

# OERaaS: Open Educational Resources as a Service With the Help of Virtual Containers

A. Corbi and D. Burgos

**Abstract**— The new and challenging virtual containers are wide spread in a variety of sectors like systems administration, application deployment and service provision. However, these tools are not that much used in education and training, either in face-to-face or online forms. In this paper we show a technological set based on these containers and we discuss about a potential role in academic settings as a support to Open Educational Resources (OER). Furthermore, we present a very specific case study carried out with freshman students of the Bachelor of Science in Computer Science. These online students concentrate on a STEM subject, which stands for Science, Technology, Engineering and Maths. The promising results out of the tool evaluation might be migrated to other subjects, academic fields and educational levels. Likewise, we present already existing repositories of containers and discuss how good they are as a distribution service of OERs

**Keywords**— virtual containers; education; STEM; Open Educational Resources; distribution platform

## I. INTRODUCCIÓN

LA DOCENCIA de materias STEM (*Science, Technology, Engineering and Math*) normalmente requiere por parte del alumnado el acceso y uso de recursos educativos de índole matemático-científico tales como simulaciones, software de diseño y paquetes de cálculo, entre otros. Ciertas instituciones y proyectos educativos promueven a su vez la libre (pero no necesariamente bajo supervisión) distribución de estos recursos como Recursos Educativos Abiertos (OERs) que se discutirán en la Sección I-C.

El alumno debe normalmente proceder a la descarga de cada una de estas herramientas, su instalación, configuración y uso apropiado y todo ello sin contar con la necesaria fase de aprendizaje sobre el funcionamiento de la herramienta en particular.

En la enseñanza presencial, este proceso puede tener lugar mediante los medios y en las mismas instalaciones provistas por la propia institución educativa. Si embargo, en enseñanza a distancia, el alumno se encuentra mayoritariamente *solo* ante estas situaciones. Esta brecha puede llegar a agrandarse incluso más en el caso de los OERs, donde el alumno, a pesar de poder tener acceso al recurso educativo, no tiene en muchos casos acceso a soporte, formación y documentación apropiada para su uso.

Algunas de estas utilidades comportan además un nivel de complejidad elevado para su correcto despliegue. Otras

requieren de unas condiciones muy específicas para su funcionamiento (sistema operativo, *frameworks* específicos, elementos de software/hardware preinstalados, permisos de usuario, versiones específicas, etc.). Ante esta situación, las únicas alternativas son las siguientes:

- Exigir al alumno la adquisición o réplica exacta de las arquitecturas y condiciones específicas necesarias por las actividades y materias impartidas.
- Permitir el acceso remoto por parte del alumno a un entorno de trabajo controlado y desplegado en la institución o fuera de la misma (en un servicio *cloud* o *hosting ad hoc* tales como Microsoft Azure, Salesforce Heroku, Redhat OpenShift, etc.).
- Limitar las tecnologías subyacentes al software requerido a estándares tecnológicos de amplio consenso e implantación, tanto *de facto* como auspiciados por organismos internacionales (W3C y HTML5, C#/Javascript y ECMA, C++ e ISO, etc.).
- Virtualizar el entorno de trabajo mediante las así llamadas *tecnologías de virtualización* [1].

De todas estas opciones, la única que puede tener un recorrido exitoso para una gran mayoría de escenarios es, sin duda, la virtualización, que justificaremos en la Sección I-A. Las razones para excluir las tres alternativas precedentes son variadas: desde las meramente técnicas a las legales [2]: salvaguardado de datos sensibles en servicios *offshore*, complejidad de las licencias de software multiusuario, etc.

Respecto a las razones meramente técnicas y complementando las arriba ya expuestas: algunas aplicaciones necesitan de importantes recursos computacionales para ser ejecutadas en un entorno de tiempo compartido y su rendimiento decae importantemente en escenarios remotos o *cloud*. Por otro lado, pueden depender de ciertos componentes preinstalados no contemplados por la institución o la empresa gestora del servicio externo. También, a pesar de que reduzcamos el conjunto de exigencias a estándares tecnológicos ampliamente *respetados y universales*, no es garantía de un correcto funcionamiento de la solución educativa. Por último, otras experiencias de laboratorio concretas, como las que se comentan en la Sección I-E pueden requerir escenarios de software altamente complejos y muy difíciles de reproducir por parte del usuario (en nuestro caso, el alumno).

Como ya se ha adelantado, en este trabajo proponemos la virtualización, y en especial los incipientes *contenedores virtuales* (tratados con detalle en la Sección I-B) como vehículo de transmisión de recursos educativos *complejos*. Estudiaremos una implementación concreta de estos contenedores, Docker, que a su vez basa gran parte de su atractivo en la gestión de un repositorio abiertamente accesible

A. Corbi, Universidad Internacional de La Rioja (UNIR), La Rioja, España, ruben.gonzalez@unir.net.

D. Burgos, Universidad Internacional de La Rioja (UNIR), La Rioja, España, carina.gonzalez@unir.net.

de estos entornos virtualizados. Este repositorio puede llegar a tener un cariz de *gestor de OERs*, con lo que podríamos estar hablando de un sistema OERaaS (*OER as a Service*) *de facto* y plenamente preparado para su uso académico. A modo de caso de uso, se comentarán las experiencias práctica llevadas a cabo en una asignatura online del área STEM en concreto (Sección II) y se discutirán los resultados obtenidos en la misma (Sección III).

### A. Entornos de virtualización clásicos

La virtualización, en su concepción *tradicional*, es la creación a través de software exclusivamente (aunque puede darse cierto apoyo por parte de componentes hardware [3]) de una versión *potencial* de un elemento tecnológico *real* como una computadora moderna en su completitud (lo que se denomina *máquina virtual* o *virtual machine*) o elementos hardware aislados como un disco o un nodo de red. Durante el proceso de virtualización se ocultan ciertos recursos de este sistema informático. Estos recursos están a su vez controlados por un así denominado *hipervisor* o monitor del escenario virtual. El hipervisor gestiona crea una capa de abstracción entre el hardware de la máquina física (*host*) y el sistema operativo de la máquina virtual (*guest*).

A pesar de tratarse de entornos emulados, la eficiencia y versatilidad de los entornos virtuales modernos rivaliza con el rendimiento de las máquinas reales [4, 5]. La Fig. 2 muestra un ejemplo de esta rivalidad para el caso de acceso a una base de datos [6] de tipo SQL.

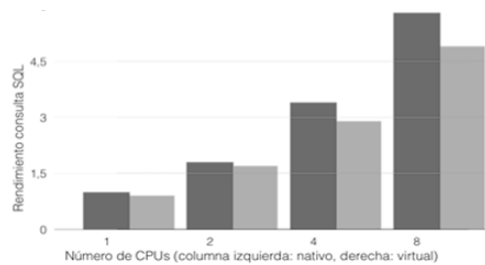


Figura 2. Rendimiento nativo vs. virtual en acceso a bases de datos

Las principales empresas y fabricantes de tecnologías de virtualización *clásica* son Oracle con VirtualBox, Parallels y VMware. Este tipo de virtualización tiene como principal desventaja la generación desde cero de nuevos entornos para cada nueva actividad de interés. Es decir, para cada tarea se ha de distribuir una nueva máquina virtual completa, con el gran consumo en ancho de banda, reducción del rendimiento y la capacidad de almacenamiento físico que ello conlleva. Hoy en día este problema ha quedado bastante subsanado gracias a los contenedores virtuales, que discutiremos a continuación. A su vez, cada nueva máquina se distribuye bajo ciertos estándares industriales consensuados (OVF, VDI, VDMK, etc.) por compañías y proyectos adscritos al *Open Grid Forum* [7].

En cuanto a la aplicación de máquinas virtuales en educación, existen interesante estudios llevados a cabo, entre otros por CISCO [8], el cual analiza las barreras que están

impidiendo la entrada de la virtualización en las escuelas. Principalmente, estas barreras son:

- falta de habilidades y conocimientos por parte de profesores y alumnado,
- falta de recursos por parte de los centros e instituciones educativos, y por último,
- desacuerdos entre los actores del sistema educativo.

IBM, por otra parte [9] vaticina un auge imparable de la virtualización en todas las esferas tecnológicas y recomienda a los servicios de IT de los colegios que se sumen a esta tendencia. Por último, otros grupos de investigación en tecnologías educativas, como [10], realizan un exhaustivo análisis de las tecnologías de virtualización existentes y su aplicación a la enseñanza de materias relacionadas con redes.

### B. Contenedores virtuales

Los contenedores virtuales [11] son *cuasi-máquinas virtuales ligeras* (típicamente basadas en el sistema GNU/Linux) destinadas a ejecutar una instancia de una aplicación específica (y no un típico escenario virtual completo con pantalla, escritorio y aplicaciones variadas). Un contenedor tiene normalmente como misión implementar un servicio web: una app Ruby on Rails, NodeJS o PHP. La manera de hacerlo es mediante la ejecución de una máquina virtual *parcial* que implemente exclusivamente el software justo y necesario.

Los contenedores se están convirtiendo en los grandes aliados de los programadores, administradores de sistemas y *devops* (*development and operations*) pues con ellos se pueden desplegar aplicaciones o sistemas con facilidad en cualquier computadora que tenga el soporte mínimo pre-instalado. La principal ventaja es su ligereza y la capacidad de poder trabajar tanto en entornos de desarrollo como de producción. La diferencia respecto a máquinas virtuales *tradicional* radica en que todas las aplicaciones *contenidas* (o *containerized*) comparten una misma capa de software subyacente, como puede verse en la Fig. 3.



Figura 3. Entornos virtuales clásicos vs. contenedores virtuales

Los principales proyectos en torno a los contenedores virtuales son Xen [12], LXC [11], Docker [13], KVM [14], OpenVZ [15], VMware ESX [16] y libvirt [17]. Existen también interesantes comparativas entre estas tecnologías como las llevadas a cabo por [18], [19] y [20].

Sin lugar a dudas, la principal aplicación de los contenedores virtuales es la distribución de servicios y aplicaciones. Sin embargo, su orientación se está ampliando

también a la ciencia e investigación con el principal objetivo de asegurar la calidad y reproducibilidad de la misma y de los cálculos numéricos que acarree cada proyecto. A modo de ejemplo, los autores de [21] examinan esta posibilidad para el caso de Docker y los de [22] para el de Xen. En este mismo sentido, desde el presente trabajo de investigación se propone también la idoneidad de los contenedores para asegurar la correcta, sencilla, uniforme y abierta distribución de recursos educativos, de los que hablaremos a continuación.

### C. Recursos educativos abiertos

Los Recursos Educativos Abiertos (REA u OER, por sus siglas en inglés) representan una opción única de democratización de la enseñanza y del aprendizaje. Estudiantes y profesores conviven en un marco de mutuo beneficio, donde ambos procesos se producen en ambas direcciones [23, 24]. Para que esto se produzca resulta indispensable establecer un hábito y unos medios de conectividad entre los miembros de las comunidades de prácticas o incluso las redes sociales utilizadas para integrar el aprendizaje formal con el informal [25]. Gran parte de una interconectividad útil y ágil se debe a la reutilización, a la catalogación y a la adaptación de los recursos y de los medios [26]. En este contexto, la mejora sustancial de la interactividad y de los medios de comunicación y relación entre los nodos de usuarios y recursos, de forma articulada pero no necesariamente organizada jerárquicamente, significa un gran avance hacia un proceso de implementación y uso de un sistema abierto de enseñanza y aprendizaje como el descrito [27]. Los contenedores virtuales juegan un rol fundamental porque permiten *atomizar* estos nodos y convertirlos en elementos versátiles al servicio del paradigma educativo.

### D. El Hub de Docker y su potencial como plataforma OERaaS

Una de las plataformas que más éxito está teniendo en el reciente ecosistema de los contenedores virtuales es Docker [28]. Docker funciona con el concepto de *imágenes* interconectables e interdependientes (Fig. 4), lo que lo dota de una gran flexibilidad y explica el éxito comercial que ya está teniendo en su todavía corta vida. Estas imágenes son *instantáneas* que, encajando como piezas de puzzle, conforman un entorno virtual operativo o *contenedor*. Cada contenedor incorpora por tanto el software, *frameworks* (bibliotecas, binarios, ficheros de configuración, *scripts* de apoyo, etc.) y configuración concreta para desempeñar una tarea concreta y sólo una.

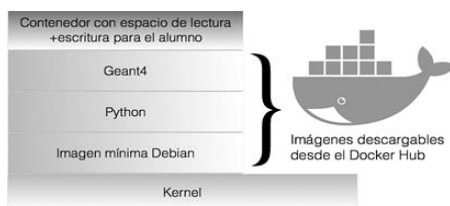


Figura 4. El Hub de Docker como posible OERaaS

Posibles ventajas técnicas aparte, Docker cuenta con un repositorio público, abierto y gratuito de imágenes donde se dispone de miles de instancias ya listas para desplegar y

asimilar a cualquier contenedor. Este repositorio recibe el nombre de *Docker Hub* (Fig. 5) o simplemente *Hub*. Cualquier usuario registrado (el registro es libre y no comporta regalía alguna) puede subir imágenes de manera totalmente gratuita al Hub y compartirlas con la comunidad de usuarios. Algunas de estas imágenes tienen un objetivo marcadamente académico pues están eminentemente destinadas a recrear entornos educativos concretos para multitud de áreas de conocimiento STEM. Por lo tanto, en cierta manera, el Hub se está comportando *de facto* como una arquitectura OERaaS, desde la cual se distribuyen y *sirven* cientos de recursos educativos.

Nombre de la imagen	STARS	PULLS
publicisworldwide/data-science-python public   automated build	0	85
gdhorne/genomic-data-science-toolbox public   automated build	2	42
calvingiles/data-science-environment public   automated build	1	383

Figura 5. El Hub de Docker como posible OERaaS

La Tabla I muestra algunas imágenes relacionadas con ciencia y educación y en qué repositorio (espacio dentro del Hub) se encuentran. Un repositorio suele estar indicado en dos partes: la primera hace referencia al autor de la imagen y la segunda a la imagen concreta (autor/imagen).

TABLA I  
ALGUNOS REPOSITARIOS DE IMÁGENES EDUCATIVAS DENTRO DEL HUB DE DOCKER

Repositorio Docker Hub	Descripción
bwawrik/bioinformaticas	Bioinformática con el lenguaje de programación Python
arkadi/mathics	Alternativa gratuita a Mathematica de Wolfram Research
sagemath/sagemath	Entorno de matemática numérica y simbólica
official/scratch	Famoso entorno de programación para niños del MIT
official/gazebo	Simulación de robots interactiva

Las imágenes pertenecientes al repositorio «oficial» son consideradas de gran relevancia en entornos concretos y suelen estar elaboradas por instituciones con una larga trayectoria. Tal es el caso de la imagen del famoso software Scratch [29] del MIT, en este caso en el terreno educativo.

### E. Trabajos relacionados

La virtualización y las máquinas virtuales tradicionales como herramientas educativas en STEM se vienen empleando desde prácticamente su nacimiento tecnológico. Normalmente, estos recursos educativos emulan un entorno de escritorio completo con todo lo necesario para que el alumno comience la realización de las tareas con la mayor facilidad de uso posible. La elección de máquinas virtuales viene sobretodo definida por el grado de complejidad del recurso a trabajar por parte del alumno/profesor en la actividad docente. Si este

recurso es complejo y necesita de una ardua configuración para su uso, una máquina virtual bien configurada parece ser una buena elección. Tal es el caso de GATE (*Geant4 Application for Tomographic Emission*) [30, 31] el cual tiene como objetivo la simulación numérica en imagen médica. GATE ofrece la descarga de una máquina virtual tradicional basada en la conocida distribución Ubuntu Linux (denominada vGATE) como las estudiadas en la Sección I-A con todo el software necesario para operar directamente con este framework.

De la misma manera, los autores de [32] han desarrollado un entorno educativo para astronomía (WWT) basado en máquinas virtuales. El proyecto CloVR tiene el mismo objetivo para la enseñanza de genética [33]. Los autores de [34] han investigado el uso de una máquina Virtualbox para la docencia química y los investigadores de [35] han realizado una acción similar para asignaturas de matemáticas. El acceso web a algunos de estos proyectos puede consultarse en la Tabla II. La Fig. 6 muestra algunas capturas de pantalla de estos entornos educativos.

El proyecto BioImg [36] tiene como finalidad centralizar un pequeño pero completo repositorio de imágenes de disco con distintos recursos educativos sobre la rama de biología. En este sentido se trata de un esfuerzo parecido al papel que desde el presente texto se reivindica para el Hub (Sección I-D).

TABLA II  
ALGUNOS ENTORNOS CIENTÍFICO-EDUCATIVOS QUE HACEN USO DE MÁQUINAS VIRTUALES CLÁSICAS

Proyecto	URL	Descripción
GATE	<a href="http://opengatecollaboration.org">opengatecollaboration.org</a>	Simulaciones numéricas en imagen médica y radioterapia
WWT	<a href="http://worldwidetelescope.org">worldwidetelescope.org</a>	Entorno para simulación de un telescopio virtual
CloVR	<a href="http://clovr.org">clovr.org</a>	Compendio de protocolos para análisis genético
BioImg	<a href="http://bioimg.org">bioimg.org</a>	Catálogo de máquinas virtuales orientadas a la enseñanza de bioinformática

Por último, mencionar que [2] explora los entornos formativos virtualizados desplegados en servicios en la nube.

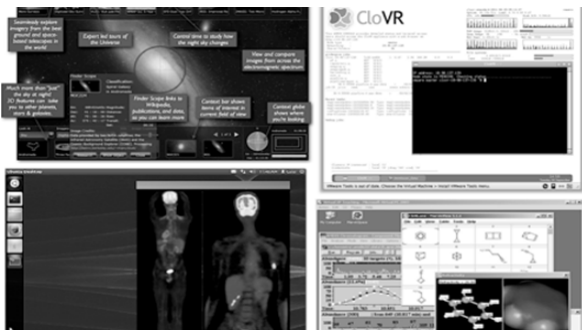


Figura 6. Algunos recursos educativos distribuidos como máquinas virtuales clásicas (WWT, CloVR, vGATE, etc.)

## II. METODOLOGÍA

Con el objetivo de comprobar en un entorno real las tecnologías y enfoques anteriormente presentados, se ha llevado a cabo una experiencia real durante el transcurso de la asignatura a distancia de *Fundamentos Físicos de la Informática* (~ 40 alumnos) perteneciente al primer curso del Grado en Ingeniería Informática [37] de la *Escuela Superior de Ingeniería y Tecnología* (ESIT) de la *Universidad Internacional de La Rioja* (UNIR). Al tratarse de una asignatura integrada en unos estudios superiores de ingeniería, desde el profesorado y coordinación docente se le ha querido dar un enfoque eminentemente aplicado. La metodología seguida durante la duración de la misma (un cuatrimestre) ha sido el estudio de herramientas informáticas de primer orden en torno a proyectos de actualidad comprendidos en experimentos de física moderna. Sin embargo, el compromiso con la ejecución de estas actividades es voluntario, dado que suelen comportar más tiempo y dedicación por parte del alumno. Es por ello que este bloque de ejercicios (Tabla III) recibe el nombre de *actividades alternativas* y si bien tienen cierto peso en la nota del curso, no son obligatorias.

La gran mayoría de estos proyectos de investigación (física de partículas, física de aceleradores, medicina nuclear, electromagnetismo, óptica, análisis de circuitos, etc.) requieren de entornos computacionales extremadamente específicos y difíciles de reproducir fuera del ámbito científico en el que fueron concebidos. En este sentido, si nos trasladamos a un entorno educativo se dan indudablemente las complejidades referidas en la introducción del presente trabajo [38, 39].

TABLA III  
ACTIVIDADES PROPUESTAS A LOS ALUMNOS MEDIANTE CONTENEDORES VIRTUALES

Actividad	Software	URL
Análisis de circuitos	NGSpice	<a href="http://ngspice.sf.net">http://ngspice.sf.net</a>
Matemática simbólica	Maxima	<a href="http://maxima.sf.net">http://maxima.sf.net</a>
Representación gráfica de funciones	GNUPlot	<a href="http://gnuplot.info">http://gnuplot.info</a>
Óptica geométrica	GNU Octave	<a href="http://octave.sf.net">http://octave.sf.net</a>
	OpenCV	<a href="http://opencv.org">http://opencv.org</a>
	Python	<a href="http://python.org">http://python.org</a>
Física de partículas	Geant4	<a href="http://geant4.web.cern.ch">http://geant4.web.cern.ch</a>
	Root	<a href="http://root.cern.ch">http://root.cern.ch</a>
Física cuántica	Ruby	<a href="https://www.ruby-lang.org">https://www.ruby-lang.org</a>
	Java	<a href="http://java.com">http://java.com</a>
Edición textos científicos	L <sup>A</sup> T <sub>E</sub> X	<a href="https://www.latex-project.org">https://www.latex-project.org</a>
	HTML5	<a href="https://www.w3.org">https://www.w3.org</a>
Medicina nuclear	DCMTK	<a href="http://dicom.offis.de">http://dicom.offis.de</a>
	ITK	<a href="http://www.itk.org">http://www.itk.org</a>
	VTK	<a href="http://www.vtk.org">http://www.vtk.org</a>
	C++	<a href="https://isocpp.org">https://isocpp.org</a>

En ediciones iniciales de esta misma asignatura, cada una de estas herramientas se distribuía de manera tradicional, es decir, mediante paquetes de software discretos a instalar por parte del alumno en su propio equipo informático. A pesar de la cuidada preparación por parte del profesorado de cada

herramienta a utilizar, siempre se daban problemas de compatibilidad y configuración en un número no despreciable de casos.

Por esta razón, en promociones siguientes se fue optando por virtualizar de manera clásica (entornos de escritorio completos como los descritos en la Sección I-A) ciertas actividades. La primera de ellas fue el laboratorio sobre física de partículas (Fig. 7). En aras de minimizar la huella computacional, esta máquina virtual no incluía un sistema de escritorio, pero sí un acceso SSH para poder establecer una comunicación bidireccional *host-guest* desde la cual se podían realizar los cálculos requeridos.

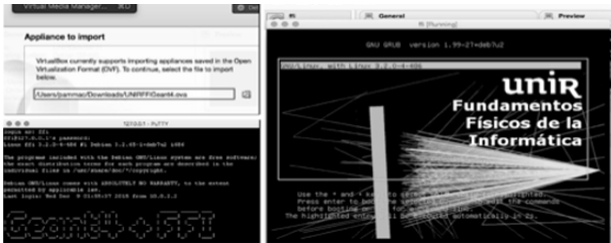


Figura 7. Máquina virtual clásica distribuida a los alumnos para la realización de la práctica sobre física de partículas. El alumno debía descargar un fichero OVF e importarlo en su propia instancia Oracle VirtualBox®

Sin embargo, en la promoción 2015-2016 se ha optado por transferir gran parte de estas actividades a contenedores virtuales basados en Docker. Esto ha supuesto una enorme simplificación del flujo de trabajo y la metodología, ya que por parte de los alumnos sólo se ha requerido la instalación del soporte Docker elemental (utilidades `boot2docker` o `dockermachine`, según versión). Una vez instalado, los alumnos han podido descargarse estos recursos desde la web del Hub o mediante la herramienta Kitematic (Fig. 8), desarrollada *ad hoc* por Docker con el objetivo de gestionar más cómodamente contenedores virtuales.



Figura 8. Kitematic mostrando las imágenes (y permitiendo la descarga pública y abierta) con todas las prácticas creadas para la asignatura de Fundamentos Físicos. Al mismo tiempo es posible ver una descripción de la funcionalidad de cada imagen y cómo trabajar con la misma

Deseamos insistir en el hecho de que estas imágenes han estado públicamente disponibles en el Hub, de manera que podían ser descargadas y utilizadas por cualquier alumno (o persona interesada), independientemente de sus institución de origen. Se trataban por tanto de auténticos OERs y el Hub asumió temporalmente las funciones de OERaaS.

Por parte del equipo docente, tan sólo fue necesario crear estas imágenes y desplegarlas en el Hub. Para crear imágenes, el equipo de Docker ha ingeniado la sintaxis *Dockerfile*. Con este mecanismo podemos indicar de manera sencilla cómo se construye cada imagen y de qué otras imágenes depende.

### III. RESULTADOS

La transición hacia el uso de contenedores virtuales ha conllevado una gran batería de ventajas de cara al alumno. Para empezar, los pasos dados hacia la simplificación del despliegue de cada componente de software ha invitado a muchos estudiantes a la realización de las actividades alternativas, en comparación con promociones anteriores (Fig. 9). Asimismo, el nivel de satisfacción con estas actividades y su facilidad de despliegue es bastante relevante.

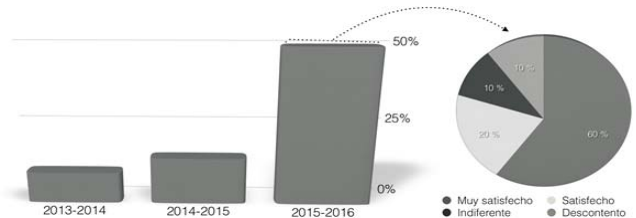


Figura 9: Evolución desde la promoción inicial hasta la actual en el compromiso y entrega con actividades alternativas basadas en el uso de herramientas de software científico. A la derecha se muestra el grado de satisfacción del alumnado con la tecnología de contenedores virtuales (desplegada en la última promoción) para la ejecución de las prácticas.

De igual manera, durante esta última promoción, el interés ha ido aumentando a lo largo del cuatrimestre, con un especial incremento en la semana 6 (Fig. 10). En esta semana se propuso a los estudiantes la misma práctica relacionada con física de partículas (descrita en la Sección II). En esta actividad el alumno debe simular un haz de partículas y sus posibles interacciones con la materia y detectores. El interés y curiosidad que despierta la tarea por sí misma, en conjunción con la facilidad de puesta en marcha gracias a los contenedores virtuales, atrajo a un gran número de estudiantes hacia la realización de actividades alternativas para el resto de duración de la asignatura.

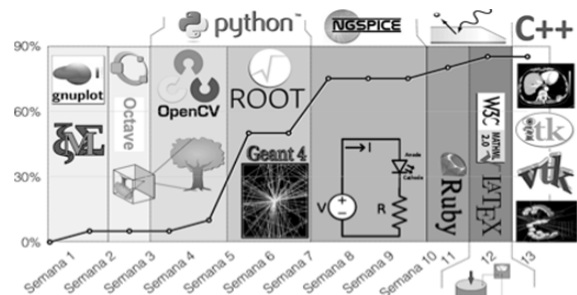


Figura 10: Evolución semana a semana del porcentaje de alumnos comprometidos con la ejecución de actividades basadas en contenedores virtuales por parte del alumno.

#### IV. CONCLUSIONES

Los contenedores virtuales representan un potente instrumento de distribución de OER. En este artículo nos hemos centrado especialmente en el proyecto Docker y en su plataforma Hub para la distribución de imágenes de contenedores virtuales. Se ha demostrado mediante un caso práctico real cómo esta herramienta puede operar a la manera de *distribuidora de recursos educativos* lo que nos ha permitido introducir el concepto de OERaaS (*OERs as a Service*). Durante el mencionado caso práctico se percibió un progresivo aumento en el interés y compromiso por parte de los alumnos con el uso de las herramientas educativas propuestas. La simplificación de la distribución de entornos informáticos para educación es una pieza clave para atraer a los alumnos hacia el uso de herramienta STEM modernas y de alta complejidad.

#### AGRADECIMIENTOS

This research is partially funded by UNIR Research (<http://research.unir.net>), Universidad Internacional de La Rioja (UNIR, <http://www.unir.net>), under the Research Support Strategy 3 (2015-2017), Research Group TELSOCK.

#### REFERENCIAS

- [1] D. A. Menascé, "Virtualization: Concepts, applications, and performance modeling," in *Int. CMG Conference*, 2005.
- [2] N. Mehic and E. Kashfi, "Education in clouds," 2011.
- [3] R. Uhlig, G. Neiger, D. Rodgers, A. L. Santoni, F. Martins, A. V. Anderson, S. M. Bennett, A. Kági, F. H. Leung, and L. Smith, "Intel virtualization technology," *IEEE Computer*, 2005.
- [4] R. Soares Boaventura, K. Yamanaka, and G. Prado Oliveira, "Performance analysis of algorithms for virtualized environments on cloud computing," *IEEE Latin America Transactions*, 2014.
- [5] K.-T. Seo, H. Hwang, I. Moon, O. Kwon, and B. Kim, "Performance comparison analysis of linux container and virtual machine for building cloud," *Advanced Science and Technology Letters*, 2014.
- [6] VMware, "SQL Server workload consolidation," Tech. Rep., 2008.
- [7] OCCI-WG, "Open cloud computing interface-infrastructure models," Open Grid Forum, Tech. Rep., 2010.
- [8] J. Bruce, "Desktop virtualization in K-12 schools," Center for Digital Education, Tech. Rep., 2010.
- [9] IBM, "Virtualization in education," Tech. Rep., 2007.
- [10] K. matematyki and W. Nauczycielski, "Desktop virtualization as a modern solution in education," in *Ročník Národnej Konferencie*, 2011.
- [11] R. Rosen, "Linux containers and the future cloud," *Linux Journal*, 2014.
- [12] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," *ACM SIGOPS Operating Systems Review*, 2003.
- [13] D. Liu and L. Zhao, "The research and implementation of cloud computing platform based on docker," in *11th International Computer Conference on Wavelet Active Media Technology and Information Processing*, 2014.
- [14] A. Kivity, Y. Kamay, D. Laor, U. Lublin, and A. Liguori, "KVM: the Linux virtual machine monitor," *Linux symposium*, 2007.
- [15] K. Kolyshkin, "Virtualization in linux," *White paper, OpenVZ*, 2006.
- [16] A. Muller and S. Wilson, "Virtualization with VMware ESX server," 2005.
- [17] M. Bolte, M. Sievers, G. Birkenheuer, O. Niehörster, and A. Brinkmann, "Non-intrusive virtualization management using libvirt," in *Proceedings of the Conference on Design, Automation and Test in Europe*, 2010.
- [18] T. Deshane, Z. Shepherd, J. Matthews, M. Ben-Yehuda, A. Shah, and B. Rao, "Quantitative comparison of xen and kvm," *Xen Summit, Boston, MA, USA*, 2008.
- [19] J. Che, Y. Yu, C. Shi, and W. Lin, "A synthetical performance evaluation of OpenVZ, Xen and KVM," in *Services Computing Conference (APSCC), 2010 IEEE Asia-Pacific*, 2010.
- [20] C. Fragni, M. D. Moreira, D. M. Mattos, L. H. M. Costa, and O. C. M. Duarte, "Evaluating Xen, VMware, and OpenVZ virtualization platforms for network virtualization," *Universidade Federal Rio de Janeiro*, 2010.
- [21] C. Boettiger, "An introduction to Docker for reproducible research," *ACM SIGOPS Operating Systems Review*, 2015.
- [22] B. Clark, T. Deshane, E. M. Dow, S. Evanchik, M. Finlayson, J. Heme, and J. N. Matthews, "Xen and the art of repeated research," in *USENIX Annual Technical Conference*, 2004.
- [23] P. Albright, "Open educational resources open content for higher education," in *Internet Discussions Forums*, 2005.
- [24] N. Pearce, M. Weller, E. Scanlon, and S. Kinsley, "Digital scholarship considered: How new technologies could transform academic work," in *education*, vol. 16, no. 1, 2012.
- [25] M. Weller, *Battle for Open: How openness won and why it doesn't feel like victory*. plus 0.5em minus 0.4em Ubiquity Press] u [, 2014.
- [26] G. Conole, *Designing for learning in an open world*. Springer Science & Business Media, 2012, vol. 4.
- [27] D. Burgos, "The structure and behavior of virtual communities engaged in informal learning about e-learning standards," Master's thesis, European University of Madrid, 2006.
- [28] V. Tuomas, "Advantages of docker," Ph.D. dissertation, University of Jyväskylä, 2015.
- [29] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman *et al.*, "Scratch: programming for all," *Communications of the ACM*, vol. 52, no. 11, pp. 60–67, 2009.
- [30] S. Jan, G. Santin, D. Strul, S. Staelens, K. Assie, D. Autret, S. Avner, R. Barbier, M. Bardies, P. Bloomfield *et al.*, "Gate: a simulation toolkit for pet and spect," *Physics in medicine and biology*, 2004.
- [31] D. Strulab, G. Santin, D. Lazaro, V. Breton, and C. Morel, "Gate (geant4 application for tomographic emission): a pet/spect general-purpose simulation platform," *Nuclear Physics B-Proceedings*, 2003.
- [32] A. Goodman, J. Fay, A. Muench, A. Pepe, P. Udomprasert, and C. Wong, "Worldwide telescope in research and education," *arXiv preprint arXiv:1201.1285*, 2012.
- [33] S. V. Angiuoli, M. Matalka, A. Gussman, K. Galens, M. Vangala, D. R. Riley, C. Arze, J. R. White, O. White, and W. F. Fricke, "CloVR: a virtual machine for automated and portable sequence analysis from the desktop using cloud computing," *BMC bioinformatics*, 2011.
- [34] T. Kind, T. Leamy, J. A. Leary, and O. Fiehn, "Software platform virtualization in chemistry research and university teaching," *Journal of Cheminformatics*, 2009.
- [35] T. Hamada and K. committers, "Knoppix/math: A live system for enjoying mathematics with computer," *ACM Commun. Comput. Algebra*, 2009.
- [36] M. Dahlö, F. Haziza, A. Kallio, E. Korpelainen, E. Bongcam-Rudloff, and O. Spjuth, "Bioimg.org: A catalog of virtual machine images for the life sciences," *Bioinformatics and Biology Insights*, 2015.
- [37] ESIT. [Online]. Available: <http://ingenieria.unir.net>
- [38] A. Madhavapeddy, R. Mortier, C. Rotsos, D. Scott, B. Singh, T. Gazagnaire, S. Smith, S. Hand, and J. Crowcroft, "Unikernels: Library operating systems for the cloud," in *ACM SIGPLAN Notices*, 2013.
- [39] V.A. Secades, O. Arranz, "Big Data & eLearning: A Binomial to the Future of the Knowledge Society," in *International Journal of Interactive Multimidia and Artificial Intelligecem*, 3(6), pp. 29-33, doi: 10.9781/ijimai.2016.364, 2016.



**Alberto Corbí** trabaja como investigador del Grupo TELSOCK, sobre Tecnología Educativa y Redes Sociales, así como profesor en la Escuela Superior de Ingeniería y Tecnología, ambos en la Universidad Internacional de La Rioja (UNIR). Ha estudiado Físicas y posee experiencia en Informática y Educación. Actualmente, su investigación se centra en sistemas recomendadores, especificaciones eLearning y en interoperabilidad. Paralelamente, es estudiando de doctorado con una tesis sobre imagen médica en el Centro Superior de Investigaciones Científicas de España (CSIC).



**Prof. Dr. Daniel Burgos** trabaja como Vicerrector de Transferencia y Tecnología, Director de la Cátedra UNESCO en eLearning y de la Cátedra ICDE en Recursos Educativos Abiertos, en la Universidad Internacional de La Rioja (UNIR, <http://research.unir.net>). Previamente, ha sido Vicerrector de Investigación y Director de la Escuela de Ingeniería en la misma universidad, así como Director del Sector de Educación y del Laboratorio de Experiencia de Usuario en el departamento de Investigación e Innovación de la multinacional Atos. También trabajó como profesor-investigador en la Open University de los Países Bajos. Ha participado en más de 40 proyectos de I+D+i con financiación pública europea. Es doctor en Ciencias de la Comunicación (Univ. Europea), Ingeniería Informática (Univ. Carlos III), Educación (Univ. Barcelona) y Antropología (Univ. Westminster).