Transgenic learning for STEAM subjects and virtual containers for OER

Daniel Burgos & Alberto Corbí

To cite this article: Daniel Burgos & Alberto Corbí (2018): Transgenic learning for STEAM subjects and virtual containers for OER, Distance Education, DOI: 10.1080/01587919.2018.1429894

To link to this article: https://doi.org/10.1080/01587919.2018.1429894

Published online: 29 Jan 2018.

Article views: 6

View related articles

View Crossmark data
Transgenic learning for STEAM subjects and virtual containers for OER

Daniel Burgos and Alberto Corbí

UNIR Research, Universidad Internacional de La Rioja (UNIR), Madrid, Spain

ABSTRACT

Transgenic learning is a disruptive approach in education. It encourages modification of moving parts of the educational chain. This article provides a view of transgenic learning focused on the delivery of enriched learning contents in STEAM areas. It discusses the mutagenic role that the virtual containers may play in current distance education. We focus on the containers’ technology and how it can bridge students, computing resources, teachers and specific IT needs. We also present an experiment carried out at UNIR University where we describe the transition from using conventional software distribution methods to the use of containers. Thanks to this virtualization approach, it is possible to deliver the necessary software-based lab scenarios. The results show that the participation and satisfaction of the students increased over time. Our experience shows that the combination of open educational resources, containers, and modern distribution channels can play a significant role in STEAM education.

Disruptive learning, transgenic learning

Genetically modified organisms (GMO) is a controversial technique to produce new life or food based on the artificial modification of DNA (Apolinario, 2015; Burton, Rigby, Young, & James, 2001; Millis, 2006). Induced by an external disruption, a significant change happens, as if it might be part of the natural evolution of a species. In doing so, adaptation is forced into the natural course, so that an additional feature is provided to that species: from a stronger plant against stormy weather or a plague, to a vitamin embedded into a cereal that does not contain it by default, through the modification of a human protein. This external intervention is conflictive from a number of approaches: ethical, scientific, societal and economic, to name a few. However, the possibility exists; and if smartly applied, it provides the human being with a new resource for progress.

Education, as a whole, nowadays, requires a disruptive boost (Collins & Halverson, 2010; Wrigley, 2009). If we teach and learn in the same way that we did for the last 20 centuries; if we use the very same academic structures that we did 10 centuries ago; if we stress some methodologies from the early years of the last century; and if we use resources from before the rise of the Internet; if all this happens, we will miss every single possibility that the last
20 years bring to the table. We will miss new, adapted, personalized ways to learn and to teach; to be more efficient, to get a better performance; to enjoy more the experience as a user; and to improve the competence and skill acquisition. Furthermore, we need to break this slow evolution in education. The young people, the technicians, the mass media, the entertainment industry, all of them are far advanced from any practical implementation in the classrooms, from kindergarten to the University.

There are open educational resources (OER), MOOCs, Virtual Reality, Augmented Reality, Emotional Intelligence, Personalized Learning, Analytics and so many resources, services and approaches to complement, enhance and evolve education, as it is now (McGreal, Kinuthia, Marshall, & McNamara, 2013; New Media Consortium, & EDUCAUSE Learning Initiative, 2015). We need a radical innovation, to design a new paradigm, to complement the existing ones, to evolve with the actual users of the system (students, teachers, professors, tutors, parents) and not always far behind from them. We need a GMO concept into learning and teaching, a transgenic approach to education. Something that makes things evolve quicker and more adapted into a very specific and practical objective.

And out of this challenge, the most difficult part is to find the right integration between informal ways of learning, teaching and using daily services, with formal courses and academic degrees; the smart combination of resources inside-outside the classroom; the update of accredited content with enriched, additional information outside the official syllabus that can fit into the same slot of educational competences (Dabbagh & Kitsantas, 2012; De-la-Fuente-Valentín, Carrasco, Konya, & Burgos, 2013).

In the context of our research, STEAM education and specifically in distant STEAM subjects and activities (such as remote labs), we need a novel GMO approach to fight against their limitations (discussed next). Our transgenic solution is based on modern and state-of-the-art virtualization technologies (i.e., virtual containers) and the possibility of freely and openly sharing virtual computing environments. In the work described in this text, we examine a gene (or set of genes) in the educational gnome that does not quite work (remote STEAM labs and their open delivery as educational resources) and we propose a new genetic variation that does the job.

**Limits of OER in STEAM**

As part of the transgenic approach, OER are becoming a trend in the academic field at the university level, as summarized by Zancanaro, Todesco, and Ramos (2015) and Hatzipanagou and Gregson (2015). Many renowned academic institutions are already distributing digital content for free, such as California State University (http://als.csuprojects.org), MIT OpenCourseWare (http://ocw.mit.edu), Washington State University (https://teach.wsu.edu/oer) and Tufts OpenCourseWare (http://ocw.tufts.edu)

In addition, non-profit institutions are also offering free OER materials, such as OpenStax (http://cnx.org) and The Open University (http://www.open.ac.uk)

Furthermore, there are associations whose objective is to collect OER with creators, collaborators, students and OER consumers, such as Open Education Consortium (https://oer-consortium.org), EDUCASE (https://library.educause.edu), OERu (https://oeru.org), OER Foundation (http://wikieducator.org/WikiEducator:OER_Foundation) and OERCommons (https://www.oercommons.org).
As stated by Albright (2005) and Pearce, Weller, Scanlon, and Kinsley (2012), OER have contributed to the democratization of education by allowing students and teachers to coexist in a context of mutual benefit. The non-hierarchical relations that arise between OER students, OER creators and teachers help to improve formal and informal learning contexts, interpersonal interactions and the educational process in general.

An OER generally comprises teaching, learning, research or even electronic/multimedia resources. These resources are available for free, for the most part, and with very unrestrictive licensing. Within the contents of an OER, the student can find any material designed for educational purposes (i.e., textbooks, lectures, interactive simulations, games, competitions, evaluation software, etc.).

In the context of computer science and engineering teaching, there are already many open source software initiatives. However, these cannot strictly be considered OER, since they usually do not come with instructions or teaching guides (i.e., proposed activities, starting tutorials, automatic self-correcting tools, etc.). For example, open software repositories like GitHub or SourceForge and their code, despite being de facto open resources, cannot be regarded as OER because they do not necessarily aspire to play an educational role.

In addition, the implementation of traditional educational software in institutions can be a tedious task, which is normally delegated to teachers, students or other members of staff who are not technicians. This difficulty arises because of the wide range of computer systems and architectures in such institutions. In the case of inter-institutional collaboration, the situation worsens, as discussed in detail by Nerantzi (2012). The lack of documentation on handling such content adds an additional challenge. OER, as with any type of learning material based on software, normally requires that the student not only be able to access these digital resources, but also deal with their installation, configuration and proper use.

In the face-to-face teaching context, the process just described can be carried out with the means and resources of the educational institution, and the student can ask for help when necessary. However, in distance learning, the student feels isolated and somehow alone in facing any type of technical or learning difficulties (Collins & Halverson, 2010). This isolation may be more intense in the specific case of OER, for which students have no institutional or official contact with the creator of the learning resource and, therefore, cannot request any type of support.

In some cases, the installation of the required tools is difficult. Other learning content also requires very sophisticated computing environments, such as:

- Specific operating system (OS) and versions
- Pre-installed libraries, frameworks and runtimes, such as Python, Java, .NET, etc.
- Specific user permits or administrator rights in order to install and run software
- Specific hardware: processors, amounts of memory, GPU resources, etc.
- In these situations, the only alternatives are:
  - To require the student to acquire or replicate the software architectures and necessary conditions for the activities and contents provided.
  - To allow remote access by the students to a controlled working environment, deployed by the institution or a third party (e.g., hostings based in the cloud, such as Microsoft Azure, Salesforce Heroku or Red Hat OpenShift).
• To limit the underlying technologies necessary for the learning resource to those that have a broad consensus and adoption (e.g., international standards, such as W3C and HTML5, ECMA and ISO, or C# and C++.
• To virtualize each of the work environments through so-called virtualization technologies, which are addressed in this article.

From our point of view, of all the options mentioned, the only one that can be successful in the vast majority of scenarios is, without doubt, virtualization, as justified below. It is currently generally considered that cloud solutions are the best response to the problems mentioned above (Sultan, 2010). However, in order to run some applications in a shared time/remote environment, significant computational resources are required and performance may decrease significantly in these cloud scenarios. In this article, we propose emerging virtual containers (which are discussed in detail in Section ‘State of the art of virtualization of e-learning tools and OER’) as a vehicle for the distribution of complex educational resources. We have studied a specific application of these containers called Docker.

A great part of Docker’s attractiveness lies in the management of an openly accessible repository of virtualized environments or images. This repository behaves as an OER management tool and can be considered as an OER as a service (OERaaS) platform, prepared for use in distance learning scenarios. This interesting feature will be addressed in Section ‘Virtual containers’, but we will first discuss the contributions of classical virtualization technologies to e-learning.

State of the art of virtualization of e-learning tools and OER

Virtualization, in its traditional or classic conception, is the implementation of a partial or complete hardware element or system exclusively through software. These elements can be disks, processors, network infrastructure, desktop or peripheral computers (including graphics, multimedia and sound reproduction). The virtualized resources are controlled by a hypervisor, which adds a layer of abstraction between the real hardware (host) and the virtual scenario (guest). The main companies involved in classical virtualization are Oracle, Parallels, VMware and Citrix. The major counterpart open source projects are the famous VirtualBox and QEMU.

Despite being virtualized environments, the efficiency and versatility of modern commercial and open/non-commercial solutions are virtually equal to those obtained through the host systems (Seo, Hwang, Moon, Kwon, & Kim, 2014).

This type of virtualization comes with a big disadvantage: each time a new activity or content is created (educational), a completely new virtual machine must be distributed. This usually implies the need for greater bandwidth for upload and download on the part of both the student and the institution. It may also imply decreased performance when running several exercises, and hence several virtual machines, simultaneously. This disadvantage can be overcome by way of virtual containers, which are discussed below in Section ‘State of the art of virtualization of e-learning tools and OER’.

Each new virtual machine is distributed using the standards and specific formats (OVF, VDI, VDMK, etc.) agreed by renowned companies and projects that are part of the Open Grid Forum (OCCI-WG, 2010).
With regard to the application of virtual machines in education, we find interesting the study carried out by Bruce (2010), who discusses the barriers to virtualization’s entry into schools. These barriers are mainly:

- Lack of skills and knowledge on the part of teachers and students
- Lack of resources on the part of educational centres and institutions
- Disagreements between actors in the educational system.

On the other hand, IBM (2007) predicts an unstoppable increase in virtualization technology in all areas and recommends that schools’ IT services join this trend. Finally, other research groups focusing on education technologies, such as Nauczycielski (2011), have conducted comprehensive analyses of the existing virtualization technology and its application to the teaching of subjects related to the creation of networks. Also, various educational institutions, have described the successful development of remote courses thanks to the concept of virtualization. Bližňák, Vojtěšek, Matušů, and Dulík (2008), Cronin, Pauli, and Ham (2013), Li (2009) and Toderick, Peng, and Lunsford (2009) are just a few examples. Theses authors strongly agree on that virtualization frees students from the restraints of having to attend a physical place in order to attend a course or a lab. With some limitations, discussed below, traditional virtualization already represents a powerful transgenic shift. Traditional virtualization and virtual machines, including STEAM educational tools, have been used since their technological birth. Generally, these educational resources emulate complete desktop environments with everything necessary for the student to begin to solve tasks with the greatest possible ease of use. The choice of virtual machine has been primarily influenced by the degree of complexity of the resource with which the student will be working. If complex configuration is required in order to use a resource, a properly configured virtual machine seems the best option. This is so, for instance, in the case of the Geant4 application for tomographic emission (GATE). GATE offers the download of a traditional virtual machine based on the popular Ubuntu Linux system, including the software necessary to operate directly with this framework.

Likewise, Goodman et al. (2012) have developed a learning environment based on the virtual machine for astronomy (WWT). The CloVR project (Angiuoli et al., 2011) shares the same objectives, but focuses on the teaching of genetics. Kind, Leamy, Leary, and Fiehn (2009) have launched a VirtualBox virtual machine for the teaching of chemistry and Hamada (2009) has done something similar for mathematics. The BioImg project (Dahlö et al., 2015) aims to centralize a complete repository of existing virtual machines with learning resources on the teaching of biology. The websites of these projects can be found in Table 1. Likewise, Figure 1 shows some images of these learning environments.

<table>
<thead>
<tr>
<th>Project</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GATE</td>
<td>opengatecollaboration.org</td>
<td>Simulations in physics and medical radiation therapy</td>
</tr>
<tr>
<td>WWT</td>
<td>worldwidetelescope.org</td>
<td>Environment to work with a virtual telescope</td>
</tr>
<tr>
<td>CloVR</td>
<td>clovr.org</td>
<td>Genetic analysis</td>
</tr>
<tr>
<td>BioImg</td>
<td>bioimg.org</td>
<td>Bioinformatics exercises</td>
</tr>
</tbody>
</table>
Virtual containers

Virtual containers (Rosen, 2014) may be considered light virtual machines, which are normally based on a GNU/Linux shared system. They are designed to run an instance of a specific application (and not a typical virtual scenario, complete with screen, desktop and varied applications). A container seeks, generally, to implement a web service: a Ruby on Rails, Node or PHP application that exposes a TCP/IP interactive port. This aim is achieved by running a virtual machine that implements all the necessary software.

Such containers are becoming major allies to programmers, systems administrators and DevOps (development and operations) professionals because they can be easily implemented in any IT infrastructure with minimum pre-installation support. Their main advantage is their lightness and their ability to work in both development and production scenarios. They differ from traditional virtual machines in that all contained (or containerized) applications share the same underlying software layer.

The main projects on virtual containers are Xen (Barham et al., 2003), LXC (Rosen, 2014), Docker (Liu & Zhao, 2014), KVM (Kivity, Kamay, Laor, Lublin, & Liguori, 2007), OpenVZ (Kolyshkin, 2006), VMware ESX (Muller & Wilson, 2005) and libvirt (Bolte, Sievers, Birkenheuer, Niehörrster, & Brinkmann, 2010). Some interesting comparisons between these technologies have also been performed by Deshane et al. (2008), Che, Yu, Shi, and Lin (2010) and Fragni, Moreira, Mattos, Costa, and Duarte (2010).
Virtual containers have, in recent years, come to fill an important niche in the administration of systems (Rosen, 2014). Container technology is currently considered the best solution to the problem of how to ensure that software runs reliably when it is transferred from one computer environment to another. A container is a complete runtime bundle package, including the target application and all its units (i.e., linked or static libraries, help programs, public or configuration files, etc.). The containerization of the application and its units eliminates differences in the underlying infrastructure. In contrast to conventional virtualization technologies (like VMware Fusion, Parallels Desktop, Oracle VirtualBox, etc.), various container applications share a core individual OS. This means they are lighter and consume fewer resources than conventional virtual machines, and allows for the large-scale distribution of educational environments with a few dozen megabytes, or even less. As a result, a container can easily be run on the hardware of the local user/student’s device, or on cheaper commercial cloud infrastructure (Joy, 2015).

Recently, many projects have emerged related to the core technologies of containers, in which numerous computer engineering companies, communities and associations are participating, from the largest to the smallest. KVM from Open Virtualization Alliance, ESX from VMware and Docker from dotCloud are only some examples (Che et al., 2010).

Just like others in this group, Docker implements a simple high-level interface in order to provide light virtual environments that run isolated processes. However, Docker has a key advantage over other options, called the Hub. The Docker Hub is an online service (registration is free) for the distribution of containers (Figure 2). It also provides search tools for the discovery and management of containers, and team collaboration (Hagstrom & Essary, 2009). As we suggest later in Section ‘Virtual containers’, the Docker Hub can implicitly behave as a service for the distribution of OER, and so can be considered an OERaaS platform.

Figure 2. The website of Docker Hub.
Without doubt, the main application of virtual containers is the distribution of services and applications. However, these containers are also attracting the attention of science research groups who consider them as a means to ensure the reproducibility of experimental results. For example, Boettiger (2015) investigates this possibility in the case of Docker, and Clark et al. (2004) do the same with regard to Xen. However, their application has been neglected in academia, particularly as a resource for distance education. In this context, it is proposed that Docker containers are appropriate for ensuring the proper, simple, uniform and open distribution of educational content. In particular, Docker and its Hub are suggested for use in the distribution of open learning resources. In our opinion, these 2 features (lightweight virtual computing environments together with their open distribution) represent a needed genetic mutation in the context of the distribution and execution of remote STEAM labs. This transgenic improvement is even more significant and powerful than the one already characterized by the classic virtualization technologies and efforts tackled in Section ‘State of the art of virtualization of e-learning tools and OER’.

The Docker Hub as distribution platform for OER

One of the most successful platforms in the virtual containers ecosystem is Docker (Tuomas, 2015). This virtual container works with the concept of interconnected and interdependent images, which endows it with great flexibility and explains the commercial success that it has already enjoyed in its short life thus far. These images are snapshots of containers, which fit together like pieces of a puzzle, forming a virtual operating environment. Each container incorporates only the framework (libraries, binaries, configuration files, support scripts, etc.), specific configuration files and software necessary to perform a task.

Docker has a public repository of open and free containers, which makes thousands of these snapshots available for download and implementation. This repository is called Docker Hub (Figure 4) or, simply, the Hub. Any registered user (registration is free and does not imply any royalties) can upload images to the Hub and share them with a vast and growing community of users. Some of these images have clearly academic goals, as they are designed to recreate specific educational environments for many STEAM subject knowledge areas. Therefore, the Hub behaves as a de facto OERaaS platform, from which hundreds of educational resources are distributed and delivered every day.

Docker and the Docker hub have already been hosting thousands of containers aimed at educational purposes. Recently, two programs have been announced to tighten the links with the educational community: the Docker Student Kit, the Docker Campus Ambassador Program and the Docker in Higher Education Community Directory.

Table 2 lists some Docker images related to science and education, along with the repository (within the Hub) in which they can be found. A repository name is usually presented

<table>
<thead>
<tr>
<th>Hub repository</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bwawrik/bioinformatics</td>
<td>Bioinformatics with the Python programming language</td>
</tr>
<tr>
<td>arkadi/mathics</td>
<td>Mathematics alternative to Wolfram Research</td>
</tr>
<tr>
<td>sagemath/sagemath</td>
<td>Environment of numerical and symbolic mathematics</td>
</tr>
<tr>
<td>official/scratch</td>
<td>MIT environment to teach programming to children</td>
</tr>
<tr>
<td>official/gazebo</td>
<td>Interactive robots simulation</td>
</tr>
</tbody>
</table>

Table 2. Some container image repositories associated with education in Docker Hub.
in the author/image format, for which the first element refers to the author of the image and the second to the image itself.

The images that belong to the official repository are considered to be of great relevance in specific environments, and are generally developed, maintained and uploaded by institutions with a long tradition. This is the case for the famous Scratch software (Resnick et al., 2009) in the field of education, developed and maintained by MIT.

As stated above, this open distribution of containers in the Docker Hub is the second genetic improvement presented in this work. It solves one of the main difficulties in the realm of OER sharing: the free, bandwidth-efficient distribution of STEAM-related materials and activities.

A practical case study on transgenic learning and virtual containers

As a practical example, we propose the experiments carried out in the School of Engineering of the Universidad Internacional de La Rioja (UNIR). UNIR is a young 100% online university, with headquarters in Spain, and premises in five Latin-American countries and in USA. It counts 26,000 students, 1,000 faculty, and 500 support staff (including 200 coaches-tutors), with a strong research force focused on Educational Technology and Innovation. A regular student profile is between thirty-five and forty-five years old, has a family and works full time.

The example presented in this paper describes, in the context of the subject of physics for software engineers at UNIR, a transition from the use of conventional methods of software distribution to the use of Docker containers for the dissemination of software for each task. Each set of tasks was delivered to the students as a virtual container. These tasks ranged from simple physics problems that had to be solved with a set of Python scripts, to more complex simulation scenarios that require complex reality software. The latter was the case for an exercise related to the study of subatomic particles and photon collisions/circuits, calculated with the Geant4 package developed by CERN. All of the containers implemented the necessary software tools for each physics task, along with examples and complementary solutions manuals, freeing the students from royalties and allowing them to focus on how to solve the problem. All the containers (and all of the proposed activities) shared common resources, such as the kernel, core libraries and Python environment.

Our reason for choosing the container technology and the Docker Hub, was the low amount of student participation in remote labs that required complex computing environments for physics simulations. After betting on these two tools, the scenario changed completely (when compared against former semesters), as is shown in Section ‘Results and discussion’.

The physics subject mentioned has an essentially applied focus. The implemented methodology consisted of the analysis of the main software tools in projects that currently are part of modern physics experiments, along with a replication of those experiments in a virtual form and a discussion phase to share and compare the experience, and assess the actual disruption along the learning flow. Participation in these activities was voluntary, since they generally call for more time and commitment from the student. For this reason, this block of exercises (Table 3) has been called alternative activities and, although they have a certain weight in the scoring of the course, their execution and resolution is not compulsory. In the first editions of the physics subject, each of these tools was distributed in a more
traditional manner, i.e., through discrete packages of software for each OS that students had to install on their own computers. Despite the careful preparation of each tool, in some cases compatibility and configuration issues emerged. For this reason, in subsequent years, the use of classic virtual machines (full desktop scenarios) was preferred for certain activities. One of these activities was the particle physics laboratory. To minimize the complexity associated with the implementation and execution of this exercise, a headless virtual machine was created (lacking both a desktop system and graphical interface). The students downloaded the machine, which they could access through an SSH session. This session, in turn, allowed for the execution of the necessary calculations (Figure 3). The only drawback to this method of distribution – that when the particle physics task required some, even minor, change, a completely new virtual image had to be rebuilt from scratch – was addressed in Section ‘Limits of open educational resources in STEAM’.

![Figure 3](image_url). The classic virtual machine distributed to the students to carry out an experiment in particle physics. The students had to download a 2 GB OVF file and import it into their own Oracle VirtualBox.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Software</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuits analysis</td>
<td>Ngspice</td>
<td><a href="http://ngspice.sf.net">http://ngspice.sf.net</a></td>
</tr>
<tr>
<td>Symbolic mathematics</td>
<td>Maxima</td>
<td><a href="http://maxima.sf.net">http://maxima.sf.net</a></td>
</tr>
<tr>
<td>Delineation and mapping functions</td>
<td>Gnuplot</td>
<td><a href="http://gnuplot.info">http://gnuplot.info</a></td>
</tr>
<tr>
<td>Optics</td>
<td>GNU Octave</td>
<td><a href="http://octave.sf.net">http://octave.sf.net</a></td>
</tr>
<tr>
<td></td>
<td>OpenCV</td>
<td><a href="http://opencv.org">http://opencv.org</a></td>
</tr>
<tr>
<td></td>
<td>Python</td>
<td><a href="http://python.org">http://python.org</a></td>
</tr>
<tr>
<td></td>
<td>Geant4</td>
<td><a href="http://geant4.web.cern.ch">http://geant4.web.cern.ch</a></td>
</tr>
<tr>
<td></td>
<td>Root</td>
<td><a href="http://root.cern.ch">http://root.cern.ch</a></td>
</tr>
<tr>
<td>Quantum physics</td>
<td>Ruby</td>
<td><a href="https://www.ruby-lang.org">https://www.ruby-lang.org</a></td>
</tr>
<tr>
<td></td>
<td>Java</td>
<td><a href="http://java.com">http://java.com</a></td>
</tr>
<tr>
<td>Text processing of scientific documents</td>
<td>LaTeX</td>
<td><a href="https://www.latex-project.org">https://www.latex-project.org</a></td>
</tr>
<tr>
<td></td>
<td>HTML5</td>
<td><a href="https://www.w3.org">https://www.w3.org</a></td>
</tr>
<tr>
<td>Medical physics</td>
<td>DCMTK</td>
<td><a href="http://dicom.offis.de">http://dicom.offis.de</a></td>
</tr>
<tr>
<td></td>
<td>ITK</td>
<td><a href="http://www.itk.org">http://www.itk.org</a></td>
</tr>
<tr>
<td></td>
<td>VTK</td>
<td><a href="http://www.vtk.org">http://www.vtk.org</a></td>
</tr>
<tr>
<td></td>
<td>C++</td>
<td><a href="https://isocpp.org">https://isocpp.org</a></td>
</tr>
</tbody>
</table>
Just as with the simulation of the interaction of fundamental particles, the vast majority of these cutting-edge scientific projects (i.e., particle physics, accelerator physics, nuclear medicine, electromagnetism, optics, circuits' analysis, etc.) require very specific contexts and computer scenarios that are difficult to reproduce outside the academic/research field in which they were designed. This means that, when we must implement these tools in an external educational environment, technical difficulties often arise.

During the 2015–2016 academic year, the School of Engineering & Technology (ESIT, http://esit.unir.net) decided to move the application of some of these activities to virtual containers based on Docker. This led to a huge simplification of workflow and methodology for all students (twenty-one -21-in total), given that all they really needed to do was install the Docker chain of basic tools. Once installed, the students could download these resources from the Docker Hub or by means of a more modern tool called Kitematic (Figure 4), developed by the Docker team to handle virtual containers more comfortably.

These images were available in the Docker Hub and all students could download and use them, regardless of the institution to which the images belong. For this reason, these images can be considered OER and the Docker Hub behaved as an OERaaS platform.

The evolution in the number of students who presented did homework with the help of virtual containers was automatically measured thanks to assignments submission tool in our online campus. At the same time, the students’ perceptions towards virtual containers were studied through a forum thread within the campus forums tool.

**Results and discussion**

With regard to the results, we see that the students’ participation and satisfaction increased over time, in part due to the ease and speed of use of the containers technology and distribution of each of the physics classes mentioned above (Figure 5).
Over the course of a semester, these resources were openly available in the Docker Hub and the students could access, download and run them as if they were OER, with total independence and without worrying about the configuration of their personal host system. Our experience shows that the combination of educational resources in containers with free and open distribution channels can be one of the cornerstones of the OER approach in STEAM subjects.

Similarly, interest increased over this last year, with a particular increase in week six, where the fourth homework bulletin was presented. (Figure 5). During this week, students conducted the particle physics exercise described above. In this activity, the students simulated a particle beam and its possible interactions with matter and detectors. The ease of implementation of this task thanks to the use of a virtual container led a large number of students to continue solving the rest of the proposed activities, which were also distributed as independent light containers. Figure 6 represents the number of lab activities that were submitted across several semesters. In the first two, distant-learning students found awkward the submission of lab-related assignments, mostly due to the complex required computing setups. However, in the 2015–2016 edition of physics for computer engineers, lab-activities gained

Figure 5. Evolution of the percentage of students committed to the execution of alternative activities based on virtual containers.

Figure 6. The left side of the figure shows the percentage of participation and dedication to alternative activities based on the use of scientific software tools. The right side shows the degree of satisfaction of students with the virtual containers technology (used in the 2015–2016 academic year).
remarkable acceptance when compared against the previous two academic years. The pie chart on the right represents the students’ opinion regarding the use of virtual containers as a tool for their homework. Clearly, 80% of them claimed to be satisfied or very satisfied with the adoption of Docker, virtual containers and the way they are distributed.

**Conclusions**

The simplification of the distribution of computing scenarios in education is a key element in persuading students to use modern and highly complex STEAM learning tools. Transgenic learning is an approach to modify selected parts of the educational process and reinsert them towards an improved performance. Virtual containers mean a disruptive progress to learning and a powerful distribution tool for OER. In this article, we focused on the Docker project and its Hub platform for the easy and open distribution of virtual containers. We have shown, through a real case study, how this tool can work as an OERaaS platform. During the same case study, a progressive increase in the interest and participation of students in the use of the proposed educational tools was noted and measured, showing the progressive support from those students to the disruptive use of this tool. The use of a virtual container represents a disruptive approach for teaching and learning physics, and it follows the principle of transgenic learning: to select and extract a piece of the learning flow, to modify it and improve it, so that it is finally put it back to boost that very same process. Indeed, the students’ interest, interaction and performance were better with the virtual container that without it.

Finally, although the focus of this paper has been on the scientific/technical teaching side of the Docker and Docker Hub tandem, we would like to stress that there is plenty of room for its use in other activities, such as digital arts. There also exist hundreds of containers devoted to image manipulation software, photo-editing tools, etc.

**Disclosure statement**

No potential conflict of interest was reported by the authors.

**Funding**

This work was partially funded by Universidad Internacional de la Rioja (UNIR) (http://transfer.unir.net), through the Research Institute for Innovation, Technology and Education (UNIR iTED, http://ited.unir.net), the UNESCO-UNIR Chair on eLearning and the ICDE-UNIR Chair in Open Educational Resources (http://research.unir.net/unesco).

**Notes on contributors**

Daniel Burgos works as Vice-rector for Knowledge Transfer & Technology, Director of UNIR iTED (http://ited.unir.net), UNESCO Chair on eLearning, and ICDE Chair in OER, at Universidad Internacional de La Rioja (UNIR, http://www.unir.net). Formerly, he also was Vice-rector for Research and Dean of the School of Engineering at the same university, Director of Education Sector and Head of User Experience Lab in the Research & Innovation Department of Atos, and Assistant Professor at Open Universiteit Nederland. He has been involved in over 40 R&D&i, EU-funded projects, and have published over 120 scientific papers. Burgos holds degrees in Communication (PhD), Computer Science (Dr Ing), Education (PhD), Anthropology (PhD) and Business Administration (DBA).
Alberto Corbi works as a senior researcher at Research Institute for Innovation & Technology in Education (UNIR ITED, http://ited.unir.net) and as an assistant lecturer at the School of Engineering (http://esit.unir.net), both part of Universidad Internacional de La Rioja (UNIR). With a background in physics (MSc in ocean-atmosphere interaction), computing and education, he is currently involved in research fields about recommender systems, eLearning standards and systems interoperability. Simultaneously, he is working on his PhD thesis around medical imaging at the Spanish Council for Scientific Research (CSIC).

References

Albright, P. (2005). Open educational resources open content for higher education. In Internet discussions forums.


