

# Common Virtualized Environment in Telematics Labs

<http://dx.doi.org/10.3991/ijoe.v9iS5.2757>

F.J. Fernández, A.J. Sierra, T. Ariza, G. Madinabeitia, J.M. Vozmediano  
Universidad de Sevilla, Sevilla, Spain

**Abstract**—The aim of this paper is to show the common scenario for practical content of subjects in the Department of Telematics Engineering by means of virtualization. This Department is responsible for a high number of laboratory groups, requiring many teachers and a high workload. A common environment for all practice groups is desired. A virtual machine was built with the same environment of the Computing Centre at School of Engineering at University of Seville. Then, this virtual machine was provided to students and is used to do practices. This article describes this experience and shows the improvements obtained.

**Index Terms**—Virtualization; Telematics labs; autonomous learning; Project-based learning.

## I. INTRODUCTION

The European Higher Education Area (EHEA) [1] is an ambitious and complex plan launched to promote European convergence in education. The effort to establishment of the new European area requires new tools to help both teacher and student, mainly in practical teachings. The University of Seville [2] and the Department of Telematics Engineering (DTE) are committed to implement this plan [3].

The high number of laboratory groups, resulting in high teaching load and a large number of teachers involved, requires a common support for all practice groups. This allows consistency in the assessment of all students. A common framework for development of projects and practices in different subjects has benefits. One the most important benefits are that students can optimize learning time to progress in specific content.

Virtualization techniques allow the establishment of flexible, versatile and low-cost laboratories to teach practical subjects of telematics. This paper shows the common virtualized platform for teaching practical subjects used by the Department of Telematics Engineering (DTE). This Department is responsible for a high number of laboratory groups in Degree in Telecommunication Technology Engineering (DTTE) and Telecommunications Engineering (TE) at University of Seville, requiring many teachers and a high workload. Thus, a common environment for all practice groups is required. The virtualized environment is used not only in the shared Computing Centre of the School of Engineering at University of Seville, but also in “electrified” classroom.

To implement this proposal, a virtual machine has been created with a similar environment to the Computing Centre. This virtual machine allows the easy deployment of a Project-Based Learning (PBL) [4] methodology.

According to this well-known method of learning the student must develop a project of some magnitude. The teacher will provide guidance and the necessary tools for the student. The students will analyze, organize, develop and test the software project, thus they will plan and organize their own work. Using the virtual machine the student can carry out their work in an environment similar to the Computing Centre.

Furthermore, the virtual machine allows that new tools can be added into this environment, such as TVSP [5], [6] (*Tool for Validation Software Project*). This tool allows validating the student software project in the Fundamental of Programming II subject in DTTE, to help the student to develop its Project.

In this paper, first we introduce a background in virtualization techniques. After, we describe experiences related to subjects at Department of Telematics Engineering. Next, we present a use case in a specific subject in DTE. Finally, we show results and conclusions.

## II. BACKGROUND

### A. Virtualization

**Virtualization**, in this context, is construction of a software version (called *virtual machine*), of an existing resource (called *real machine*). Among different types of existing Virtualization (Hardware, Desktop, Software, Memory, Storage, Data and Network), hardware virtualization is focused on creating a virtual machine that acts like a real computer with an operating system.

Software running on the virtual machine is separated of underlying hardware resources. Thus, a computer that is running Microsoft Windows can host a virtual machine that works like a computer with the Ubuntu Linux operating system.

We highlight 3 types of virtualization [7]:

- a) **Virtualization with Type 1 hypervisor:** Virtualization software runs directly on the hardware performing a simulation of real hardware. The operating system of the virtual machine is running on virtualization software and does not require any modification. Different operating systems can be executed simultaneously. Fig. 1 shows the structure of a system with this kind of virtualization.
- b) **Virtualization with Type 2 hypervisor:** Virtualization software runs on the host operating system as an application within the machine. The guest operating system that runs on top of virtualization software usually requires modifications to run in this virtual environment. Fig. 2 shows the structure of a system with this kind of virtualization.

c) **Paravirtualization:** In this case, the virtual machine OS is modified to allow calling virtualization software functions. The OS uses Virtual Machine Interface (VMI) for this call. The Fig. 3 shows structure of a system with paravirtualization. VM1 uses paravirtualization and VM2 uses a type 1 hypervisor.

*B. Virtual machines*

A virtual machine is therefore a software program that emulates all the hardware components of a computer and allows running other software programs making them believe that are actually running on the emulated hardware. The emulated hardware can be:

- a) Motherboard including the BIOS.
- b) CPUs
- c) Memory
- d) Hard Drives, CD / DVD, floppy.
- e) Network Adapters
- f) USB Drivers
- g) Sound card.
- h) Etc.

Sometimes you can combine real hardware with emulated hardware in the same virtual machine. For example, a virtual machine can use a physical hard drive, USB devices, etc.

Creating a virtual machine consists of choosing the hardware to be simulated. In Fig. 4, the settings of the virtual machine created using VMware Player can be seen.

Then the virtual machine can be started and the operating system and application software can be installed, in the same way as a physical machine.

The hardware features of the real machine can be modified once created the machine. It is possible also add or remove devices. Therefore if more storage is needed, a new hard drive could be added later, the memory size could be increased, new network interfaces may be added, etc.

It should also be noted that there are tools that allow making virtual clones of physical machines on site. Thus, a physical machine can be used as a base for creating a virtual machine.

Virtualization software also allows creating virtual networks. To do this, they are able to emulate switches, routers and networks that interconnect the various virtual machines among them, with the host machine and the Internet. Fig. 5 shows the network configuration tool of the VMware software.

*C. Related works*

Different proposals are aimed to hands-on labs remotely with Internet technologies, especially in distance education.

So, Hu et al. [8] have designed the Tele-lab. The Tele-lab server builds a virtual security laboratory using lightweight virtual machines and features a pure web user interface. Thus, students can easily exercise security over the Internet.

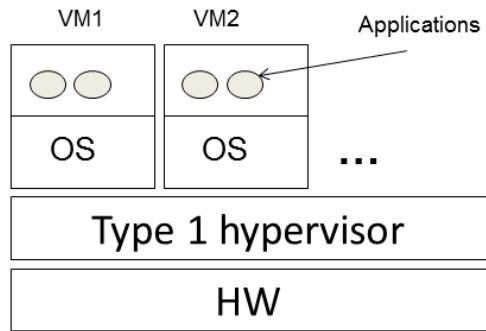


Figure 1. System architecture with type 1 Hypervisor.

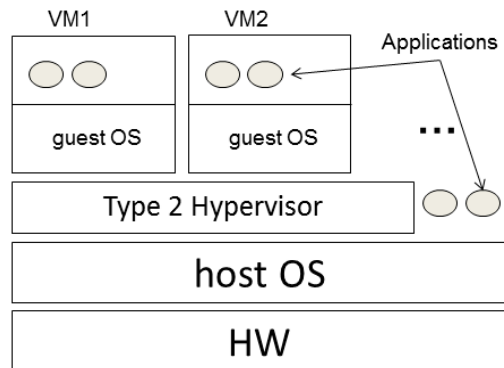


Figure 2. System architecture with type 2 Hypervisor.

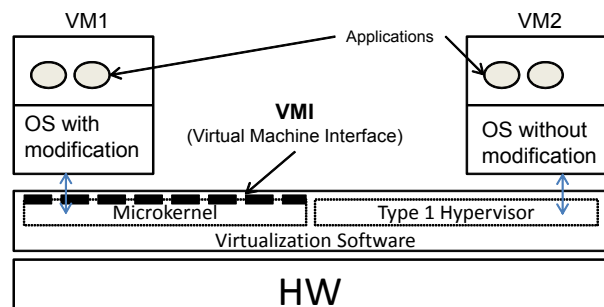


Figure 3. System architecture with Paravirtualization.

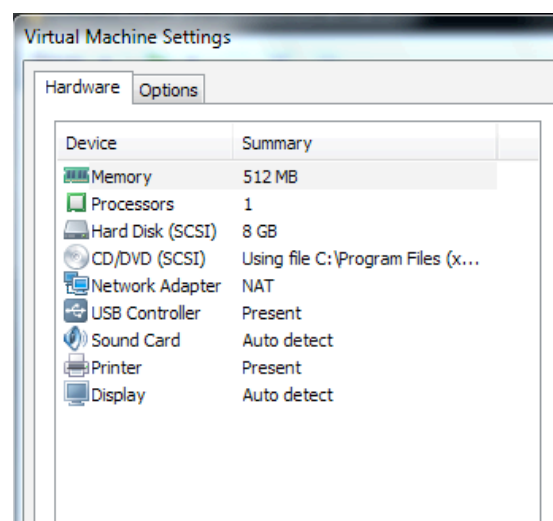


Figure 4. Virtual machine settings.

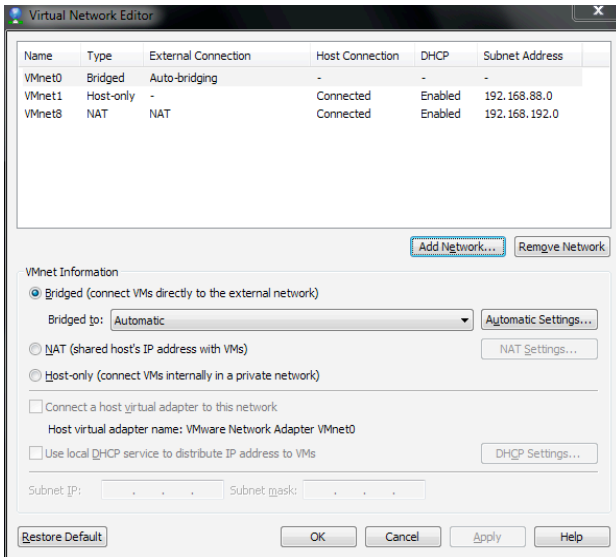


Figure 5. Virtual network configuration tool.

Li et al. [9] performed something different. In this case, Virtualization technology enables multiple virtual machines and their applications to run simultaneously on a single physical computer. This eliminates the need to have multiple physical machines with diverse operating systems, typically deployed in remote network security labs.

Tao et al. [10] show a work that shares their experience of using VMware virtual machines in supporting hands-on web security education, and developing multiple virtual web security lab modules based on the virtual machines, in the lab modules are part of their NSF SWEET (Secure Web dEvelopment Teaching) project.

Ros et al. [11] show a virtual machine-based framework and an automatic evaluation system implemented, in the context of a distance university, in a course that is taught entirely online.

Bandi et al [12] report the development of a “Virtual Laboratory for Systems and Software” that teaches the basics of Linux. Their goal of that laboratory is to give the students a platform through which they can learn the essentials of Linux. The student interacts with a virtual instance of a Linux machine through the browser on his machine to learn basic tasks of Linux system administration.

### III. DESCRIPTION OF THE EXPERIENCE

The Department of Telematics Engineering provides a virtual machine to support the practice contents of most of its subjects. These subjects belong to Degree in Telecommunication Technology Engineering (DTTE) and Telecommunication Engineering (TE). The virtual machine provides the same environment that students use in the Computing Centre (CC) of the School of Engineering to accomplish laboratory projects and practices, using Linux OS.

Professors of the Department who are responsible for subjects with practical work decided the type of virtual machine and the software which had to be included so that students could do their practices. The result was a virtual machine with a minimal Linux OS where all unnecessary software has been removed. Attempt was made to make

the machine as light as possible and that the requirements of the host machine hardware were minimal, so that any current computer could run it.

All installed software is free, so that the problems which could arise with software licenses have been avoided. Therefore, each student can have an unlimited number of virtual machines. Furthermore, the virtual machine is easily expandable and easily allows providing any other software that students may need.

The virtual machine is provided for two virtualization products:

- VMware Player [13]
- VirtualBox [14]

The virtualization software, where the virtual machine runs, is virtualization software that uses a type 2 hypervisor, so students can keep their current operating system with little modification. Virtualization software is installed as a program, and students do not need to configure anything to start using a virtual machine. Moreover, as all the software support is installed and configured on the virtual machine, the student takes very little time to start working, focusing on the content of the subjects.

Afterwards, this virtual machine has been cloned to computers used in computer labs. The students have the same experience working in the lab or in their own computers. Having the same environment reduces the doubts arisen by the students about it. This avoids having to waste time on side issues not related to subjects, and reduces the time for the maintenance of equipment.

The instructions to download and install the virtualization software and the virtual machine are provided using the e-learning platform (Blackboard Learning System [15]). These instructions are available in every subject supported by the virtual machine.

These instructions are organized in three phases. This section describes the steps for installing, configuring and running the virtual machine on Windows OS. Similar steps can be applied to any other operating system. The phases are as follows:

#### **Phase 1: File Download.**

Download the chosen virtualization software. The following two virtualization programs are supported:

- VMware Player
- VirtualBox

Download both, compression software 7-zip, and the compressed virtual machine.

#### **Phase 2: Software installation.**

The chosen virtualization software must be installed, VMware Player or VirtualBox, and also the 7zip compression software.

Then the downloaded virtual machine must be decompressed. Decompression can be done with other programs such as WinRAR or WinZip, that also support the 7z format, and they represent an alternative to the 7-Zip program.

#### **Phase 3a: Running the virtual machine (VMware Player).**

To run the virtual machine, double click on the file with extension “vmx”, which has been decompressed.

#### **Phase 3b: Running the virtual machine (VirtualBox).**

SPECIAL FOCUS PAPER  
COMMON VIRTUALIZED ENVIRONMENT IN TELEMATICS LABS

To run the virtual machine in VirtualBox, first create a configuration compatible with VirtualBox. To do this, once VirtualBox is running in the "Machine" menu, select the item "New ...". Steps are followed until the "Boot disk" is requested. At this point, select "Use existing hard disk" and select the file with extension "vmdk" that was decompressed in the previous step. Then continue on to complete the process.

To turn on the virtual machine just press the "Start" button.

We emphasize that these instructions are the same in all subjects of the Department. The student is able to autonomously install the virtualization software and begin using the virtual machine with these instructions.

Once the student has the virtual machine installed on its computer, he or she can develop the practical content of the subjects as if he or she were in the Computing Centre of the Engineering school. This minimizes the time that the student consumes in the Computing Centre and resources are used more efficiently.

Though the final grading will be carried out at the Computing Centre, most of the development work by the student can be done using the virtual machine provided by the Department.

Additional instructions are provided in specific subjects when it is necessary to install software on this virtual machine. So, this scheme is flexible and scalable.

#### IV. SUBJECT INVOLVED

Since academic year 2011-2012 this virtual machine has been used in several subjects taught by the Department of Telematics Engineering (DTE). We plan to use this virtual machine in new subjects from now on.

In the 2010-2011 academic year, the Degree in Telecommunication Technology Engineering (DTTE) at the School of Engineering at the University of Seville started [16], [17]. The syllabus is divided into 240 credits, spread over four years, at the rate of 60 credits per year, as follows:

- 162 credits for common subjects (basic or specific telecommunication branch),
- 57 credits for specialty subjects, that they are offered in four specialties:
  1. Telematics
  2. Electronic Systems,
  3. Sound & Vision, and
  4. Telecommunication Systems,
- 9 credits for electives and
- 12 credits corresponding to the degree work.

The involvement of Department of Telematics Engineering in the delivery of the syllabus is as follows:

1. 36 credits of core subjects, according to the following distribution:
  - 12 credits in first-year (six in the first quarter and six in the second),
  - 18 credits in the second year (six in the first quarter and twelve in the second) and
  - 6 credits in third year (first quarter).
2. 48 specialty core subject, divided as follows:

- 24 credits in third year (4.5 credits in the first quarter and 19.5 credits in the second) and,
  - 24 credits in fourth year (first semester).
3. 13.5 credits in specialty optional subjects, in the second semester of the fourth year.
  4. 4.5 credits in optional subjects in another specialty, in the second semester of the fourth year.
  5. 4.5 credits in optional subjects offered by the school, in the second semester of the fourth year.

One of the requirements for the design of the syllabus is to strengthen as much as possible the practical nature of teaching and this requirement has been intensively conducted in the subjects taught by the department. So much so, that of the 20 subjects assigned to the department, only one of them is of no practical content.

The Table 1 shows the involved subjects. For each subject, we specify its name, total credits, lab credits, course, and the establishment year for (European Higher Education Area) EHEA. In this table we also indicate current subjects where the virtual machine is being used and the future establishment subjects.

TABLE I. SUBJECT USING THE VIRTUAL MACHINE

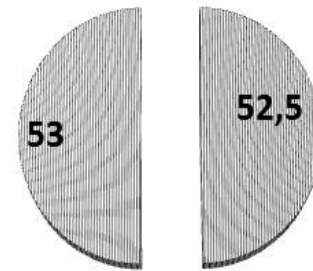
	Subject	Credits	Lab	Course	Year
Current subjects	Fundamentals of Programming I	6	3	1	2010/11
	Fundamentals of Programming II	6	4,5	1	2010/11
	Fundamentals of Internet	6	1,5	2	2011/12
	Structure and protocols of public networks	6	0	2	2011/12
	Fundamentals and Applications of Telematics Services	6	4,5	2	2011/12
	Multiservice Networks	6	1,5	3	2012/13
	Operating Systems	4,5	1,5	3	2012/13
	Software Engineering	6	4,5	3	2012/13
Future establishment	Security	4,5	1,5	3	2012/13
	Teletraffic	4,5	1,5	3	2012/13
	Network management	4,5	1,5	3	2012/13
	Network planning&simulation	4,5	3	4	2013/14
	Advanced network architecture	4,5	3	4	2013/14
	Advanced Telematics Services	4,5	3	4	2013/14
	Database design	4,5	3	4	2013/14
	Telematics project	4,5	3	4	2013/14
	Distributed Systems	4,5	3	4	2013/14
	Telematics System Management	4,5	3	4	2013/14
Mobile Application Design	4,5	3	4	2013/14	
Interactive Application Design	4,5	3	4	2013/14	

It can be seen that for a total of 105.5 credits taught by the department, 52.5 credits are practical, which is about half the teaching load (a 49.76% of the total, exactly), as we can see in the Fig. 6.

The descriptors in the different subjects taught by the department are described below:

1. Fundamentals of Programming I: fundamentals in structured programming, algorithms and the use of an operating system.
2. Fundamentals of Programming II: Dynamic data structures. Paradigm in Object-Oriented Programming.
3. Fundamentals of Internet: Layer Model. Link level (HDLC), network, and transport. Routing and client/server architecture.
4. Structure and protocols of public networks: Basic telephone network. Internet, mobile networks and CATV. Digital hierarchies.
5. Fundamentals and Applications of Telematics Services: Dynamic programming Web applications. Dynamic Web Applications. Using databases.
6. Multiservice Networks: Digital Subscriber Loop. Multiservice networks. Medium Access Methods. Network Segmentation.
7. Operating Systems: Architecture of Operating Systems. Distributed Operating Systems.
8. Software Engineering: Object Orientation. UML. Java.
9. Security: Encryption. Protocols. IPsec. Network security and services. Authentication, authorization and accounting.
10. Teletraffic: Queuing theory. Quality of Service. Multimedia real-time.
11. Network management: Models and network management protocols.
12. Network planning & simulation: Simulation network and dimensioning.
13. Advanced network architecture: MPLS, Virtual Private Networks, IMS.
14. Advanced Telematics Services: Domain Name Service, Centralized Directories, Email and Secure Shell.
15. Database design: Normal forms, database design. Installing and configuring servers.
16. Telematics project: Methodology and project design of networks and services.
17. Distributed Systems: Distributed object systems. Web Services: Architecture and Implementation.
18. Telematics System Management: Administration, configuration and maintenance in application servers.
19. Mobile Application Design: SSOO for mobile devices. Special Requirements. I/O Systems.
20. Interactive Application Design: Frameworks for interactive applications. Specificities according to the return network.

The virtual machine has been intensively used in the subject Fundamentals of Programming II. The following section shows the use case of the virtual machine in this subject.



#practical credits	#non-practical credits
52,5	53

Figure 6. Practical and non-practical credits

## V. USE CASE

This section illustrates the use of the virtual machine in a specific subject at Department of Telematics Engineering. We have chosen the subject Fundamentals of Programming II to illustrate the use of the virtual machine because it is a subject with a high practical content and have lots of students. First, we present the context, objectives, content, methodology and evaluation system in FPII. Then, we present the use of virtual machine in this subject.

### A. Context

FPII is a term core subject at the first-year of Degree in Telecommunication Technology Engineering. This course consists of 6 credits ECTS (European Credit Transfer and Accumulation System) corresponding to 150 hours of student work. 1,5 ECTS are dedicated to theoretical contents (15 classroom hours and 15 hours of study) and 4,5 ECTS are dedicated to practical contents (45 classroom hours and 45 hours of study). The course features a high practical content.

The course aims to consolidate basic skills obtained in a previous course (Fundamentals of Programming I or FPI) by solving a complex problem. The principles and fundamentals of programming are developed by means of problems resolution, in practical sessions.

### B. Objectives

The objectives of this course are the resolution of problems through structured programming techniques by the decomposition of complex problems, the management and implementation of abstract data types, the development of programs in a programming language such as C, providing both language syntax and the methods for handling data structures, and the design of applications using Object-oriented programming (OOP) techniques. This technique includes features such as data abstraction, encapsulation, messaging, modularity, polymorphism, and inheritance, in Java.

### A. Contents in FPII

The content in FPII of this course includes the resolution of problems through imperative programming using abstract data types, the development of programs in a programming language such as C using abstract data types, the resolution of problems through OOP techniques

and the development and implementation of an application in an OOP language such as Java, using OOP techniques.

### B. Methodology in FP II

The used methodology in FP II for the acquisition of knowledge includes the lectures (classes where the teacher makes a theoretical exposition of the subject matter, explaining the basics of the subject), laboratory practices (these classes are dedicated primarily to the student address and resolve the problems proposed by the teacher), active use of e-learning technologies (this technology has been used for storing contents, submission of practices, continuous evaluations, forums, internal e-mail), and Project-Based Learning.

Every student must complete a course work that will consist in developing an application, comprising the steps of understanding the problem, designing of the program, coding it in C and subsequent testing. Furthermore, every student must complete another course work that will consist in developing an application, comprising the steps of understanding the problem, designing of the program, coding it in Java and subsequent testing.

### C. Evaluation in FP II

The evaluation of the course in FP II is based on the attendance at practices, theoretical exam and two Course Projects (C and Java respectively).

In the course project in C the student must develop and implement a program for handling abstract data types. In the course project in Java the student must design and implement an application using Object-oriented programming (OOP) techniques. These projects consist of programs whose requirements will be provided at the beginning of the course. This work must be defended by the student in an exam. The teacher will request modifications to the program.

### D. Using the virtual machine

A link on the e-learning platform (WebCT) with instructions for installing virtualization software is offered to use the virtual machine. Accurate and necessary instructions for downloading and installing the virtual machine are provided.

Practices that are developed in the laboratories of the Computing Centre of the school can be completed at home by using the virtual machine. The main idea is that students can have a computer with an environment similar to the Computing Centre. So if required, the student can choose to complete the practical in this environment, and can validate the correct operation using the virtual machine provided.

Thus, students can opt to use in developing the proposed course work, by one side the Computing Centre, by the other side a computer at home with a similar environment to the Computing Centre environment, or finally a computer at home with the Virtual Machine.

The Computing Centre of the Engineering School of the University of Seville provides support to five undergraduate degrees, three grade of Excellence Campus International of Andalucía Tech, and seven Masters. The adoption of the new degrees and the adoption of the new Education Higher European Area (EHEA) require that the Computing Centre resources must be managed with a high degree of efficiency to support all qualifications.

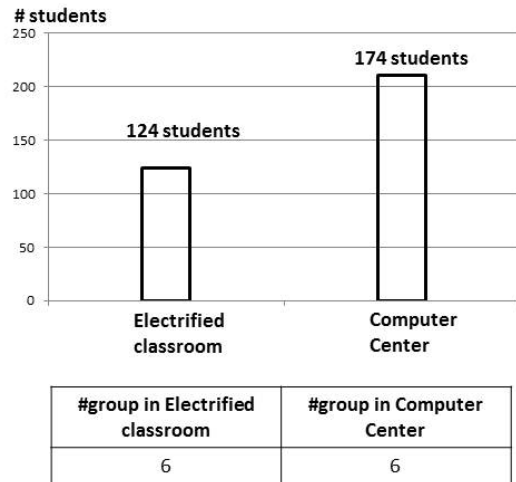


Figure 7. Student distribution.

The increased number of students and subjects require new infrastructure, not centralized in Computing Centre but in the student's home.

The use of a similar environment to the Computing Centre has the disadvantage that the students' work is evaluated in the Computing Centre, and may occur that, due to small differences in the development environment, in some cases, the operation is slightly different.

Using the virtual machine, you have the advantages of being able to work at home without having to go to the Computing Centre, and frees up resources of the Computing Centre of the school, minimizing the differences in the operation of the project at home and in the Computing Centre.

Due to limitations of the Computing Centre, some practice groups have had to develop practices in "electrified classroom". These electrified classrooms have no computers, but they have power outlets in each table. This enables the student to bring their own laptops to develop their practices in the classroom. Students may enroll in these groups voluntarily.

The number of groups that were held in electrified classrooms during 2011/12 is shown in Fig. 7. In electrified classrooms, while using student's laptops, is essential to have a common environment for the normal development of the laboratory. The virtual machine provides the common environment.

The student uses a tool for validation of his project. This tool allows the student to check the proper operation of the project in an autonomous way. This tool and instructions are provided through the virtual learning platform (Blackboard Learning System). The execution can be performed in the virtual machine.

## VI. CONCLUSIONS

This paper has shown the virtualization environment for practices and laboratories performed at the Department of Telematics Engineering (DTE) as support for the development of practical content in the subjects of this Department. The environment of the Computing Centre of the School of Engineering has been used as a reference to create a virtual machine.

The virtual machine allows the easy deployment of a Project-Based Learning (PBL) methodology. So, students

SPECIAL FOCUS PAPER  
COMMON VIRTUALIZED ENVIRONMENT IN TELEMATICS LABS

can analyze, organize, develop and test the software project, planning and organizing their own work.

The Virtual Machine has been used in subjects DTE since 2010/11 course, corresponding to the establishment of the new DTTE. This virtual machine represents a common point in teaching laboratories.

From the student's point of view, this single environment leads to an increase in the development of their practices, without detriment to the acquisition of knowledge. So, this tool represents a basis for the development of the student's abilities.

Moreover, negative experiences in evaluating projects and practices of the subjects have been avoided due to the virtual machine. These undesirable situations arise when a homogeneous and common environment is not used.

Finally, the virtual machine has also enabled the development of practices in electrified classrooms in the School of Engineering. So, the use of the rooms of the Computing Centre has been improved.

ACKNOWLEDGEMENT

The authors would like to thank to all teachers of the Department of Telematics Engineering. These results would not have been possible without their dedication.

REFERENCES

- [1] "Bologna beyond 2010 – Report on the development of the European Higher Education Area, Background Paper for the Bologna Follow-up Group", prepared by the Benelux Bologna Secretariat -, Leuven/Louvain-la-Neuve Ministerial Conference, 28-29 April 2009.
- [2] University of Sevilla, [www.us.es](http://www.us.es).
- [3] A.J. Sierra; T. Ariza; F.J. Fernandez, "Establishment EHEA for telecommunication technologies engineering degree", Global Engineering Education Conference (EDUCON), 2012 IEEE Publication Year: 2012, pp. 1-6. <http://dx.doi.org/10.1109/EDUCON.2012.6201158>
- [4] L. ChanLin, "Technology integration applied to project-based learning in science", *Innovations in Education and Teaching International*, 45, 2008, pp. 55-65. <http://dx.doi.org/10.1080/14703290701757450>
- [5] A.J. Sierra; T. Ariza; F.J. Fernandez; G. Madinabeitia, "TVSP: A tool for validation software projects in programming labs", Global Engineering Education Conference (EDUCON), 2012 IEEE Publication Year: 2012, pp. 1-6. <http://dx.doi.org/10.1109/EDUCON.2012.6201196>
- [6] A.J. Sierra; T. Ariza; F.J. Fernandez; G. Madinabeitia, "Tool for validation software projects in programming labs", *International Journal of Engineering Pedagogy (iJEP)*, Vol 2, No 2, 2012. <http://dx.doi.org/10.3991/ijep.v2i2.2080>
- [7] A.S. Tanenbaum, *Modern operating systems*, 3rd ed., NJ: Pearson Education, 2009.
- [8] J. Hu, D. Cordel and C. Meinel, "Virtual Machine Management for Tele-Lab IT-Security Server", Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC 2005), 2005.
- [9] P. Li, T. Mohammed, "Integration of virtualization technology into network security laboratory", 38th ASEE/IEEE Frontiers in Education Conference, Saratoga Springs, NY, October 22 – 25, 2008.
- [10] L. Tao, L.C. Chen, and C. Lin, "Work in progress - Improving Web security education with virtual labs and shared course modules," 40th ASEE/IEEE Frontiers in Education Conference, Washington, DC. October 27 - 30, 2010.
- [11] S. Ros, A. Robles-Gómez, R. Hernández, Agustín C. Caminero, and R. Pastor, "Using virtualization and automatic evaluation: Adapting network services management courses to the EHEA," *IEEE Transactions on Education*, Vol. 55, No. 2, May 2012. <http://dx.doi.org/10.1109/TE.2011.2160544>
- [12] K. C. Bandi, A. K. Nori, V. Choppella, S. Kode, "A virtual laboratory for teaching Linux on the Web", *IEEE International Conference on Technology for Education*, 2011.
- [13] VMware virtualization, <http://www.vmware.com/>.
- [14] VirtualBox, <https://www.virtualbox.org/>
- [15] WebCT (Course Tools) or Blackboard Learning System, <http://www.blackboard.com/>.
- [16] BOLETÍN OFICIAL DEL ESTADO, Núm. 245, martes 11 de octubre de 2011. Ministerio de la Presidencia. Gobierno de España. <http://www.boe.es/boe/dias/2011/10/11/pdfs/BOE-A-2011-16011.pdf>.
- [17] Grado en Ingeniería de las Tecnologías de Telecomunicación. Escuela Técnica Superior de Ingeniería. [http://www.us.es/estudios/grados/plan\\_199?p=4](http://www.us.es/estudios/grados/plan_199?p=4).

AUTHORS

**F.J. Fernández, A.J. Sierra, T. Ariza, G. Madinabeitia, and J.M. Vozmediano** are with Departamento de Ingeniería Telemática, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Sevilla, Spain. E-mail: {fjfj, antonio, teresa, german, jvt}@trajano.us.es.

This article is an extended and modified version of a paper presented at the EDUCON2013 conference held at Technische Universität Berlin, Berlin, Germany from March 13-15, 2013. Received 13 May 2013. Published as resubmitted by the authors 27 July 2013.