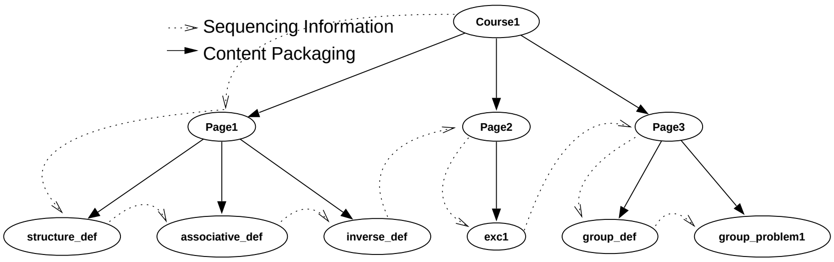


Figure 2: Simple Sequencing Datastructure

## 3.3 Delivery and Monitoring

The role of the delivery and monitoring component is to watch the actions of the user and to react, if necessary. Possible reactions consist of re-planning parts of the course and adding or removing parts of the plan. Reactions can occur, if assumptions underlying the plan generation turn out to be wrong (e.g., knowledge about concepts, motivational aspects), or, if the learner explicitly utters requests such as inserting some content or asking for the next best learning step. More specifically, by having explicitly represent an instructional vocabulary, there is no reason why the learner himself should not have access to this vocabulary and sends his own instructional goals to the planner.

With respect to the above example, if Eva fails to show her knowledge about the prerequisites, the monitoring component reacts by re-planning and inserting new, remedying nodes. To a limited extent, this information can be represented in IMS SS. However, this would require to plan ahead every possible action of the student and every sensible reaction of the system. This is not feasible in a tutorial setting as envisaged in this thesis.

## 3.4 Integrating Content

Writing learning materials is expensive and therefore the reuse of existing learning materials is an important goal. Separating structure from content, as possible with XML was a first, important step towards reusability. However, today a myriad of knowledge representations for learning materials exist. IEEE LOM [IE02] provides at least a common representation of metadata, but is insufficient for automatic usage of learning materials as envisaged in this thesis. So the challenge remains how to integrate content not written in a specific knowledge representation and not constrain unnecessarily content sources.

The idea is to define a general, abstract ontology of instructional objects and to provide the possibility of defining mappings from third-party content to these objects.

Luckily, the Semantic Web faces a similar challenge. Lots of data exists in the Web, so how can this data be described, how can it be used for reasoning? Research done on semantic nets and reasoning led to the formulation of the resource description framework RDF [Wo99] and more recently to the Web ontology language OWL [Wo02].

In the context of this paper, RDF is used to assign a semantic to a learning object. While the XML-binding of LOM prescribes the structure and only unofficially defines the semantics of the elements, RDF makes the semantics explicit. Furthermore, it allows to extend the metadata and to make the extensions accessible.

RDF descriptions refer to a defined vocabulary, in this case an instructional vocabulary. The vocabulary is described in OWL. More specifically, OWL is used for defining the instructional objects (definition, example, etc.) themselves, and the relations between these objects. Thus, it defines an generic ontology of learning objects. By using the OWL ontology mapping facilities, third-party content providers can use the representation to define a mapping from their vocabulary to the generic instructional objects. This provides a mechanism for planning with third-party content.

Figure 3 provides an example of the instructional vocabulary used by the planner (middle part) and mappings from two different content knowledge representations (OMDOC and a fictitious physic knowledge representation). By using these mappings, the instructional planner can easily apply its instructional knowledge to a different domain.

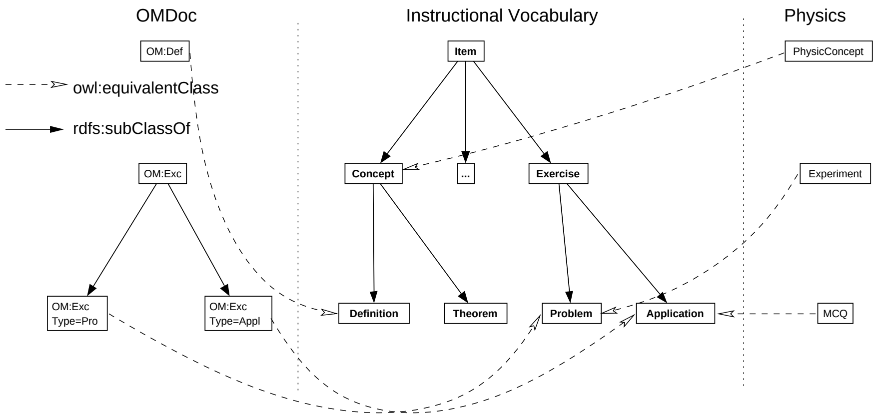


Figure 3: Different ontologies with mappings

## 4 Conclusion and Further Work

My work tackles the problems of generating courses of learning materials from different sources adapted to an individual learner and his tasks and delivering the course in a reactive manner. XML-technologies (RDF and OWL) will be applied for representing the instructional knowledge and to allow inclusion of third-party content. Existing learning standards with an XML-binding will be used for the representation of the course itself. To which degree XML can further serve as a means remains to be investigated. It may turn out to be supportive when offering the instructional planner as a Web service, so that other learning environments have a standardized way of access.

To my knowledge, no other system has made explicit the instructional knowledge to such an extend as described in this work. Several systems offer some degree of adaptivity, some allow to adapt the rules that govern the adaptivity. However, no system I know of explicitly defines an instructional vocabulary as done in GTE and none offers the flexibility for an instructor to define several strategies by referring to such a vocabulary.

Most work still needs to be done. First steps will be to investigate the instructional vocabulary of objects, tasks, and methods. Evaluations need to be undertaken to determine whether and to which degree adaptivity indeed supports learning.

## References


[Bl56] Bloom, B. (Hrsg.): *Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain*. Longmans, Green. New York, Toronto. 1956.

[Br01] Brusilovsky, P.: Adaptive hypermedia. *User Modeling and User Adapted Interaction*. 11(1/2):87–110. 2001.

[IE02] IEEE. 1484.12.1-2002 IEEE standard for learning object metadata. 2002.

[IM03a] IMS Global Learning Consortium. IMS content packaging information model. June 2003.

[IM03b] IMS Global Learning Consortium. IMS learning design specification. February 2003.

[IM03c] IMS Global Learning Consortium. IMS simple sequencing specification. March 2003.

[Kh01] Kohlhase, M.: OMDOC: Towards an internet standard for mathematical knowledge. In: Lozano, E. R. (Hrsg.), *Proceedings of AISC'2000*. LNAI. Springer Verlag. 2001.

[MBA+01] Melis, E., Büdenbender, J., Andres, E., Frischauf, A., Goguadze, G., Libbrecht, P., Pollet, M., und Ullrich, C.: Activemath: A generic and adaptive web-based learning environment. *Artificial Intelligence and Education*. 12(4). 2001.

[PM86] Peachy, D. R. und McCalla, G. I.: Using planning techniques in intelligent tutoring systems. *International Journal of Man-Machine Studies*. 24:77–98. 1986.

[Re83] Reigeluth, C. M. (Hrsg.): *Instructional Design Theories and Models: An Overview on their Current Status*. volume I. Lawrence Erlbaum Associates. Hillsdale, NJ. 1983.

[SKPK01] Specht, M., Kravcik, M., Pesin, L., und Klemke, R.: Authoring adaptive educational hypermedia in WINDS. In: Henze, N. (Hrsg.), *Proc. of the ABIS 2001 Workshop*. 2001.

[SO98] Specht, M. und Oppermann, R.: ACE - adaptive courseware environment. *The New Review of Hypermedia and Multimedia*. 4:141–162. 1998.

[Va95] Vassileva, J.: Dynamic courseware generation: at the cross point of cal, its and authoring. In: *Proceedings of ICCE'95, Singapore*. S. 290–297. December 1995.

[vM98] van Marcke, K.: GTE: An epistemological approach to instructional modeling. *Instructional Science*. 26:147–191. 1998.

[Wa90] Wasson, B. J.: *Determining the Focus of Instruction: Content planning for intelligent tutoring systems*. PhD thesis. Department of Computational Science, University of Saskatchewan. 1990. Research Report 90-5.

[Wo99] World Wide Web Consortium. Resource description framework (RDF) model and syntax specification. February 1999.

[Wo02] World Wide Web Consortium. Web ontology language (OWL) abstract syntax and semantics. November 2002.