

# A laboratory support framework based on a networked knowledge representation

Cédric Boey  
Université Libre De Bruxelles, BEAMS  
cboey@ulb.ac.be

Vincent Raman<sup>1</sup>, Marc Jaumain<sup>2</sup>, Frédéric Robert<sup>3</sup>

**Abstract** - We present a computer-based environment (or framework) called AUTONOMY specifically targeted to support and accelerate individual student learning during laboratory sessions. This environment results from several years of analysis of electronics laboratories at the Université Libre de Bruxelles, Applied Science Faculty. This framework basically allows the student to simultaneously:

- follow the laboratory protocol,
- browse several courses (text, figures, sound, videos) connected together and with the lab protocol via hypertext links: pedagogical content is presented in a "multi-scenario" way
- play additional interactive multimedia tools (typically JAVA applets, FLASH animations, etc)
- browse internet (if allowed) to find additional external information

The database contains:

- Courses stored like trees whose sheets are (sub)chapters
- Knowledge items, answering basic questions, or leading further in the comprehension, but not directly shown in the tree
- Links between the course (sub)chapters and knowledge items in a bidirectional way, which makes it possible to represent knowledge in a multidimensional network rather than in a linear way.

By providing to the student various and simultaneous ways of accessing information of various types and levels, our environment is expected to increase the quality of the learning during the lab

*Index Terms* – knowledge, computer, laboratory, multimedia, interactivity

## INTRODUCTION

Many teaching assistants of engineering related courses are frustrated people: they would like to develop the autonomy,

initiative and critical thinking of students during the labs while spending not too much time on answering often same questions : basic and common questions on both principles and measurement instruments manipulation. To help them, we have developed a framework called AUTONOMY that students can use to answer their questions before asking the teaching assistant. This tool is an interface to a database of knowledge about electronics containing text, images and flash animations organized by topics and turn it into an html description that can be displayed. The framework has been fully implemented in JAVA and parts of several courses have successfully been entered into the database that still needs to be completed before we could realize some real tests with students to prove its efficiency.

## CONTEXT

This project fit into different courses of the electronic department of the Université Libre de Bruxelles. These courses start at the beginning of the third year. That knowledge of previous disciplines such as physics or electricity is required. The main subjects of these courses are fundamentals like diodes, transistors and operational amplifiers together of the basics of digital electronics. All the future graduated engineers have to go through these courses so that educational efforts are very rewarding.

## PEDAGOGICAL APPROACH

As described above, the main aim of this framework is to encourage the development of the autonomy of students. However, we would like this application achieve other aims.

*1. Allow the teaching assistant to concentrate on higher level interaction.*

Bloom[1] described one taxonomy(revised by Anderson[2]) of learning containing six levels which are ranging from recalling a fact to ability to make a synthesis and to evaluate.

<sup>1</sup>Vincent Raman, Université Libre De Bruxelles, BEAMS, vraman@ulb.ac.be

<sup>2</sup>Marc Jaumain, Université Libre De Bruxelles, BEAMS, mjaumain@ulb.ac.be

<sup>3</sup>Frédéric Robert, Université Libre De Bruxelles, BEAMS, f robert@ulb.ac.be

During the lab we would like students to develop abilities in the higher steps of this taxonomy. However a step can't be broached before the previous has not been mastered. One of the major aims of this application is to allow the teacher working on the higher levels by providing students with lower levels (facts, principles ...) answers available in the repository.

### II. Easy and quickly answers to frequent questions

The framework offers the possibility to the teacher to modify the contents of a page very quickly. When he detects a common error or misconception, he can directly create a page correcting it.

### III. Better individualization of student learning scenario.

The structures of the application permits to build several scenarios of the progress of an exercise or a lab. In the majority of cases, the progress is linear and doesn't allow of the best students a way to go further in the comprehension and, in the same way, he can't allow student to adapt his learning speed. We would here offer different scenarios in order to adapt the teaching situation depending of the student. If a student shows a weakness, the application can propose additional explanations or exercises

### IV. Break frontiers often perceived between lectures, exercises, labs and scientific domains

Many students have a very compartmentalized perception of the different courses and they don't make the link between them. Thanks to appropriate reminder of previous courses, we would put back in question this conception. In the same way, both lessons and labs will seem more coherent than they were before.

### V. Encourage students to look further in the discipline.

The application contains the possibility to add external links. We want to develop initiative and stimulate student's curiosity. In this way, the framework will provide students with some links about additional information

## APPLICATION DESCRIPTION

The computer-based environment is composed out of a relational database, a content management software and a web interface

### I. Database Structure

Autonomy depends on a MySQL database whose structure is made up of 3 main parts.

The first part represents course bodies. Courses are stored like trees whose nodes and leaves are (sub)chapters. Each node (or leaf) has a title, a brief description and some keywords to make search across nodes easy.

The second part represents any knowledge useful for the students. Each knowledge item has a title, a brief description, some keywords to make search across knowledge items easy and an HTML representation. This HTML representation, which can contain images, FLASH animations or JAVA applets, is needed to visualize the knowledge item content in a web browser in a didactical way. The knowledge items can

contain course chapter, answer to basic questions that students often ask or lead further in the comprehension.

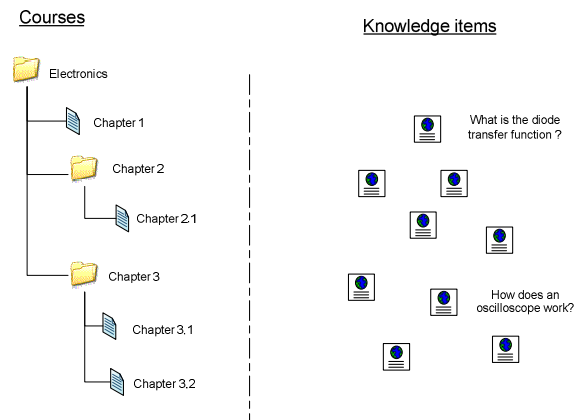


FIGURE 1  
REPRESENTATION OF COURSES AND KNOWLEDGE ITEMS

The third and last part represents the links between all of these items. Every leaf of a course tree can contain one or more knowledge items and its visualization will be a sequence of the HTML representations of these knowledge items. One knowledge item can, of course, belong to more than one leaf or to no leaf at all. In addition, every leaf and every knowledge item can be related to other leaves or other knowledge items so that a list of related, and interesting, topics will be displayed in the browser when the leaf or the knowledge item is visualized.

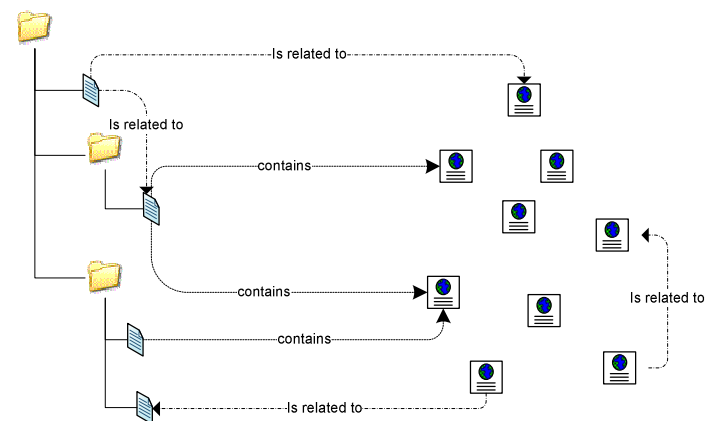


FIGURE 2  
LINKS BETWEEN COURSES AND KNOWLEDGE ITEMS

Such a structure allows the student to browse courses in a classic rigid way but with multiple links to knowledge that he could need during his lecture. In addition, the student can travel across knowledge just by searching keywords or jumping from link to link so that knowledge is represented in a multidimensional network rather than in a linear way.

This knowledge structure suits any scientific course, like electronics, and particularly the laboratory sessions. The courses and laboratories follow a step by step structure. Nevertheless during the course lecture or the laboratory,

students may need to remember knowledge seen before and which are the basis for understanding a point. For instance, transistor mechanism is a knowledge viewed in one of the very first chapters but which goes beyond and is needed all the time. Other knowledge items cannot be classified in a course chapter but, however, must be known by everyone. For example, in their courses, students are not explained how to use an oscilloscope but they need this knowledge to achieve the laboratory sessions. With this structure, students can always find easily and rapidly the information they seek.

## II. Content management

The content management software is a .NET MDI (Multiple Document Interface) application, used by a professor or a teaching assistant, which communicates with the MySQL database to add and modify Autonomy knowledge content.

The first window allows creating knowledge items. Title, description and keywords can be added by filling textboxes. Title and keywords are double-leveled: the title and the keywords that the professor sees and uses in his content management software are different of the ones that the student will see and use in the web interface. This is because the professor concentrates in the pedagogical side when he manages the knowledge content while the student will focus on the informative side. The search across the items is then sensibly different according to the context.

The HTML representation is modified thanks to a WYSIWYG (What You See Is What You Get) HTML

editor. Content can be pasted from all common applications like Microsoft Word or Microsoft PowerPoint: text format is converted in html and images are automatically generated for special items like equations. FLASH animations and JAVA applets can be added too. When a knowledge item is saved, all images, FLASH animations and JAVA applets are moved to the web server and all links to these are updated. With this tool, the professor can fill knowledge items with really didactic and interactive content. Links to other knowledge items or to course tree nodes can be also directly added in the text itself to make interactions more direct.

The second window allows creating course trees. For each course, nodes can be added, with title, description and keywords, again with a double-leveled view. A node is declared as a node, i.e. other nodes or leaves will be its children, or as a leaf. A leaf has no children but contains several knowledge items which can be ordered freely by the professor. The leaf HTML visualization will be the sequence of the knowledge items HTML representations.

The screenshot shows the 'FAT2' software interface for editing a knowledge item. On the left, a table lists various topics like 'caractéristique i-v d'une diode', 'Définition d'une diode', and 'Utilisations typiques d'un RC passe-bas'. The main area is a WYSIWYG editor for the selected item, 'caractéristique i-v d'une diode (modèle angle droit)'. The editor contains a graph of current  $I_A$  versus voltage  $V_{AK}$ . The graph shows a horizontal line at  $I_A = 0$  labeled 'diode bloquante' and a vertical line at  $V_{AK} = 0.6V$  labeled 'diode passante'. A threshold voltage  $V_{TH} = 0,6V$  is indicated. To the left of the graph is a diode symbol with arrows for current  $I_A$  and voltage  $V_{AK}$ . Below the graph, there is text explaining the diode's behavior and a rich text editor toolbar with buttons for 'Sauver' and 'Annuler'.

FIGURE 3  
KNOWLEDGE ITEM EDITOR WINDOW

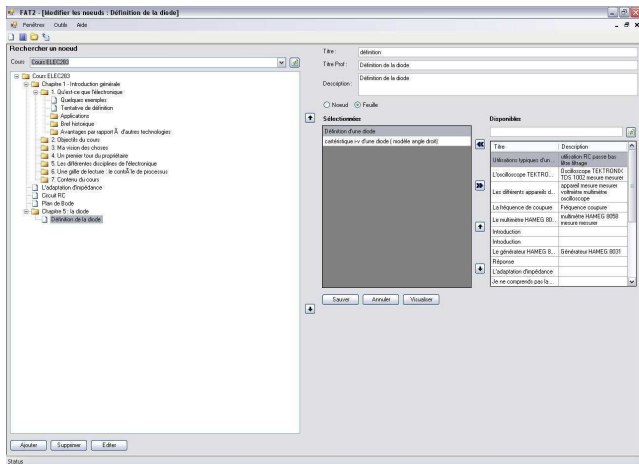


FIGURE 4  
COURSE TREE NODE EDITOR WINDOW

The third window enables linking knowledge items and tree nodes together. The left part is used to select the item (a course tree leaf or a knowledge item) that will be linked to other items. The right part is used to select all the items linked to the selected item. These items, according to their type, are ordered freely by the professor. The links are unidirectional, they will be shown when the item is visualized in the web application to help the student to have a better comprehension of the visualized item.

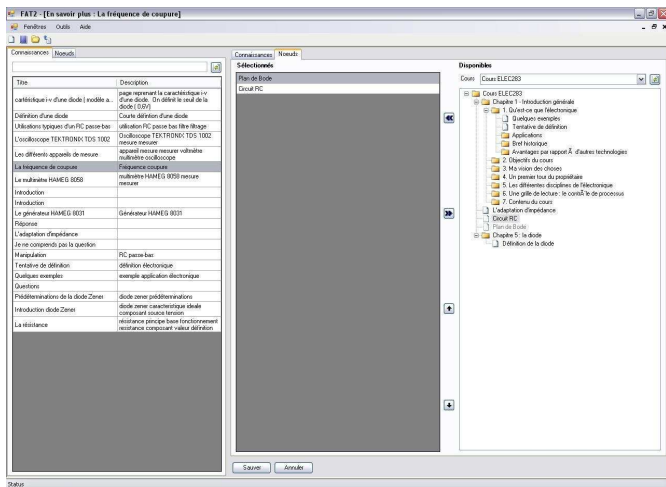


FIGURE 5  
LINK EDITOR WINDOW

### III. Web interface

The student uses a web interface to navigate through the knowledge structure. The upper part contains links to all the courses stored in the database and a search tool. All courses are visible to encourage students not to stay in a single course scope and to make links between previous courses and the present one.

When a course (or a laboratory) is selected, its tree, with selectable leaves, is developed in the left part. The tree is always shown during the navigation to help students to keep in mind the course linear process. Forward and backward links in the lower part even allows a linear navigation across leaves.

Leaves contain HTML content and are visualized in the center part. HTML content of the visualized leaf knowledge items are displayed one above the other. All the links encoded between the leaf or one of its knowledge items and other leaves or knowledge items are displayed in the right column of the center part. These links are separated in two categories. The first one consists in the links to leaves of the same course that are directly in relation with the topic. The second one consists in the links to separate knowledge items or leaves from other courses. These links stimulate student curiosity by regrouping topics which are not from the course itself. They can lead further in the comprehension of the subject or can be useful to accomplish a task.

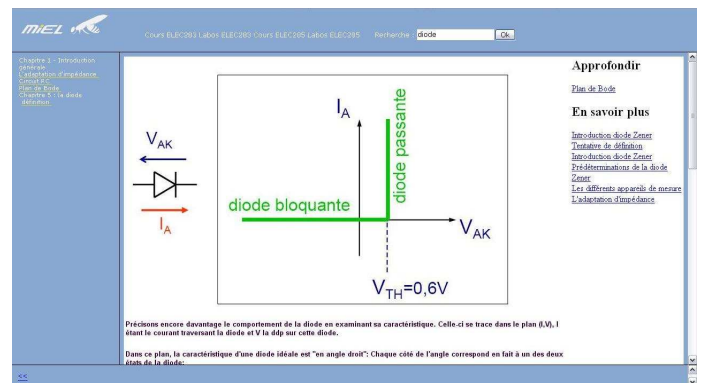


FIGURE 6  
WEB APPLICATION EXAMPLE

The search tool enables finding a specific course leaf or knowledge item in the database. The student enters several keywords and the tool displays all the leaves and the knowledge items corresponding to all the keywords. The leaves are grouped by course and the separate knowledge items are grouped together in a subcategory. This makes it easy to rapidly find the answer to a question that a student could ask himself.



FIGURE 7  
SEARCH TOOL

### CONCLUSIONS AND PERSPECTIVES

In this paper we have presented a tool simplifying and enhancing teaching conditions during electronics labs sessions. The framework generates HTML browsable content based on *knowledge items* structured into a database. To encode the data, all common office applications based documents can be cut and pasted into our tool and is

afterwards automatically translated into HTML. Furthermore knowledge is represented in the database differently for management purpose (at the teacher point of view) and for navigating purpose (at the student point of view). Some chapters have already been encoded showing that it is possible to quickly and easily extend it to a whole course. However the framework will only become fully useful as soon as multiple courses will be encoded into the database. In the future we will set up laboratory test protocols to evaluate the impact of our framework on student learning speed.

#### **ACKNOWLEDGMENTS**

The authors would like to thank the team of the CTE (Centre des technologies au service de l'enseignement) and Martin Hiernaux for their collaboration

#### **REFERENCES**

- [1] Bloom B. S. "Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain". New York: David McKay Co Inc, 1956
- [2] Anderson, Krathwohl," A Taxonomy for Learning, Teaching, and Assessing: A revision of Bloom's Taxonomy of Educational Objectives.", 2001