

Virtual Laboratory Ontology for Engineering Education

Josep Prieto-Blázquez, Ivan García-Torà, Jordi Herrera-Joancomartí and Ana-Elena Guerrero-Roldán
Open University of Catalonia, jprieto@uoc.edu, igarcia@uoc.edu, jherrera@deic.uab.cat, aguerrero@uoc.edu

Abstract - Virtual Laboratories are an indispensable space for developing practical activities in a Virtual Environment. In the field of Computer and Software Engineering different types of practical activities have to be performed in order to obtain basic competences which are impossible to achieve by other means. This paper specifies an ontology for a general virtual laboratory. The proposed ontology provides a mechanism to select the best resources needed in a Virtual Laboratory once a specific practical activity has been defined and the main competences that students have to achieve in the learning process have been fixed. Furthermore, the proposed ontology can be used to develop an automatic and wizard tool that creates a Moodle Classroom using the practical activity specification and the related competences.

Index Terms – Ontology, OWL, Practical activities, Protégé, Virtual Laboratory.

INTRODUCTION

Quick and continuous developments in information and communication technology, together with cultural changes, have increased the consciousness and the demand for distance learning that would allow the students to access their course at anytime and anywhere.

Virtual Distance Learning and Blended Distance Learning are two modalities of distance education that make an intensive use of information and communication technologies through Internet, where asynchronous communication is allowed in space and in time between the students and the academic staff. Thus, students have maximum flexibility, adapting their studies to their own rhythm, and surroundings at any specific time [1], thereby consolidating a growing, dynamic online community [2].

According to computer curricula 2005 [3] and a review from ICECE05[4], practical laboratory activities are an essential part of any computer curriculum since they strengthen concepts presented during lectures. For that reason, new virtual spaces are required in a Virtual University so practical activities can be carried out. Such spaces are called Virtual Laboratories (VLabs) and should include different kind of activities ranging from designing and implementing solutions, to testing and documenting several applications, systems, devices, and processes.

The concept VLab is defined in different ways in the literature. Some authors [5] take a simple vision of a Virtual Lab as a local computer hosting that may include some simulation capabilities [6]-[7]. An accurate definition can be found in [8] where the term VLab adds other information

technologies to create a highly interactive virtual environment tailored to the needs of students and teachers. Other authors introduce other important aspects like pedagogical and academic factors into a Virtual Lab [9]-[10]. In [11] a general structure for a VLab is proposed. Taking into account these previous definitions, this paper defines a Virtual Laboratory (VLab) as an interactive virtual space that incorporates all the technological, pedagogic and human resources for carrying out practical activities, adapted to the needs of the students and teachers in a virtual learning environment.

The main contribution of this paper is the representation using a data model based on Ontologies of a general structure for a VLab. Such an approach has two main advantages. On one hand, the ontology provides a mechanism to select the best resources needed for a VLab, and on the other hand, such semantic representation allows us to develop an automatic and wizard tool that creates a Moodle Classroom for teaching and learning based on practical activity specification and its related competences.

The paper is organized as follows. Firstly, details of the general structure for VLabs are presented. Secondly, an Ontology for VLab is proposed. Next, a possible implementation through Moodle is described. Finally, a brief conclusion closes the paper.

VIRTUAL LAB STRUCTURE

This section provides a brief summary of the general VLab structure presented in [11] since the Virtual Laboratory Ontology specified in this paper is based on such structure. The proposal defines and classifies the general resources needed in a VLab for engineering education. Next, such resources are briefly classified and described:

- **Technological resources.**
 - A. Virtual communication environment (VCE)
 - B. Simulator (SIM)
 - C. Remote laboratory (REM)
 - D. Automatic assessment tool (AAT)
 - E. Virtual machine (VRM)
 - F. Software support (SWS)
- **Pedagogic and strategic resources.**
 - G. Learning methodology (MET)
 - H. Evaluation (AVA)
 - I. Support documentation (DOC)
- **Academic staff resources.**
 - J. Student (STU)
 - K. Teacher (TCH)

Technological resources are next described:

The **VCE** is a resource that allows communication between the users of the VLab: students and teachers. VCE can be implemented using shared mailboxes, blogs, wiki, videoconference, public forums and virtual blackboard from different applications, e.g., *Blackboard*, *BSCW*, *WebCT* or *Moodle Classroom*.

SIM is a tool that imitates experiments, states or processes. One of the main features of simulators is their interactive capability. Interactive simulations are quickly gaining importance as a means to explore, comprehend and communicate complex ideas [12]-[13]. Interactive simulators can be implemented using a combination of computers, high-resolution graphics, simulation programming languages, and an Internet connection.

A **REM** provides the student with a real environment to perform practical activities. It consists of computers and/or devices set up to allow remote access [14]-[15].

The **AAT** enables interactive processes to take place between students and the automatic assessment system that allows students to learn about iteration processes. AAT process consists of the compilation and execution against test data and user-specified data (benchmark).

A **VRM** is a tool that lets users create separate environments, each of which emulates the hardware of a complete physical computer. Students and teachers can run multiple operating systems at once on a single physical computer and switch between them. Virtual machines can be found implemented in different software solutions. The most important ones are VMWARE, Xen virtual machine monitor and Microsoft Virtual PC.

The **SWS** are specific programs or applications that run in the local computer. For instance: compilers, emulators, editors and offices programs.

Regarding the pedagogical and strategic resources the following descriptions are provided:

The **MET** is a pedagogical tool to improve the learning process of the students. There are several proposals of learning methodology from a teacher-lecture-centered environment to student-centered [16]. Practical activities of laboratory require the last one.

The **AVA** is a pedagogical resource that allows the students to achieve their learning objectives. In a Virtual University, the system evaluation should be flexible and based on continuous evaluation [17].

The **DOCs** are all information that will help students to achieve the objectives of the practical activities.

Finally, the academic staff resources include:

The **STU** is the person enrolled at a university who studies a particular subject.

The **TCH** of a Virtual Lab is a member of the academic staff who helps students reach their individual objectives, offering each student personalized attention.

VIRTUAL LAB ONTOLOGY

The general VLab structure presented in [11] and described in the previous section can be represented in different ways. However, an ontology representation of such information provides a powerful tool to retrieve and interact with the data contained.

The most used definition of an ontology in the field of engineering is of Gruber [18]. He defines an ontology as the explicit specification of a conceptualization, a conceptualization being an abstract and simplified view of the concepts and relationships it wants to represent. Such representation allows an easy way of sharing and reusing the knowledge contained in.

Ontologies can be specified using different languages, but in recent years two of them, DAML+OIL [19] and OWL [20], focus the main attention since both include support for representing instances. Such instantiation allows to convert, in practical terms, ontologies into knowledge bases.

Web Ontology Language (OWL) was carried by the W3C to formalize Ontologies on the Web. The virtual lab ontology presented in this paper has been implemented using OWL since, with regard to other languages that have been established to represent knowledge, OWL adds more vocabulary for describing properties and classes: among others, relations between classes, cardinality, equality, richer typing of properties, characteristics of properties, and enumerated classes.

More precisely, the virtual lab ontology has been developed using Protégé [21], a platform environment for creating and editing Ontologies and knowledge bases. Protégé is being developed at Stanford University, based on Java and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development. Protégé is one of the best tools used to model Ontologies and the most important thing over its competitors are its specifications and the large support of things like the OWL-full language, while other programs only support RDF, OWL-lite.

The version of the program used was Protégé-3.3.1, the last table revision at the beginning of this project. OWL represents the knowledge through a hierarchy of classes. All classes start at a meta root class called *owl:thing* and all classes inherit the properties of its "superclass".

Like any object-oriented programming language, a class cannot be used by itself, but instead one or more instances are created. Furthermore, each class has its own properties and there are two types:

- *Object properties*, used to link the classes and make relations between them.

- *Data type properties*, used to store individual data.

I. Ontology specification

The structure of the virtual laboratory ontology is depicted in Figure 1. The main class of our ontology is “*VLabResources*” which includes all resources needed for any practical activity in an engineering education program.

The information contained in the “*VLabResources*” class together with other specific information of the field of knowledge, provided in the classes “*Subject*”, “*Competence*”, “*PracticalActivity*” and “*Task*”, allow to define the necessary resources of the Virtual Lab to perform the practical activities in a virtual learning environment.

Notice that, the specification of the classes “*Subject*”, “*Competence*”, “*PracticalActivity*” and “*Task*” providing

input information of the field of knowledge is intentionally left open because is out of the scope of the paper. However, a proposed ontology that details any of this information more precisely can be used to enhance our system (see [22] for an example of competence ontology).

Figure 2 represents the relations between these classes using object properties. The subjects are assigned to teachers or other academic staff members. Every subject has competences assigned, and every competence has practical activities required to complete and develop those competences. Finally, a practical activity is formed by one or more particular tasks, and each one requires one or more resources from the VLab structure to be completed.



FIGURE 1
VLAB ONTOLOGY STRUCTURE (“VLAB.OWL”)

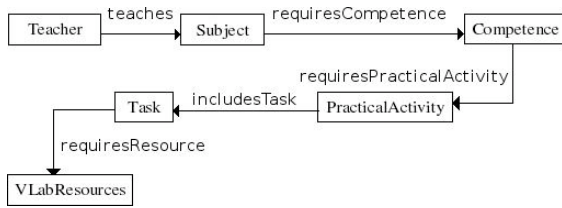


FIGURE 2
VIRTUAL LAB CLASSES

The correctness of the proposed virtual lab ontology in terms of computability and semantic correction has been validated using standard reasoner tools like Pellet [23] or FaCT++ [24]. Such validation is especially important when connecting the proposed ontology with other ones that may enhance the system (for instance, a more elaborated ontology defining competences or subjects).

II. Example: Forum Class

Due to the paper length constraints, only the *Forum* class is examined in order to illustrate one particular example of the implementation details of the virtual lab ontology.

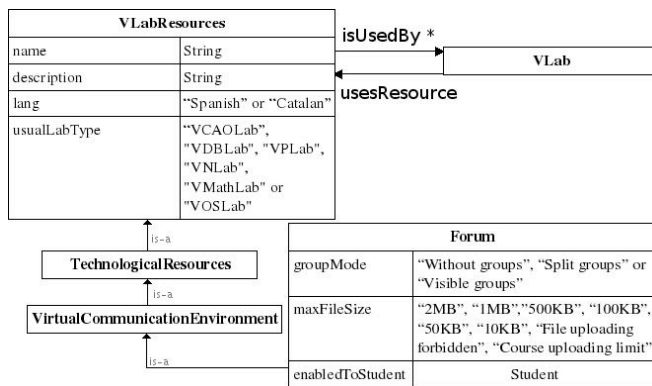


FIGURE 3
FORUM CLASS

The forum class is a subclass of *VLabResources* and, for that reason, inherits its properties (see Figure 3):

- A data type property called *name*. A string that can be only assigned once.
- A data type property called *description*. A string that can be only assigned once.
- A data type property called *usualLabType*. It can only be one of *VCAOLab* (Virtual Computing Architecture and Organization Lab), *VDBLab* (Virtual Database Lab), *VPLab* (Virtual Programming Lab), *VNLab* (Virtual Networking Lab), *VMathLab* (Virtual Mathematic Lab) or *VOSLab* (Virtual Operating System Lab).
- An inverse object property *isUsedBy*. Its range is the *VLab* class and its inverse property is *usesResource*.

Forum properties:

- A data type property called *groupMode*. It can be only one of *Without groups*, *Split groups* or *Visible groups*.
- A data type property called *maxFileSize* that can be only one of *2MB*, *1MB*, *500KB*, *100KB*, *50KB*, *10KB*, *File uploading forbidden* or *Course uploading limit*.
- An object property *enabledToStudent*. Its range is the *Student* class. This object property can be assigned as many times as students are assigned to the forum.

Note that inverse object properties are used to query data from another point of view (i.e., *VLab usesResource Student* or *Student isUsedBy VLab*, where *isUsedBy* is the inverse property of *usesResource*).

All the object properties have their own inverse object property to allow easier SPARQL queries.

The example of the XML/RDF OWL code for this class is shown below:

```

<owl:Class rdf:about="#Forum">
  <owl:disjointWith>
    <owl:Class rdf:about="#TeacherBoard"/>
  </owl:disjointWith>
  [ more disjoints are omitted ]
  <rdfs:subClassOf>
    <owl:Class rdf:about="#VirtualCommunicationEnvironment"/>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">
        >1</owl:cardinality>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="maxFileSize"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:cardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#int">
        >1</owl:cardinality>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:ID="groupMode"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
  
```

VLAB IMPLEMENTATION

As it has been stated before, the proposed virtual lab ontology allows an automatic generation of a VLab Moodle Classroom based on specific input data from the subject, competences, practical activities and tasks (see Figure 2). Such automatic generation is briefly summarized in this section.

I. Client application side

From the teacher's point of view, the virtual lab generation can be described in six steps. Figure 4 shows the different windows that the proposed system displays at every step.

- **Step 1:** The teacher selects a subject from the list. Then he selects which competences should be acquired.

- **Step 2:** The system shows a list of tasks required by the previously selected competences. Each task is formed by practical activities that, again, can be selected or deselected by the teacher.
- **Step 3:** The teacher provides some information like the course name, summary and some optional descriptions to be displayed or not.
- **Step 4:** A screen with the needed tools, grouped by the task and classified by type, is shown. By default, all the tools and the most reasonable type of VLab are checked.
- **Step 5:** The ontology is updated and the Moodle classroom with the assigned resources is created. The screen displays details about the process and ends with a link to Moodle classroom page.
- **Step 6:** The Moodle classroom page is active. It includes all the needed tools and properties that the teacher decided to make available.

obtain all the displayed information. The ARC2 classes library is used to import the ontology file (called "VLab.owl"), serialize it to a MySQL database and then perform the queries.

- On step 6 some modified versions of Moodle scripts generate a Moodle Classroom and its tools by simply calling a function for each one. Using this approach the system can use the same configuration and database as the original Moodle does. Once the tools are created, ARC2 is used again to update an OWL file called "ontology-importer.owl". This operation is done by inserting the new VLab properties using SPARQL+ and exporting the ontology in RDF/XML (OWL) format over the original file.

CONCLUSIONS

1. Server application side

Regarding the server side, the process previously described performs the following operations:

- From step 1 to step 5 the PHP scripts of the server, using SPARQL query language and the VLab ontology,

Virtual laboratories are virtual spaces where practical activities can be carried out. Depending on the activity to perform and the competences students have to achieve, a different virtual laboratory has to be designed providing appropriate resources.

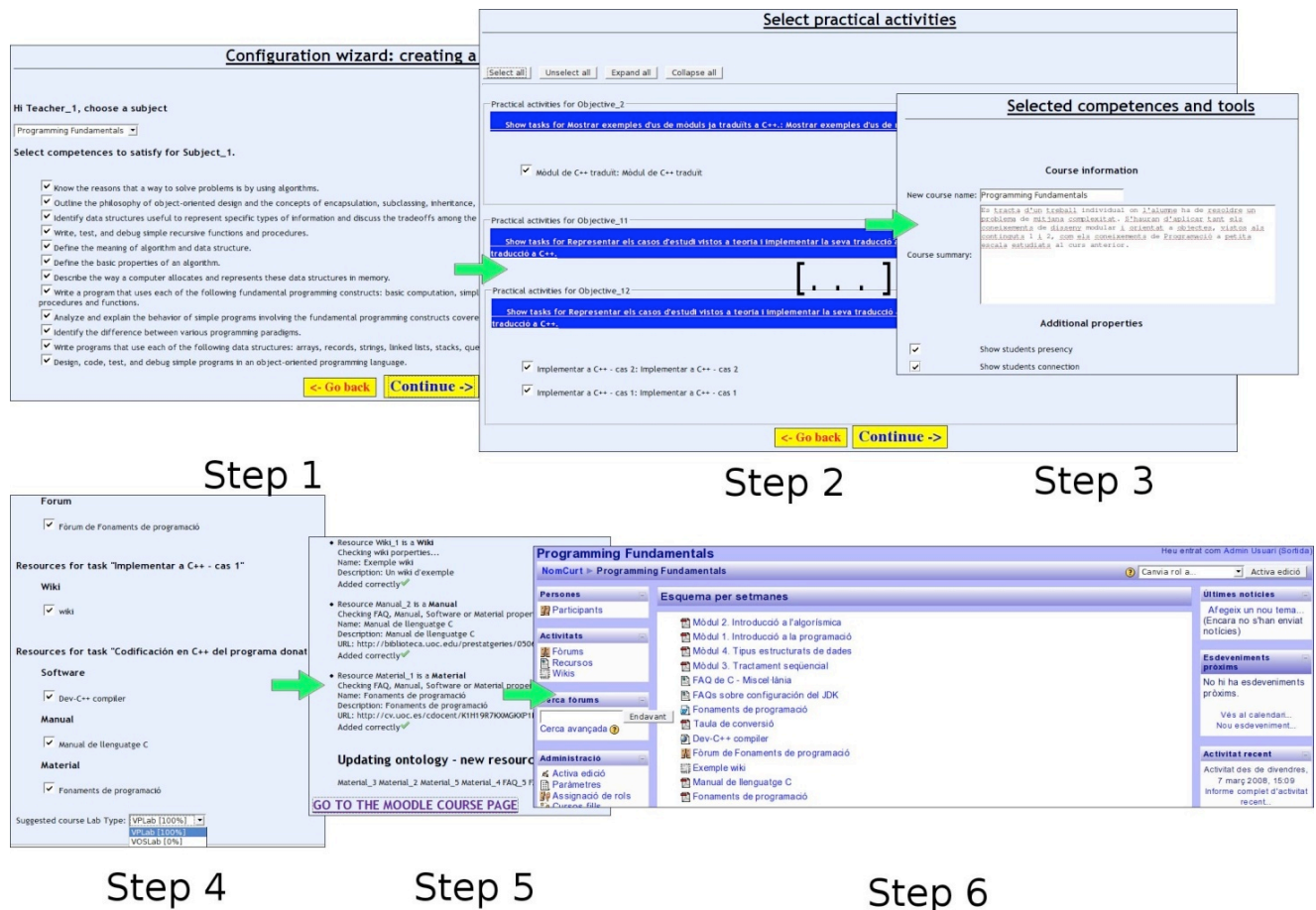


FIGURE 4
VLAB IMPLEMENTATION STEPS

In this paper a virtual laboratory ontology has been proposed. The ontology describes different resources a virtual lab has to include. It also specifies the relations between such resources, the characteristics of the practical tasks that have to be developed, and the competences students have to achieve. The paper also presents a system that uses the proposed ontology to design and construct a virtual lab. The teacher provides information about the specific tasks and competences of practical activities and the system creates a virtual lab as a Moodle classroom with all the appropriated resources.

ACKNOWLEDGMENT

This work was partially supported by the Spanish MEC and the FEDER funds under grant TSI2007-65406-C03-03 "E-AEGIS", TIN2006-15107-C02-01 "PERSONAL" and CONSOLIDER CSD2007-00004 "ARES", funded by the Spanish Ministry of Science and Education.

REFERENCES

- [1] Simonson, M., R., "Teaching and Learning at a Distance". Upper Saddle River, NJ: *Merrill Prentice Hall*, 2000.
- [2] Keegan, D.; Holmberg, B; Sewart, D., "Distance Education International Perspectives", NY: *St. Martin's Press*, 1988.
- [3] I. J. T. F. on Computing Curricula, "Curriculum guidelines for undergraduate degree programs in computer engineering", *IEEE Computer Society Press and ACM Press*, December 2004.
- [4] Tovar, E.; Castro, M., "Building common spaces in engineering education: A review from icece05", *IEEE Transactions on Education*, vol. 50, 2007, pp. 79–84.
- [5] Chiu, K., "What are the benefits of a virtual laboratory for student learning?", *HERDSA Annual International conference*, Melbourne, Jul. 1999, pp. 12–15.
- [6] Harms, U., "Virtual and remote labs in physics education", *Second European Conference on Physics Teaching in Engineering Education*, Budapest, Jun. 2000.
- [7] Leitner, L., J.; Cane, J., W., "A virtual laboratory environment for online it education", in *SIGITE '05: Proceedings of the 6th conference on Information technology education*. New York, NY, USA: ACM Press, 2005, pp. 283–289.
- [8] Noor, A., K.; Wasfy, T., M., "Simulation of physical experiments in immersive virtual environments", *Engineering Computations: Int. J. for Computer-aided Engineering and Software*, vol. 18, no. 3-4, 2001, pp. 515–538.
- [9] Levert, C.; Pierre, S., "Designing distributed virtual laboratories: Methodological and telecommunications aspects", *International Journal on E-Learning*, vol. 2, no. 3, 2003, pp. 18–28.
- [10] Rak, A., M.; Godziemba-Maliszewski, M., "A proposal of virtual laboratory structure", *Proceedings of The 23rd IEEE in Instrumentation and Measurement Technology Conference*, Apr. 2006, pp. 847–850.
- [11] Prieto-Blázquez, J.; Arnedo-Moreno, J.; Herrera-Joancomartí, J., "An Integrated Structure for a Virtual Networking Laboratory", *IEEE Transactions on Industrial Electronics*, vol.55, no. 6, in print. 2008.
- [12] Repenning, A.; Ioannidou, J., "Collaborative use and design of interactive simulations," in *CSCL Proceedings, Stanford, CA*, 1999.
- [13] Corter, J., E.; Nickerson, J., V.; Esche, S., K.; Chassapis, C., "Remote versus hands-on labs: a comparative study", *FIE'04: 34th Annual Conference on Frontiers in Education*, vol. 2, Oct 2004, pp. F1G:17–21.
- [14] Sicker, D., C.; Lookabaugh, T.; Santos, J.; Barnes, F., "Assessing the effectiveness of remote networking laboratories", *FIE'05: 35th Annual Conference on Frontiers in Education*, Oct 2005, pp. S3F–7; S3F–12.
- [15] Ma, J.; Nickerson, J., V., "Hands-on, simulated, and remote laboratories: A comparative literature review", *ACM Comput. Surv.*, vol. 38, no. 3, 2006, pp. 7–10.
- [16] Brush, T.; Soye, J., "Implementation and evaluation of a student centered learning unit: A case study", *Journal Educational Technology Research and Development*, vol. 48, no. 3, Sept. 2000, pp. 79–100.
- [17] Kaczmarczyk, L., C., "Accreditation and student assessment in distance education: why we all need to pay attention," in *ITiCSE '01: Proceedings of the 6th annual conference on Innovation and technology in computer science*. New York, NY, USA: ACM Press, 2001, pp. 113–116.
- [18] Gruber, T., R., "A translation approach to portable ontologies", *Knowledge Acquisition*, vol. 5, no. 2, 1993, pp. 199–220.
- [19] Connolly, D.; Harmelen, F.; et al., "DAML+OIL Reference Description", *W3C*, <http://www.w3.org/TR/daml+oil-reference>, [Accessed March 2008].
- [20] Bechhofer, S.; Harmelen, F.; et al., "OWL Web Ontology Language Reference", *W3C*, <http://www.w3.org/TR/owl-ref/>, [Accessed March 2008].
- [21] PROTÉGÉ, "Protégé website", <http://protege.stanford.edu/>, [Accessed March 2008].
- [22] Sicilia, M., A., "Ontology-based approach to formalization of competencies", *Competencies in Organizational E-learning: Concepts and Tools*. IGI Global, November 2007, pp. 185-206.
- [23] PELLET, "Pellet web site", <http://pellet.owldl.com/>, [Accessed March 2008].
- [24] FACT++, "Fact++ web site", <http://owl.man.ac.uk/factplusplus/>, [Accessed March 2008].

AUTHOR INFORMATION

Josep Prieto-Blázquez, Teacher and program director of Computer Engineering. Open University of Catalonia, jprieto@uoc.edu

Ivan García-Torà, Technical programmer, specialist in OWL and Protégé. Open University of Catalonia, igarcia@uoc.edu

Jordi Herrera-Joancomartí, Teacher of Computer Science. Autonomous University of Barcelona, jherrera@deic.uab.cat

Anna Guerrero-Roldán, Teacher of Computer Science. Open University of Catalonia, aguerreror@uoc.edu