Remote Computational Labs for Educational Activities via a Cloud Computing Platform

J. Damian Segrelles
dquilis@dsic.upv.es

Germán Moltó
gmolto@dsic.upv.es

Miguel Caballer
micafer1@upv.es

"Instituto de Instrumentación para Imagen Molecular (I3M). Centro mixto CSIC – Universitat Politècnica de València – CIEMAT, camino de Vera s/n, 46022 Valencia, España"

Abstract

This paper describes the application of a Cloud Computing platform (ODISEA) to deploy and manage the infrastructure required to support remote computational labs across subjects that address computer-related topics such as Cloud Computing, Big Data and Scalable Architectures. ODISEA enables the lecturer to describe using a high level language the requirements (hardware, software and configuration) of the computational labs. The system provisions the required computing resources from multiple Cloud providers and automatically configures them for students to be able to access the remote labs to carry out the hands-on labs. This platform has been used for different subjects in a Master's Degree in Parallel and Distributed Computing and in an online postgraduate course to support the hands-on labs for well over 300 students across 9 countries since the academic course 2013/2014. The results show that ODISEA enables lecturers to easily deploy remote computational labs, thus being able to offer students access to a wide variety of computational infrastructures to train their appropriate skills. In turn, students gain anytime and anywhere connectivity to the hands-on labs.

Keywords: Cloud Computing, Virtualization, Technology-Enhanced Learning, Big Data.

1. INTRODUCTION

In general, Computational Virtual Environments (CVE) provide significant benefits for lecturers to support educational activities. These CVEs are presented as simulators or virtual laboratories such as (Le, X., Dijiang, H. and Wei-Tek, T., (2014)) or (Fang, Nielson and Kawamura, 2013)).

From the point of view of the lecturers, it is fundamental that the students solve realistic problems both in presential, blended and on-line frameworks. In this sense, the CVEs play a fundamental role in the development of the individual and groupal skills (see for example (Merabet and Jaime Sanchez, 2009; Di Blas and Paolini, 2014)) and are also perfectly integrated both for presential (Domingues, L., Rocha, I., Dourado, F., Alves, M. & Ferreira, E. C., 2010; Barella et al., 2009), blended (Erdosne Toth, E., Morrow, B. L. and Ludvico, L. R., 2008; Sancho, P., Corral, R., Rivas, T., González, M. J., Chordi, A. and Tejedor, C., 2006; Dantas and Kemm, 2008) and on-line (Balamuralithara and Woods, 2009; Bourne, J., Harris, D. and Mayadas, F., 2005) educational activities.

In this work, we will use the term CVE to describe a remote computational lab. This is a collection of computer resources that can run in disparate locations and which, together, are configured to support a certain educational activity. Students connect to the remote computational lab via SSH or Terminal Server depending on the underlying Operating System employed in the infrastructure (GNU/Linux or Windows, respectively). For example, consider the case of a remote lab with the Octave numerical software installed to which the students connect to carry out some activities.

There are many scenarios in which CVEs are of special importance. Consider the case of subjects that use proprietary software that is not freely available for students to install it in their computers. In addition, consider distributed computing subjects that require students to access complex infrastructures, such as clusters of PCs, Hadoop clusters (for Big Data), Grid infrastructures, etc. The ability to provide students with fully configured remote labs, dynamically deployed by the lecturer, introduces significant benefits for students. These benefits include, but are not limited to, access to a wide variety of infrastructures, and ubiquitous and 24x7 access to those infrastructures.

However, there are challenges and difficulties that need to be efficiently solved for CVEs to be widely used among educators, which are enumerated below:

a) Elasticity and Scalability of the CVEs. To manage the variation of the number of students involved and the number of CVEs required in the educational activities. The challenge is to use the minimum number of CVEs to adequately support the educational activities for the current number of students enrolled. It is important to be able to dynamically increase and decrease the number of CVEs to accommodate changes in the number of students (late admissions, dropouts, different teaching group sizes). In situations where the educational center cannot provide the required Physical Hardware Resources (PHRs) to execute the CVEs it is possible to provision additional resources from Iaas Cloud providers on a pay-by-the-hour model (Sultan, 2010). In the case of a public Cloud provider, such as Amazon Web Services (AWS), Windows Azure or Google Cloud Platform, a pay-per-use model is employed so that users are charged for the resources consumed in terms of hours of computing, network traffic, etc.

b) Reproducibility, Sharing and Reusability of the CVEs. The ability for the lecturer to reproduce a very same replica of a CVE at a later point in time, and share it so that the same CVE is reused among different educational activities, within the same subjects and across subjects. Cloud computing (Buyya, Broberg and Goscinski, 2011; Hwang, Dongarra and Fox, 2011) enables the users to rapidly provision a pool of configurable resources (computing and storage, mainly). These represent the basic resources required for CVEs, which are customized with a specific hardware and software configuration to suit a particular educational activity.

This way, it is possible to deploy the CVEs as customized VMs that run on a set of PHRs. For that, one can use pay-as-you-go resources provisioned from a public Cloud provider, such as Amazon Web Services

(AWS)[1], Windows Azure[2] or Google Cloud Platform[3]. It is also possible to deploy the CVEs on an on-premises Cloud infrastructure offered by the educational organization, using Cloud Management Platforms (CMPs) such as OpenNebula (Borja Sotomayor, Ruben Santiago Montero, Ignacio Martín Llorente, 2008) or OpenStack[4].

c) User Friendly Interface to manage CVEs. The lecturer should be able to manage all these features (elasticity, scalability, reproducibility and sharing) in an autonomous and friendly way. The usability is widely recognised as critical to the success of any software tool that interacts with the end users (Parlangeli, O., Marchigiani, E. and Bagnara, S., 1999).

2. OBJECTIVES

This paper evaluates a cloud platform to deploy CVEs on-demand both on public and on-premises Clouds, with the precise software configuration required to perform specific educational activities. This platform is named ODISEA (On-demand Deployment of virtual Infrastructures to Support Educational Activities).

The main objective of this work is to validate the ODISEA platform in 6 educational activities performed through 6 subjects involved in two different degrees in the Universitat Politècnica de València (UPV) since the academic course 2013/2014, and also assess its use from the point of view of the lecturers. The parameters to evaluate are the following:

- The capability to provide elasticity and scalability of the CVEs with respect to the number of students and educational activities.
- The capability to reproduce a precise clone of a CVE at a later point in time, both for failure recovery and for periodical activities.
- The capability to share and reuse the CVEs among different educational activities promoting the use of diverse CVEs within these activities.
- The user-friendly interface of the platform to manage the CVEs.

3. MATERIALS AND METHODS

To accomplish the aforementioned goals we have relied both on public Cloud providers and on-premises Cloud deployments, where a set of CVEs required in educational activities have been deployed through the ODISEA platform.

Cloud Providers

For the evaluation of the parameters defined as objectives in section 2, we have relied on two different Cloud platforms:

- An on-premises IaaS (Infrastructure as a Service) Cloud platform with a total 128 cores and 352 GB of RAM managed by OpenNebula 4.2 to provision the VMs required for the CVEs. This infrastructure can be easily reproduced in an educational institution, since it just relies on commodity hardware and open-source software.
- The Amazon Web Services (AWS) public Cloud, which is the pioneer and current largest Cloud provider. It allows users to deploy VMs and storage capacity on a pay-as-you-go basis. This would allow lecturers or educational institutions to access a large amount of resources without any investment, only paying for the resources that have been used. Cloud providers typically bill by the hour (as is the case of AWS) or by the minutes (as is the case of the Google Cloud Platform).

ODISEA Platform Architecture

The goal of ODISEA is to simplify the definition of CVEs and to perform their automated deployment, by dynamically provisioning and relinquishing the related PHRs, together with the appropriate

[1] Amazon AWS: http://aws.amazon.com
[2] Microsoft Azure: http://azure.microsoft.com
[3] Google Cloud: https://cloud.google.com
[4] OpenStack: http://www.openstack.org/

configuration to fit the requirements of a certain educational activity.

The CVEs are described using a higher level declarative language called RADL (Resource and Application Description Language) (Caballer et al. 2014)). Each description is a recipe of the hardware, software and configuration. This recipe can later be instantiated on multiple Cloud infrastructures. An infrastructure can be composed of one or more Virtual Machines (VMs) with possibly different configurations. The usage of a high-level recipe means that the same computing infrastructure can be deployed on an on-premises Cloud and a public Cloud. Also, to have recipes for each infrastructure enables to automate their deployment process for different educational activities, such as a periodic course.

The infrastructure descriptions are made available to the lecturer through a web application. Infrastructures can therefore be instantiated, which means to actually create and configure the VMs to deploy the computing infrastructure in a Cloud back-end, with a lecturer's single click. This triggers all the work of ODISEA which is in charge of instantiating the VMs, by interacting with different IaaS Cloud providers, dynamically installing the specified software and providing the configuration among the VMs in order to create an available CVE that satisfies the requirements specified in the high level, declarative RADL description.

Figure 1 summarizes the main components of the ODISEA platform. It consists of:

- **Catalog of Virtual Machine Images (VMIs)**. This catalog is implemented by means of the VMRC (Virtual Machine image Repository and Catalog) (Carrion et al. (2010)), which indexes a set of virtual appliances with different Operating Systems and/or applications;
- **Repositories for software and educational material**. This repositories are required to install the appropriate dependencies to turn the CVEs into learning environments with the appropriate software tools and materials,
- **Infrastructure Manager (IM) (Caballer et al.(2014)).** This component in charge of the deployment and configuration of the virtual infrastructures on a Cloud provider, with the features described in the RADLs.

The main components that comprise this platform have been released as open-source[5] and the graphical front-end of the platform, provided by the IM, can be publicly accessed online[6], for demonstration purposes. In particular, this paper focuses on the application of ODISEA to different learning activities, and not in the architecture itself. Additional technical details regarding ODISEA can be found in (Moltó et al. (2014)).

ODISEA Platform Methodology

This section outlines the methodology employed to create and deploy the CVEs. This procedure can be adapted to different subjects.

Step 1. Identify the requirements of each CVE.

In this step, the lecturer has to identify the hardware, software and configuration required for the CVEs and the number of CVEs required. The hardware configuration determines the computing and storage capacity (both hard disk and RAM) that each CVE will have. The software configuration includes applications and libraries to be installed, either from existing repositories or other sources. Other configuration includes user accounts, so that multiple students can access the CVE using their own credentials, practical guides for the hands-on labs, datasets, etc.

Notice that the number of CVEs depends on the amount of students and the workload that each student will perform in the CVE. A detailed study has to be performed per subject in order to provision the minimum

[5] Available at GitHub: https://github.com/grycap
[6] http://www.grycap.upv.es/im

number of CVEs that enable students to perform the practice sessions with the appropriate level of service.

Step 2. Select/Generate the Virtual Machine Images (VMIs).

This step requires either (A) to produce the required Virtual Machine Images (VMIs) with the specified configuration or (B) rely on existing vanilla VMIs and let ODISEA configure them right after they boot. Depending on the complexity and time to install the software, it might be necessary to create the VMIs but this is not mandatory. Using automated configuration simplifies the work of the instructor at the expense of requiring additional time to configure the CVEs when the VMs are provisioned from a Cloud infrastructure.

Step 3. Generate receipt of each CVE.

This step involves creating the high level recipes that describe the aforementioned hardware, software and configuration requirements. For that, we use RADL, which provides the lecturers with expressiveness enough to describe a CVE. Describing the details of RADL is out of the scope of this paper, since these are described in the IM's documentation[7]. Each CVE is, therefore, described by a single document, which specifies all the required VMs and their configuration. These documents, being a text-based document can be shared and exchanged among lecturers to reuse the descriptions and create additional CVEs for different subjects.

Step 4. Deploy and undeploy CVEs

In this final step, the CVEs are deployed before the educational activity starts through a Command Line Interface (CLI) or via a web-based Graphical User Interface (GUI), as shown in Figure 2. This enables to provision the computational resources just in time for the educational activity. Notice that when the CVEs are no longer required they are relinquished in order to cut down the costs, in the case of using a public Cloud and to release the resources for other CVEs from other subjects, in the case of using an on-premises Cloud.

This approach to deploy remote labs (CVEs) introduces an unprecedented degree of flexibility for professors, which can easily provision the resources required for their students to perform the practice activities.

Educational Activities

The ODISEA platform has been used in our university since the academic course 2013/2014. The educational activities employed for the assessment appear in five academic subjects in the Master's Degree in Parallel and Distributed Computing (MCPD)[8] and an Online Postgraduate Course (OPC)[9] at the Universitat Politècnica de València (UPV). The subjects and the number of students involved are shown in Table 1.

Table 1: Code, description and number of students in each subject.

Degree	Code	Description	N
MCPD	CCGC	Grid and Cloud Computing Concepts	9
MCPD	MPG	Grid Programming Models	6
MCPD	IAG	Advanced Grid Infrastructures	6
MPCD	MPC	Cloud Programming Models	6
MCPD	IAC	Advanced Cloud Infrastructures	6
OPC	CursoCloudAWS	AWS Cloud Services	298

The Educational Activities (E.A.) involved in this work are described below, explaining the use of each CVE employed. A brief description of each CVE is shown in Table 2 and the relation between subject,

[7] RADL: http://www.grycap.upv.es/im/doc/radl.html

[8] http://www.upv.es/entidades/MCPD/indexi.html

[9] http://www.grycap.upv.es/cursocloudaws

educational activity and CVE is shown in Table 3.

Table 2: Description of the CVEs used in the educational activities.

Code	Description
GT4	CVE for developing grid applications with the GT4.2[10]
GLiteUI	CVE for interacting with Advanced Grid Infrastructures such as EGI[11], using the CLIs of gLiteUI package[12].
HADOOP	A CVE to perform Big Data analysis consisting of a cluster Hadoop with 10 nodes.
ANSIBLE	A CVE to be automatically configured by the students using Ansible, a configuration tool.
ONE	A CVE employed to submit VMs to an OpenNebula testbed deployment.

Table 3: Educational Activities (E.A.x) in which each CVE has been used.

Code	Educational Activities
GT4	E.A.1(CCGC) - E.A.2(IAG)
GLiteUI	E.A.2(IAG) - E.A.3(MPG)
HADOOP	E.A.4 (MPC)
ANSIBLE	E.A.5 (IAC, CursoCloudAWS)
ONE	E.A.6 (IAC)

E.A.1. Basic Grid Applications. This activity belongs to CCGC and 9 GT4 CVEs are required, one for each registered student. The student develops basic Grid applications. The CVEs have to be enabled during all academic period of the subject (8 sessions of 4 hours). The availability has to be 24x7 through SSH connections using the students laptops or PCs to perform the classroom tasks and autonomous homework. The VMs that support the CVEs are deployed on the on-premises Cloud. When the subject is completed, the CVEs are undeployed and the resources released.

E.A.2. Basic Grid Application based on Advanced Infrastructures. This activity belongs to IAG. 6 GT4 CVE are required, one for each registered student and only one GLiteUI CVE, where multiple user accounts are available for students. The students develop basic Grid Services based on the Web Services Resources Framework (WSRF) and implement basic Grid applications that interact with advanced infrastructures such as EGI. All CVEs have to be available during all academic period of the subject (8 sessions of 4 hours), accesible 24x7 through SSH connections using the student laptops or PCs to perform the classroom tasks and autonomous homework. The VMs that support the CVEs use the on-premises cloud to deploy them. When the subject is completed, the CVEs are undeployed and the resources released.

E.A.3. Advanced Grid Applications. This activity belongs to MPG and a only one GLiteUI CVE, where each student has been assigned an user account, has been required. The students develop advanced Grid applications that interact with advanced infrastructures based on the gLite Middleware such as EGI. The CVE has to be available during all the academic period of the subject (8 sessions of 4 hours). The availability have to be 24x7 through SSH connections using the students laptops or PCs to perform the classroom tasks and autonomous homework. The VMs that support the CVEs use the on-premises Cloud to deploy them. When the subject is completed, the CVE is undeployed and the resources released.

E.A.4. Analysing Big Data with MapReduce. This activity belongs to MPC and only one HADOOP CVE (where each student has one user account) has been required. The students create programs based on the MapReduce programming

[10] Globus Toolkit. https://www.globus.org
[11] European Grid Infrastructure. http://www.egi.eu
[12] gLite Middleware. http://www.glite.eu

model to extract information from large datasets available in a Hadoop cluster. It has to be available 24x7 and accessible via SSH. The CVE is deployed on the on-premises Cloud and on the AWS public Cloud to carry out different performance analysis.

E.A.5. Automatic Configuration of Infrastructures. This activity is performed both in IAC and CursoCloudAWS. Twice as many ANSIBLE CVEs as students (two for each student) have been required. The students perform the configuration of a set of VMs using the Ansible DevOps tool. They are dynamically deployed by the students with the help of some scripts and they have to be available 24x7, accessible via SSH. They are deployed on AWS. Notice that this involves a potentially large number of VMs, although they are typically used for a few hours and students reside in different time zones (it's a worldwide online course). Therefore, a reasonable number of simultaneous VMs is typically found.

E.A.6. Provision of VMs from an on-premises Cloud. This activity belongs to IAC and only one ONE CVE (where each student has one user account) has been required. Students use the functionality of OpenNebula to manage VMs and dynamically allocated storage from an OpenNebula deployment. It requires a CVE to which the students connect via SSH (anytime, 24x7) using the assigned credentials.

4. RESULTS

4.1. CVEs Implemented in ODISEA Platform

The result of this experience has been the CVEs implemented by means of ODISEA to support the different educational activities described in section 3. The CVEs implemented are shown in Table 2.

Step 1. Identify the requirements of each CVE.

For each CVE, we identified details about the number of VMs required, Operating System used in these VMs, the software and hardware requirements, and configuration aspects. All these requirements (R1 through R7) are summarized in Table 4 (see appendix).

Step 2. Select/Generate the Virtual Machine Images (VMIs).

Two VMIs have been generated, whose requirements are shown in Table 4. The features (S.O, hardware and software requirements) are exposed in Table 5.

Table 5: Generated VMIs to support the CVEs.

VMI	S.O	Hw	Sw
VMI_1	S.L.5 10	4GB,X86/64 10GB HD	JDK 1.6
VMI_2	Ubuntu 12	2GB,x86/64 10GB HD	--

Step 3. Generate receipt of each CVE.
All configuration requirements and dependences between the VMIs and VMs have been implemented in the platform creating a configuration recipe using the high level declarative language of ODISEA for each CVE identified. Table 6 is a summary of the recipes.

Table 6: Features of the recipes implemented for each CVE.

CVE	VMI	VMs	Sw	Conf Req.
GT4	VMI_1	1	GT4 Ant1.7	R1, R2
GLiteUI	VMI_1	1	gLiteUI	R1, R3, R4
HADOOP	VMI_2	10	Hadoop	R1, R5
ANSIBLE	VMI_2	2	Ansible	R6
ONE	VMI_2	2	Open-Nebula	R1, R7

The ODISEA platform provides automatic deployment and configuration of CVEs. However, for the platform to be productive, CVEs should be deployed in a matter of minutes and not hours. This would foster elasticity, the ability to deploy additional CVEs to accommodate increased workload (additional students). This also includes the ability to increase the number of resources within a single CVE (i.e., increasing the number of nodes of a Hadoop cluster).

Therefore Table 7 shows the times required to deploy a Hadoop cluster and to add and remove an additional node to a running cluster in the two different Cloud platforms that we used, the OpenNebula (ONE) on-premises Cloud and the Amazon Web Services (AWS) public Cloud. These Hadoop clusters are used to carry out activities related to Big Data and include not only the deployment of the clusters themselves but also the installation and configuration of additional software required for the practice lessons, such as Hive, Pig and others.

Table 7: Deployment times for the Hadoop CVE on the different Clouds

Action	ONE (6)	ONE (11)	AWS (11)	AWS (51)
Creation	12:05	22:13	9:29	14:26
Node Addition	5:13	5:33	5:46	11:23
Node Removal	1:18	1:32	3:17	6:36

You can notice that it is possible to automatically deploy and configure a 51-node Hadoop cluster in slightly less than 15 minutes, and add and remove nodes in 5-11 minutes, depending on the platform. Notice that a Hadoop cluster is a rather complex virtual infrastructure. Therefore, the ability to enact this deployment in such a reduced amount of time gives an idea of the benefit of ODISEA. This is a reproducible environment than can be deployed as many times as needed. Compare this with a manual installation of such CVE for each subject or academic year.

5. ASSESMENT AND DISCUSSION

Once identified the application of ODISEA to the deployment of CVEs across different subjects, this section discusses the benefits that such a platform should provide, as stated in the introduction.

The capability to provide elasticity and scalability of the CVEs with respect to the number of students and educational activities.

ODISEA allows deploying and undeploying CVEs in an elastic way. The only limitation for deploying new CVEs is the amount of PHRs offered by the underlying clouds. In this sense, ODISEA allows to use on-premises (Open-Nebula, OpenStack and any OCCI-compliant Cloud Management Platform) and public clouds (AWS, Google Compute Engine, etc.) depending on the PHRs required. For example, in the E.A.5, when the online course was performed, a public cloud was used and the number of students was variable (students can enrol anytime). This way, a sudden increase in the number of students enrolled in the course can be easily accommodated with the deployment of an additional remote laboratory with the required configuration in the cloud so that new students can connect to perform their practical lessons. Being able to dynamically adapt the number of CVEs to the actual number of students accounts for better resource utilization and, in the case of using a public Cloud, leads to reduced cost.

The capability to reproduce a precise clone of a CVE at a later point in time, both for failure recovery and for periodical activities.

Some CVEs used in the educational activities have been cloned when a failure has happened or for performing periodical activities. For example, in the E.A.1, E.A.2 or E.A.5 it is necessary that each student performs configuration tasks in his/her own CVE to do the activity. If the work performed by the student results in a misconfigured CVE (because they need to have root/administrative privileges to carry out the activities), another CVE is spawned to start from a fresh environment. This ability introduces unprecedented flexibility for lecturers, since they can deploy reproducible environments on-demand for their students. In turn, students feel confident that no matter what they do, the CVEs can be restored to their initial state.

The capability to share and reuse the CVEs among different educational activities promoting the use of diverse CVEs within these activities.

Two CVEs implemented in ODISEA platform (GT4 and GliteUI) have been shared among different educational activities (see Table 3). The CVEs are shared among close knowledge areas as it happens with Grid Computing in the MCPD. Moreover, the capability of ODISEA to share the CVEs foster the use of a greater number of CVEs in the subjects. For example, the subjects of IAG or IAC have used two different CVEs that have been reused in other subjects. This ability to reuse the CVEs across subjects paves the way for integrating such a platform at the educational center level so that lecturers can reuse previously defined CVEs to be incorporated within their subjects.

The user-friendly interface of the platform to manage the CVEs.
Figure 1 shows the graphical user interface that the IM provides. A web-based dashboard provides the instructor with a description of the CVEs and the ability to deploy them on-demand on the different Cloud providers in which the instructor has valid credentials. This enables to manage the life cycle of CVEs in order to deploy and relinquish them as required.
The interface supports multiple users which can declare their RADLs (the description of the CVEs) to be shared by other users, thus leveraging sharing. This capability enables lecturers to deploy CVEs with just a web browser either on the on-premises Cloud provided by the educational center or, alternatively, on a public Cloud on which the lecturer has a valid account.

5. CONCLUSIONS

This paper has described the application of the ODISEA platform in 6 different computationally-related subjects, involving both traditional face-to-face education and fully asynchronous online education. This has allowed to deploy Computational Virtual Environments (CVEs) for students to carry out the hands-on labs on virtualized infrastructures deployed on multiple Cloud platforms that range from on-premises Clouds, within the educational center, to public Clouds, on a pay-as-you-go model.
This approach introduces a degree of flexibility that could not be imagined some years ago. Lecturers automatically deploying the infrastructure required for their remote labs just in the time for the educational activity, with the simplicity of a web browser, and the capability perform the deployment on multiple Clouds.
We truly believe that educational institutions will adopt Cloud computing either on-premises or public Clouds to outsource the computing infrastructure required to execute these CVEs. With this approach, students have been able to access a wide myriad of virtualized infrastructures that, otherwise, would have require a larger amount of physical resources and human dedication to their manual configuration.
By introducing Cloud computing and automatic configuration, this complexity is removed by enabling dynamic deployment and configuration from a web-based graphical user interface.

6. ACKNOWLEDGEMENTS

The authors wish to thank the financial support received from Vicerrectorado de Estudios, Calidad y Acreditación of the

Universitat Politècnica de València (UPV) to develop the PIME project "Análisis y Evaluación de Impacto del Cloud Computing en la Gestión de Entornos Virtuales Computacionales en la Enseñanza", with reference A014 and the "Ministerio de Economía y Competitividad" for the CLUVIEM project with reference TIN2013-44390-R. GM would also like to thank the "Escola Tècnica Superior d'Enginyeria Informàtica de la Universitat Politècnica de València" for the economic support to produce this paper.

6. REFERENCES

Balamuralithara, B. & Woods, P. C. (2009), "Virtual laboratories in engineering education: The simulation lab and remote lab", Computer Applications in Engineering Education 17(1), 108-118.

Buyya, R., Broberg, J. & Goscinski, A. M. (2011), Cloud computing: Principles and Paradigms, Wiley.

Bourne, J., Harris, D. & Mayadas, F. (2005), "Online Engineering Education: Learning Anywhere, Anytime", Journal of Engineering Education 94(1), 131-146.

Caballer, M., Blanquer, I., Moltó, G. and de Alfonso, C. (2014), "Dynamic management of virtual infrastructures", Journal of Grid Computing 13(1), 53-70.

Carrion, J. V., Molto, G., De Alfonso, C., Caballer, M. and Hernández, V. (2010), A Generic Catalog and Repository Service for Virtual Machine Images, in "2nd International ICST Conference on Cloud Computing (Cloud-Comp 2010)".

Dantas, A. M. & Kemm, R. E. (2008), "A blended approach to active learning in a physiology laboratory-based subject facilitated by an e-learning component.", Advances in physiology education 32(1), 65-75.

Di Blas, N. & Paolini, P. (2014), "Multi-User Virtual Environments Fostering Collaboration in Formal Education", Educational Technology & Society 17(1), 54-69.

Domingues, L., Rocha, I., Dourado, F., Alves, M. & Ferreira, E. C. (2010), "Virtual laboratories in (bio)chemical engineering education", Education for Chemical Engineers 5(2), e22-e27.

Erdosne Toth, E., Morrow, B. L. & Ludvico, L. R. (2008), "Designing Blended Inquiry Learning in a Laboratory Context: A Study of Incorporating Hands-On and Virtual Laboratories", Innovative Higher Education 33(5), 333-344.

Fang, N., Nielson, K. & Kawamura, S. (2013), "Using Computer Simulations with a Real-World Engineering Example to Improve Student Learning of High School Physics: A Case Study of K-12 Engineering Education".

Hwang, K., Dongarra, J. & Fox, G. C. (2011), Distributed and Cloud Computing: From Parallel Processing to the Internet of Things [Paperback], Morgan Kaufmann; 1 edition.

Le, X., Dijiang, H.,& Wei-Tek, T., (2014), "Cloud-Based Virtual Laboratory for Network Security Education". IEEE Transactions on Education, 57(3), 145-150.

Merabet, L. B. & Jaime Sanchez (2009), "Audio-Based Navigation Using Virtual Environments: Combining Technology and Neuroscience", AER Journal: Research and Practice in Visual Impairment and Blindness 2(3).

Mell, P. & Grance, T. (2011), "The NIST Definition of Cloud Computing". NIST Special Publication 800-145.

Moltó, G., Segrelles, J. D. and Caballer, M. (2014), Gestión de recursos computacionales en el Cloud para actividades educativas, in "XX Jornadas sobre la Enseñanza Universitaria de la Informática (JENUI 2014). ISBN:,pp. 99-106.

Parlangeli, O., Marchigiani, E. & Bagnara, S. (1999), "Multimedia systems in distance education: e_ects of usability on learning", Interacting with Computers 12(1), 37-49.

Sancho, P., Corral, R., Rivas, T., González, M. J., Chordi, A. & Tejedor, C. (2006), "A blended learning experience for teaching microbiology.", American journal of pharmaceutical education 70(5), 120.

Sultan, N. (2010), "Cloud computing for education: A new dawn?", International Journal of Information Management 30(2), 109-116.

APPENDIX

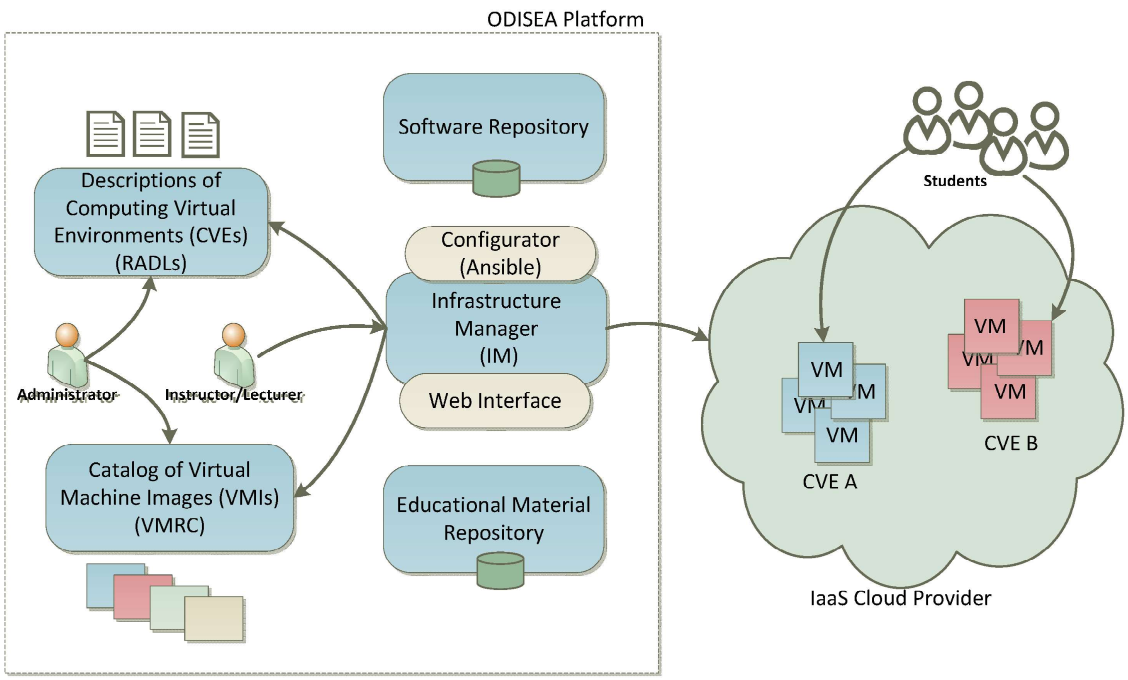

Figure 1: Architecture of ODISEA (as introduced in Moltó et al, 2014).

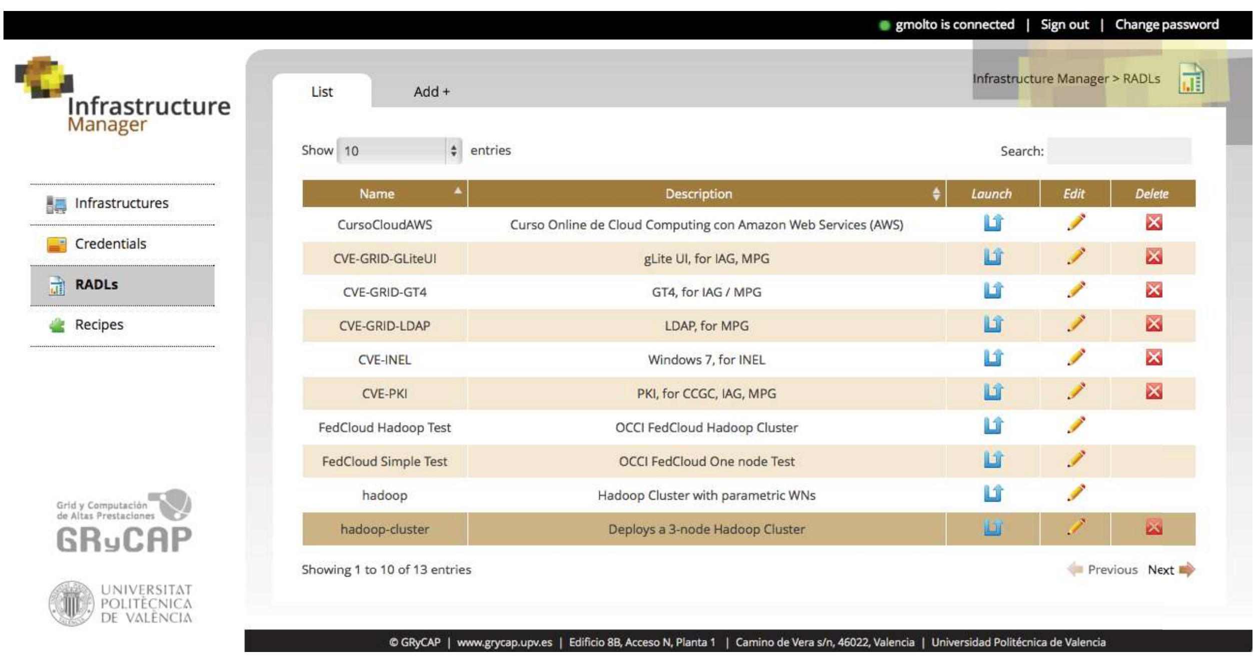

Figure 2: ODISEA graphical user interface.

Table 4: Number of VMs and requirements hardware and software of all CVEs implemented at the ODISEA platform validation.

Code	VMs	O.S	Hardware	Software	Configuration Requirements
GT4	1	S.L.5 10	4GB,X86/64 10GB HD	GT4, JDK1.6 Ant1.7	R1.-Accessible via SSH with a password. R2.-User privileges to develop Grid applications.
GLiteUI	1	S.L.5 10	4GB,X86/64 10GB HD	gLite UI JDK 1.6	R1.-Accessible via SSH with a password. R3.-X.509 cert. to connect to European Grid infrastructure (EGI). R4.-Multiple users to perform the gLite CLI.
HADOOP	10	Ubuntu 12	2GB,x86/64 10GB HD	Hadoop	R1.-Accessible via SSH with a password. R5.-Multiple user and different pre-staged large datasets.
ANSIBLE	2	Ubuntu 12	2GB,x86/64 10GB HD	Ansible	R6.-Accessible VMs via SSH without password.
ONE	2	Ubuntu 12	2GB,x86/64 10GB HD	Open-Nebula	R1.-Accessible via SSH with a password. R7.-Multiple users to access the testbed.