Generating Instructional Hypermedia with APHID

Judi R. Thomson
Department of Computer Science
University of Alberta
Edmonton, AB, Canada
E-mail: thomson@cs.ualberta.ca

Jim Greer, John Cooke
Department of Computer Science
University of Saskatchewan
Saskatoon, SK, Canada
E-mail: {greer, cooke}@cs.usask.ca

ABSTRACT
We propose a method (APHID) that assists an instructional designer to define format, structure and sequence within an instructional hypermedia application. Our method uses concept maps and instructional patterns, as well as data, navigation, and presentation models to support partial automation for creating instructional hypermedia.

KEYWORDS: instructional hypermedia, XML, concept maps, instructional patterns

INTRODUCTION
The goal of instruction is to communicate the concepts of a certain domain to learners. There are instructional strategies that help an instructor decide when to introduce, when to review, or when to teach for detail. Hypermedia applications that are intended for instruction must reflect these different instructional strategies, as well as the concepts to be learned, in their design and presentation.

This paper presents a process for creating instructional hypermedia applications called *Applied Patterns for Hypermedia Instructional Design (APHID).* APHID is a structured approach to hypermedia design that uses several models. It uses a set of models similar to the three presented in OOHDM [3]. One difference from OOHDM is that APHID explicitly represents instructional components and sequences as a fourth model.

A prototype environment has been developed that uses the APHID approach to assist instructional designers to create hypermedia applications. Instructors create an explicit, detailed, model of the domain of instruction that includes a description of the relationships between domain concepts. Instructors also identify the instructional content to be used and meta-information about that content, and then add these to the model of the domain. The APHID software automatically generates the navigation model and applies presentation rules to it to produce a hypermedia application.

Hypertext 2000, San Antonio, TX.

Support for flexible instructional design is provided by allowing instructors to adjust design attributes of the finished application to match the specific needs of the students. Novice designers of instructional material are supported by the definition (and implementation) of several instructional patterns, which are presented to the designer as instructional strategies. These patterns provide descriptions of teaching strategies known to be successful in particular situations. The patterns also give suggestions of how to incorporate those teaching strategies into the design of a hypermedia application.

For example, one instructional pattern (spiral teaching) suggests that novice learners benefit from a teaching approach that presents material in small steps and that students revisit previously taught material before covering new material. In this particular case, APHID would generate an application in which concepts would naturally be reviewed at least twice. Introductory pages about a concept would present simpler material than subsequent pages and the amount of information presented to the learner at once would be kept small. All of these settings can be customized by the designer to refine the spiral curriculum strategy, but APHID suggests guidelines and default values for all of the required attributes.

MODELLING
The difference between instructional hypermedia and any other sort of hypermedia is that the former requires two kinds of navigation: navigation that follows from the model of the domain, and navigation that follows from the model for the sequencing of instruction[1]. The APHID environment utilizes both a domain model and a semantic or instructional model to facilitate the representation of both kinds of navigation.

Within APHID, the domain model consists, in part, of a model of the types of instructional materials, such as explanations, quizzes or simulations, which make up the application. These types are called *instructional classes*. The second part of the data model is meta-information about the interacting objects that make up the hypermedia application. Each such object is called a *data element*. A data element is, in one sense, an instance of an instructional class.

The semantic model provides the information necessary for ordering the instruction. Sequencing of information, from an instructional perspective, is dependent on the structure of the knowledge being taught, rather than on the static structure of

the data. The semantic structure can be represented in a concept map [2] that is then used to guide the sequencing and organization of the hypermedia application.

CREATING HYPERMEDIA

APHID provides a support environment in which instructors can design and create well-built hypermedia applications with a minimum expenditure of time. From the instructional designer's point of view, the creation of a hypermedia application (once the models are complete) is as simple as setting two parameters (type of application and level of user) and clicking a button. The APHID software decides which concepts to include, which data elements to include, and which hyperlinks to create between pages, based on the models and the supplied parameters.

Given the graph-representation of a domain (the concept map), APHID traverses the graph and at every node determines if the concept represented by that node should be included or excluded from the final application. If the concept is included, APHID selects the appropriate data elements and the necessary links to other concepts, and continues. The result is a list of navigational nodes in *tutorial-order*, the order in which nodes would be presented if a linear tutorial were constructed. Most hypermedia applications created with APHID are not linear, instead the