Virtual Activities Framework as a Methodological Proposal in the European Higher Education Area

J.A. ÁLVAREZ, J. ROCA, E.M.GARZÓN, J.F.SANJUÁN
Department of Computer Architecture and Electronics
University of Almería
04120 Almería
Spain
http://www.ace.ual.es/~jaberme

Abstract: - An important reflection about methodologies to improve the learning process is being prompted by the recent educational shift trend towards EHEA. It is a must that the learning profile of the student should be adaptative, all knowledge should be acquired by student in such a way that it can be adapted to quickly changing environments. In order to improve and promote the student's learning stage, we have developed several software tools. These tools have been integrated inside the virtual campus tool (webCT) provided by our University. Such a toolkit pretends to make easier the process of acquiring certain competences, marked as objective, for two subjects related to computer architecture curricula, by guided autonomous learning.

Key-Words: - European Higher Education Area, Computer Architecture Curricula, Educational Methodology.

1. Introduction

A new educational paradigm definition -within university education-, has emphasized the need of understanding education in a wider context where the student can acquire, update -this also means the ability to find and maintain information sources- and select appropriate knowledge for a certain context, as stated in many international and national EEES related forums [1,2,3].

In short, the student is to learn in such a way that all knowledge he/she posses can be adapted to new and changing situations, that is, the student should develop the ability of continuous learning. The new educational paradigm is student centered; this involves several changes in the:

- Definition of the objectives for each subject.
- Methodology of educational activities.
- Procedure of student evaluation.
- Role of the teacher, shifted to be a tutor for the process of learning, helping the student to reach certain competences.

Imminent imposition of the European Credit Transfer System (ECTS) forces us to reorganize current educational methodology in use for developing high education.

In this paper we present the activities carried out in the framework of subjects such as Computer Organization, Computer technology and Advanced Peripherals Devices[4], related to Computer Architecture curricula. These activities were mainly based on using tools we have developed and integrated inside the virtual learning environment provided by our University.

The description of this methodological experience is organized into two sections, the former section begins with an interesting reflection about the profile which we must develop EHEA oriented activities with, to be inserted into Computer Sciences curricula subjects.

Furthermore, we describe the activities that have been experimented and its relation with the competences defined in [5]. Closing this section we present an analysis where data from students' surveys and data from teachers' own experience are considered. Latter section describes our main conclusions and future work.

2. Active Methodologies

New methodologies adopted under EHEA must be more ambitious. They must provide knowledge as former methodologies did, but now offering tips to easily extract hidden conclusions from this base knowledge, key for better assimilation of concepts, moreover students must face a new way of solving problems. This new methodology must provide these tips using specific tools for each area. Teaching skills are, also, of valuable help when designing extra activities for a certain subject.

2.1. Learning Under New Methodologies

Each taught lesson, ideally, should be a tailored lesson for each student individually, because each student has different cognitive patterns. Students, who catch concepts quickly use to question themselves continuously about already explained stuff. On the other hand, non-outstanding students may require more time, detailed explanations and also continuous reminders on key concepts. The student may not learn with the easiness that he is supposed to do. This last idea suggests a necessary shift from current scene where the teacher is the main character to the more convenient scene where both, student and teacher, have the same importance.

The teacher must have an accurate point of what the student already knows, so the lesson can be so simple that nobody misunderstands it. It is also necessary that the student can experiment the "teach-to-learn" activity. To allow this, several activities were designed for the student to check theoretical concepts by means of simulation software inside a virtual environment, this way the student can play the teacher role for himself. Through these activities it is possible to inspect the degree of assimilation achieved in different subjects' sections with the temporary versatility offered by these environments. This way we intend to put the student nearer to the subjects' conceptual map. As the student goes getting the conceptual map, a guide must be present, no matter what the time is it, it needs to be there when the student needs it.

TABLE I
SUBJECTS WHERE THE EXPERIENCE WAS CARRIED OUT.

Car.	Subject Name	T	C	Al/gT	Al/gP	Atot
Obligatory	Computer Organization and Technology	ITIS	1º	125	30	125
	Computer Organization and Technology	ITIG	1º	72	30	144
Elect ive	Advanced Peripheral Devices	ITIS e ITIG	2º	40	20	40

A virtual tutoring system was enabled during this experience so the students could solve their problems without restrictions.

The Table I shows a description of the subjects we concentrated on during this experience, named as "Car" we refer to type of subject, named as "T" we refer to career, named as "C" we refer to the subject's course, named as "Al/gT" we refer to the number of students per theory group, named as "Al/gP" we refer to the number of students per practice group, named as "Atot" we refer to the total amount of students. Subjects, described in the aforementioned table, conforms the context in which the activities were carried out during 2005/2006 course.

As mentioned all these activities were designed taking into account transversal and specific competences defined in [5]. Table II states the relationship between these activities and the corresponding competences.

TABLE II

Activity	Competences
Evaluation of practical exercises by running simulations in a virtual environment.	***Transverses*** Autonomous learning Knowledge of a second language Synthesis and analysis skills ***Specific*** Technical documentation Understanding and evaluation skills Internal and external specs. Hardware technology Programming Computer Architecture

All these activities were defined on a voluntary basis, but of course, they were evaluated for good in the students' final evaluation. The evaluation criteria was defined from scratch, students were informed that their participation would be an important part of their evaluations. We thought this was a convenient way for us to invite them to enter into a new working concept they were not already prepared.

Taking, as a reference, past activities carried out during the last course [6], we decided to limit the number of activities to those that used simulators within the virtual environment, to avoid students' to-do list overhead caused by the convergence of two methodologies, the one in use and the one in practice.

3. Developed Tools.

The addition of support tools for teaching, as WebCT for example, makes easy to extend teacher's office hours, so the student can find a near (in time and in place) answer to his/hers doubts.

The student now is to play active learning, and the new methodology must follow appropriate routes to reach this destiny. So activities should be designed with such an objective in mind. We integrated into

WebCt tools that benefits aforementioned activities. WebCT is a powerful tool developed in perl, this is an issue that empowers WebCT for administrative jobs. Uploads of any kind of stuff into a subject's WebCT environment is done, thanks to intuitive file managers that makes possible loads and downloads of files to and from WebCT.

As said, WebCT via its file managers allowed us to include our tools into the virtual environment. Our tools were developed using Java just to take advantage of the Internet/Web capabilities of this language as well as its portability. When including any of our tools into WebCT everything comes down to a simple translation of files and a subsequent linkage process of newly uploaded java program files with appropriate graphical/visual elements as web pages, this way the student can identify tools without hesitation. Following we present a description of each tool/simulator:

- **Jlógica:**

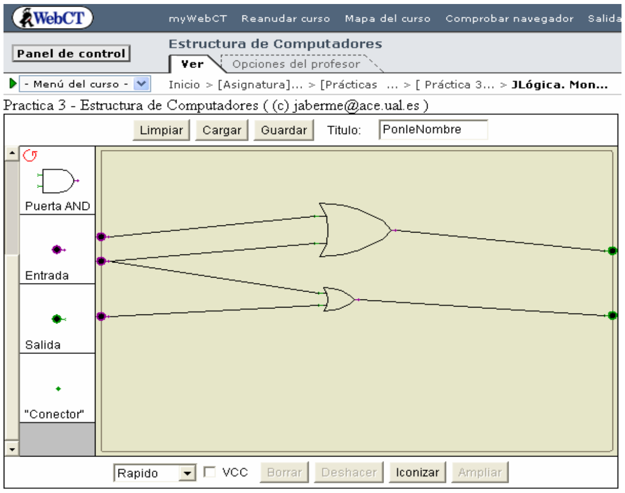
Its main purpose is to simulate complex logical circuits using less complex digital modules [7].

JLogica expects to involve students with the digital circuit simulation, aspect that is promoted with software for simulation based in components and object oriented. Simulation software benefits students in the sense that they have the source code, they can inspect, read, understand, modify and test the code that simulates the circuit's behaviour and every component's behaviour. The possibility of inspecting the code allows the student to get directly involved with the theoretical concepts explained in class. We achieve with this an effective transformation that shifts the role of the student from a passive receiver of information to an active builder of his/hers own knowledge.

With JLogica the student can build complex gates using AND, OR and NOT gates as base building bricks. These complex gates can conform more complex constructions and these can result in a final and operative circuit as an ALU. JLogica also allows to obtain Java code from a correct circuit, this code implements the behaviour of the physical design so performance tests can be ran. In the particular case of Computer Organization, JLogica is used to learn how gates works and how they are assembled together to build functional units of computers as an ALU.

JLogica consist of a basic set of elemental logical gates (AND, NOT, OR) it also has several useful elements as I/O tackles, link tackles see Fig.1, all these pieces are placed in the left hand toolbar. These pieces are the basic building blocks the student has to use to build circuits. Moreover, JLogica has a couple more of toolbars, the former placed on the top of the working area that works with the graphical container of the circuit as a whole, contained in this toolbar we can see options like *Clean*, *Load*, *Save* (option used to save the circuit in a binary file specific to JLogica) and a text area used to give a name for the circuit, this name can be used to identify this circuit in case we decide to iconify it and make it a new gate or building block as part of more complex constructions; and the latter in which we can find options for sets of gates or for the whole circuit, options as *Remove* that removes completely the selected gate, as well as, its connections from the circuit, *Undo*, that reverts recent changes made to the current circuit, iconify that saves current circuit design as a complex gate that from now will appear on the toolbar placed left-hand as an newly available building block that can be reused easily.

Apart from all mentioned options you can find a couple more of options which have to do with simulation issues. These options are the list box in which you can select the speed of propagation inside the circuit and a checkbox named as VCC used to simulate a switch that turns circuit power on or off, it is the power supply.



Practica 3 - Estructura de Computadores ((c) jaberme@ace.ual.es)

Titulo: PonleNombre

Rapido ☐ VCC

Figure 1. Screen shot from JLogica

- **Virtual E-S:**

It is common that subjects related with hardware, have a remarkable degree of difficulty for such students. Having this handicap in mind, we developed a Java based software component that we named CG (Container for Graphics). This tool incorporates an interpreter used to command the CG tool; this interpreter is controlled by a small debugger that avoids client applications from send ambiguous commands to CG. CG accepts descriptions of graphical scenes using the language designed to be used with the interpreter. CG will then compile the "program" (which defines a scene) into a graphical object (scene) to be represented by the CG framework as shown in Figure 2.

CG can be used as graphical support for any practical subject that can be potentially benefited from visual support. In our particular case, we integrated CG as a support framework for Advanced Peripheral Devices. From this integration, a new tool was designed, Visual E/S. This tool is able to represent hardware related scenes. Following we will describe how we used Visual E/S for teaching interactions with the computer's Parallel Port. Now students will face and test the parallel port without dealing with test circuits and hardware (they used to implement a digital counter with resistors, displays, etc; this assembly is not part of the competences defined in the subject's objectives but necessary for testing purposes), students now will have to deal just with the necessary hardware.

Visual E/S is a tool for Advanced Peripheral Devices perfectly integrated in WebCT, near the subject contents (pdfs, teacher’s info, etc). The following line represents an example of how a scene can be built using our language:

```
define puertoparalelo[
 poligono 0,0 -.2,1.5 3.7,1.5  3.5,0  ytranslacion 8
circulo escala .4,.4 xtranslacion -9.0 ytranslacion .5
...
]
```

Scene that will consequently be interpreted by Visual E/S graphical engine (CG) in order to generate a scene like the one shown in Figure 2, that shows the starting point for the specific lab related to Parallel Port. This exercise intends to show how parallel port works, how it behaves and how one should use its communication protocol to send data through it. The simulator is showing the state of the parallel port, the information on the wires connected to its pins and the state of each led from the eight leds display.

It is now up to the students the implementation of a C program able to communicate with the parallel port to show -using the display- a countdown from 9 to 0.

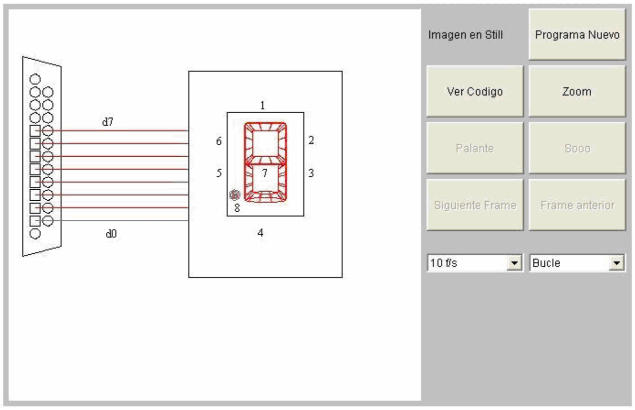


Figure 2. Screen shot from Visual E/S tool.

The display and the parallel port are part from Visual E/S tool, this scene is easily built with the designed language, now the students will center themselves exclusively on the implementation and the basis of the communications protocol. Table III shows the sample program that the student should have to implement to communicate with the display, the code is exactly the same as they had to write when the real test circuit needed to be implemented, with the exception of lines of code numbered as 3,4,7 and 12. Line #3 includes the program that draws the basis of the parallel port using our language, also includes communication functions like the one used in line #7. Line #4 represents the graphical implementation of the display and functions to act on the display like the one used in line #12. Line #7 will configure the file that will register changes that will be later drawn. Line #12, after each call to the function outportb, is the one responsible of transmitting changes made in the state of the parallel port to the simulated parallel port.

TABLE III

```
1.- #include <stdio.h>
2.- #include <dos.h>
3.- #include "puertoParalelo.h"
4.- #include "Display.h"
5.- int main(){
6.-     unsigned char BYTE;
7.-     CG_SetFichero("c:\ficheroEscenas.cg");
8.-     for ( ; ; ) {
9.-         printf("numero :\n");
10.-        scanf("%c",&BYTE);
11.-        outportb(0x378,BYTE);
12.-        CG_actualizaLeds(BYTE);
13.-         getch();}}
```

Table III shows how with, minimal modifications to the legacy lab code, this subject can be provided with a graphical tool converting arid concepts into more attractive stuff to learn.

Visual E/S has an offline counterpart version, that's, there's no need of having an internet connection to work with this lab, because we provide this lab as an offline installable application. So the student can copy this application, write his parallel port code and test it with the simulator in his own computer even if there's no internet available. Visual E/S allows students to send their code and the file containing the simulation with the parallel port described in our graphical language.

Together with these tools, we inserted into WebCT, complementary, third party tools. Some of these tools, like Aritmetica [8], were developed in our same department and others, like ArquiCompu [9], were developed by others universities and released for free. Following a brief description is dropped:

- **ArquiCompu:**

This application suits both experimented subjects. We can select from a wide range of arithmetic algorithms so the application will then illustrate how the computer accomplish the selected arithmetic tasks like pencil and paper algorithms for multiplication, Booth algorithm for fast multiplication based on shifts, division with and without restoration. Numerical conversion operations, such as converting numbers from decimal to IEEE754 format and viceversa. Error detection like Hamming code. Memory hierarchy: cache memory, associative memories. All of them used in Computer Organization and Computer Technology. The application also allows us to analyze the prioritized management of interruptions in cases as chained interruptions and nested interruptions, extensively studied in Advanced Peripheral Devices.

TABLE IV

RELATIONSHIP BETWEEN SUBJECT AND ACTIVITY

Subject	Activity
Computer Organization and Technology	Arithmetic Operations. Numerical Conversions Operations. Error Dectection. Memory hierarchy.
Advanced Peripheral Devices	Chained interruptions and nested interruptions controller.

Table IV represents the classification of proposed activities and the subject where the activity is carried out. Figure 3 shows the simulation of a process in which chained interruption request are to be treated, this is a concept students must deal with in Advances Peripheral Devices.

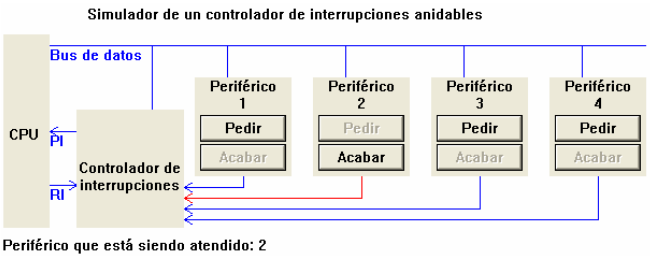

Simulador de un controlador de interrupciones anidables
Bus de datos
CPU
PI
RI
Controlador de interrupciones
Periférico 1
Pedir
Acabar
Periférico 2
Pedir
Acabar
Periférico 3
Pedir
Acabar
Periférico 4
Pedir
Acabar
Periférico que está siendo atendido: 2

Figure 3. Screen shot from ArquiCompu.

4. Evaluation

In this section we present the evaluation of the experience described. This evaluation is based in several aspects. On one side, the results of the surveys from each activity were analyzed, results were grouped by subject. We must note that although activities were different for each subject, all of them had a common factor, this was the use of simulated concepts to enlighten and underline the main ideas and concepts, avoiding the student getting lost with secondary concepts. Surveys were designed to collect student's opinions (see Table V); the survey was available for download. As an additional point of evaluation, the experience of teachers was also added.

Table VI shows a summary of surveys presented to groups of fifty students in obligatory subjects and groups of thirty students in elective subjects.

TABLE V

SUMMARY OF QUESTIONS

P1	Do the contents looks like more attractive to you after this activity?
P2	Was it of any help during your learning stage?
P3	Did you complete the activity independently?
P4	Did you have, could you find, all resources that you needed?
P5	Hours of work employed, aprox. ?
P6	Difficulty degree
P7	Evaluation according the degree of difficulty

As an average, for subjects presented in the survey, when asking about the degree of satisfaction with the new approach adopted for each subject and with the development of all activities (questions P1, P2, P3, P6) answers were in most cases acceptable or even normal (2 points), many of the answers were "enough". Only P3 get a lower consideration. It shouldn't be interpreted negatively because we inquired about the degree of autonomy during the development of the activity. Questions P4 and P5 were designed with a character merely informative.

They only evaluated the availability of resources and the time dedicated to the activity, these answers will be used as a factor of correction for coming experiences.

TABLA VI
SURVEYS' RESULTS GROUPED BY SUBJECTS

Subject		Mean	DT	Median	Mode
Computer Organization	P1	3,25	0,91	3,0	Normal
	P2	3,19	0,76	3,0	Enough
	P3	2,45	0,77	2,0	Yes
	P4	2,27	0,58	2,0	Yes
	P5	2,18	0,65	2,0	>5 h
	P6	3,53	0,75	4,0	Normal
	P7	1,54	0,79	1,0	Agreed
Advanced Peripheral Devices.	P1	2.94	0.80	3.0	Normal
	P2	3.59	0.62	4.0	Enough
	P3	2.90	0.93	2.0	Yes
	P4	1.75	0.62	1.0	Yes
	P5	2.71	0.56	3.0	>5 h
	P6	2.91	0.74	4.0	Enough
	P7	2.48	1.10	3.0	understimated

5. Conclusions and future work.

In this paper we presented results obtained after the experimentation carried out by a working party from Computer Architecture and Technology area, whose main objective is to incorporate active methodologies into our University's Computer Sciences curriculum subjects. Our intention is to call into practice certain methodological reflections described in the EEES convergence process as well as checking the degree of acceptance and effectiveness that these activities achieve between our students.

All experiences were mainly designed to study how applications integrated into the WebCT virtual environment influenced on the learning process. We must underline that results obtained from surveys dropped acceptable results on the set of proposed questions. Positive answers for questions P2 and P3 drive us to believe on the goodness of this kind of methods, such results encourages us to develop, improve and integrate many more tools in WebCT.

Two years of intensive work and study about this kind of experiences gave us as result the conviction that the student's learning process has been benefited.

We are interested in testing for a much longer period of time so we could study the complete evolution of the students in their academic and professional life. Having this as a goal, we can then consider already tested experiences as a starting point from which we can, and will, evolve towards the active methodologies from a practical perspective.

References:

[1] *The European Higher Education Area.* Joint Declaration of the European Ministers of Education. Convened in Bolonia on the 19th of June 1999.

[2] *Realising the European Higher Education Area.* Communiqué of the Conference of Ministers responsible for Higher Education in Berlin on 19 September 2003.

[3] *The European Higher Education Area- Achieving the Goals.* Communiqué of the Conference of Ministers responsible for Higher Education in Bergen on 19 May 2005.

[4] J.Roca, *Periféricos Avanzados*. Universidad de Almería. 2006. ISBN:84-96270-78-5.

[5] *Libro Blanco. Título de Grado en Ingeniería Informática.* Agencia Nacional de Evaluación de la Calidad y Acreditación. Marzo 2005.

[6] J.F. Sanjuán, J.A. Álvarez, E.M. Garzón; J. Roca. A Methodological Experience based on Computer Sciences Curricula under European Higher Education Area. *International Conference on Engineering and Computer Education (ICECE05), ISBN: 84-6098-149-5. 2005.*

[7] J.A. Álvarez, V.J. Ruiz, J.F.Sanjuan, J. Roca, C.A. Bermejo. JLógica, un entorno de diseño de sistemas digitales. *IX Jornadas reenseñanza Universitaria de la Informática (JENUI 2003). Pag. 581-584.*

[8] J.J. Fernández, I. García and E. M. Garzón. Educational Issues on Number Representation on Arithmetic in Computers: An Undergraduate Laboratory. *IEEE Transactions on Education. 2003 V.46. Pag. 477-485.*

[9] José García. ETCII. Investigación, Aplicaciones y Experiencias Concretas. Jornadas UNED 2000. Conocimiento, Método y Tecnologías en la Educación a Distancia.