A Knowledge-Based Instructional Assistant to Accompany LEO:
A Learning Environment Organizer

John W. Coffey
The Institute for Human and Machine Cognition
and The Department of Computer Science
The University of West Florida
Pensacola FL 32514
jcoffey@ai.uwf.edu

Abstract
This paper describes preliminary work on enhancements
to a computer-mediated instructional program entitled
LEO: a Learning Environment Organizer. LEO is a non-
linear course presentation program that is based upon the
notion of an advance organizer (Ausubel 1968). The
Knowledge-based Instructional Assistant described here
combines knowledge of the learner’s attainment in the
course with knowledge of various attributes of the
instructional media in order to make recommendations of
media that might benefit the learner. Matches between
attainment profiles and media are based upon weighted
nearest neighbor measures (Wettschereck & Aha 1995) of
similarity.

Introduction
This work takes its impetus from several essential
elements of Ausubel's (1968) Assimilation Theory. The
most fundamental idea of Ausubel's theory is to
"determine what a student knows and teach accordingly."
Ausubel was, of course, referring to a strategy for
traditional teachers in traditional face-to-face classroom
settings. The principle still holds (perhaps especially so)
in distance learning settings that tend to create greater
difficulties for teachers in their efforts to assess and
understand what their students know.

A computer program entitled LEO, a Learning
Environment Organizer (Coffey 2000; Coffey & Cañas
2001), provides the framework for the approach described
here. LEO is an editor/browser based upon concept maps
(Novak & Gowin 1984). LEO presents a graphical
representation of the topics in a course of study, essential
dependency or prerequisite relationships among the topics
(if any exist), and links to instructional media that contain
content pertinent to the topic. The student utilizes the
organizer to view and access information pertaining to the
topics to be studied. The system tracks student progress
through the topics, displaying an indication of those that
have been completed and those that have not.

This basic system is augmented by a Knowledge-
Based Instructional Assistant that can combine knowledge
of the content in a course with knowledge of the student,
in order to suggest materials that might be of interest to
the student. These suggestions are based upon an
emerging learner attainment profile that starts with a
pretest and evolves as a student works through the course,
taking tests and having deliverables evaluated. The
remainder of this paper will present a brief survey of work
in knowledge-based instructional systems, a description
of LEO, and a discussion of how the Knowledge-Based
Instructional Assistant (KnowBIA) augments LEO's
capabilities.

Knowledge-Based Instructional Systems
The idea of knowledge-based instructional systems
arises from the fields of user modeling (Finin & Drager
1986), adaptive hypermedia (Brusilovsky 1996), and
intelligent tutoring systems (Burns & Parlet 1991). A
basic distinction in knowledge-based instructional
systems can be made between those that are generic – that
are not constituted in a specific knowledge domain – and
those that are domain specific. Recent literature has
described generic tutoring system approaches such as a
multi-agent approach that uses planning techniques to
plan tutoring actions and presentations (Nkmbmou &
Kabanza 2001), case-driven approaches such as Riesbeck
and Schank's case-based tutors (Riesbeck & Schank 1991)
and Goal-based Scenarios (Schank 1996).

Domain-specific systems have been created in areas
that include the teaching of language (Heft 2001; Mayo
Mitrovic & McKenzie 2000), algebra (Canfield 2001;
Virvou & Moundridou, 2000), cataloging (DeSilva &
Zainab 2000), etc. Typical approaches of these systems
include assessing student work at problem solving and
pointing out errors in the student's solution. Others
provide suggestions for the improvement of the emerging
solution. Some provide pertinent example solutions to
help the student solve problems. Other adaptive systems
may allow the user to control the amount of intervention
the system provides (Kay 2001). The current work
describes a generic system shell used to create environ-
ments in specific instructional domains that recommend
instructional materials for the individual student.
The Learning Environment Organizer

This section describes LEO, an organizer that is part of a distributed knowledge modeling system named CMapTools (Cañas et al. 1998). A knowledge model created with CMapTools is a browsable hypermedia that provides instructional content that may be organized with LEO. Figure 1 presents a view of an Organizer pertaining to a computer science course entitled Data Structures.

An Organizer takes the form of a graph with two different types of nodes, instructional topic nodes, and explanation nodes. Topic nodes correspond to the topics in the course and are adorned with icons providing links to content and assignment descriptions, and an indicator of the completion requirements for the topic. The topic nodes are linked together by double lines that convey prerequisite relationships. The icons beneath the topic nodes represent links to online instructional content.

When the user clicks on an icon associated with a topic, a pull-down menu appears that presents a selection of links to the content that is pertinent to the topic. The topic nodes have color codings to indicate student progress through the course of instruction. Explanation nodes elaborate the relationships among the topic nodes and have no adornments.

As shown in Figure 1, the Organizer presents the user a navigation tool that affords a global, context view of the entire organizer (the small rectangle in the upper left), a local (focus) view of the course structure, and a “Display Status Panel” that enables the user to show or hide subsets of the organizer.
Students must be registered with the system and log on to work on a course. Once the student has logged in, the system retrieves the progress record associated with the userid and the Organizer for which the logon occurred. The progress record contains information on the student's basic progress by topic, submissions of deliverables, whether the deliverables have been graded, etc. The attainment profile indicates level of mastery of various dimensions pertaining to a topic.

**The Knowledge-Based Instructional Assistant**

Figure 2 depicts how the Knowledge-Based Instructional Assistant fits into the Organizer system. The user starts with a pretest to establish baseline knowledge for the course, in the form of the profile of strengths and weaknesses in individual topic areas. Using information in this profile, the system recommends media that address deficiencies. As the student works through the course within LEO, the recommendations of the system are made known through the media selection mechanism. As illustrated by Figure 1, a specific resource may be suggested by placing the word "Recommended" next to the menu item that allows its selection. As the student works through the topics in the organizer, his/her competency profile is updated, as are the resources that are suggested for the student to peruse.

The remainder of this section identifies and discusses issues pertaining to the cataloguing of resources, the nature of the student assessment that drives the system, the nature of the links established by the Instructional Assistant between test questions and the content, and the way in which the student utilizes the services of the Assistant.

**Cataloging Resources and Test Items**

Resources and test items are catalogued by multiple attributes that include the topic or topic areas (by rank order), depth or complexity of the item, a resource's size (brief vs protracted), a description such as whether the resource/question is conceptual or procedural in nature, the volatility of the resource (how susceptible it is to change/obsolescence) the certainty of the resource, etc. Resource attributes will be represented and queried as searchable XML metadata records (Coffey & Cañas 2001) in order to create matches.

The development of an indexing vocabulary is domain specific (Kolodner & Leake 1995) requiring basic data with context-dependant extensions. A fundamental issue is the granularity with regard to how specific the indexing should attempt to be. Kolodner and Leake suggest that it is impossible to determine a set of attributes that completely describe a domain. To account for such difficulties, multiple parameters must exist and an overlap must exist between the attributes tested in a question and the attributes ascribed to a piece of instructional content.

**Student Assessment**

A critical component of this system is the ability to assess what the student knows. The student takes a pretest that leads to an initial, relatively low-fidelity profile of what s/he knows. The profile creates a multi-dimensional picture of individual strengths and weaknesses of the student for each of the topics in the course. For example, the pretest might contain both conceptual and procedural questions in gradations from elementary to advanced for the target audience. The various dimensions of interest for testing purposes are domain-specific.
The testing component can present questions in the domain of interest with answers that are rendered as multiple choices evaluated by the system, or as questions for the instructor to grade. Although an approach based solely on test performance permits automatic matching of resources, it is anticipated that the instructor will be able to edit the student profile manually, based upon student submissions and other items that can be evaluated. The student’s capabilities are mapped along the various dimensions for each of the topics of interest in the course. As described in the next section, the competency profile is mapped to the media in the course dynamically, to flag those items the system suggests the student review.

Matching Instructional Resources to the Student Profile

The ideas in this work leverage the integration of LEO with CMapTools. CMapTools can be used to produce browsable multimedia knowledge models that serve as the instructional content. Such knowledge models are comprised of many learning resources – particularly concept maps that represent the structural knowledge in the domain, and accompanying resources such as texts, graphics, video and web pages that elaborate the important concepts in the concept maps.

The process of matching possibly useful instructional materials to the attainment profile is informed by methods of case retrieval in case-based reasoning approaches. The indexing mechanism works on multiple indices that can be weighted for importance, with (for example) higher weights for the conceptual-process dimension and the difficulty dimension. The search for media identifies media relevant to deficiencies in the student's attainment, and, depending on the strength of the match, the assistant can make a relatively stronger or weaker recommendation.

The system creates matches on the basis of student strengths and weaknesses in topic areas and suggests resources that vary from conceptual to drill and practice. As an example, if the profile indicates that a student is missing conceptual knowledge in an area, it might suggest foundational readings. If the student has deficiencies in applications of knowledge, the system might point the student at problems to be solved. This manner of profiling and suggestion addresses an existing problem in LEO of having a single cutoff point at which the student is deemed to have completed a topic. Topics that were successfully completed by minimal attainment can remain on the agenda of things that the system suggests the student revisit.

The assistant necessarily starts with relatively little information about the student. Over time, the system can determine a user preference for media types that can figure into the weighing. Depending on the strength of the match with the profile, the assistant could take incremental actions from minimally to strongly suggesting that the student examine a certain item.

Two potential ways for the student to utilize the assistant's advice have been identified – by noting recommendations when accessing media from the pull-down menus, and by use of a query system. The first approach is demonstrated in Figure 1 where the fourth item in the pull-down menu has a "Recommended" indication. Figure 3 illustrates the second approach – a query dialog in which the student can view and select media that are recommended by the system. The user can select whether to view all the resources in a given course, or only those the system recommends.

matching instructional resources to the student profile

292 FLAIRS 2002
References


