ABSTRACT

Online learning and virtualization are two key and complementary technologies which are increasingly used for today’s education missions. Online learning offers great accessibility to a variety of educational materials and activities, whereas virtualization allows a variety of educational software to be easily distributed using virtual machines (VMs). However, in current practice, the use of VM-based education is largely disconnected from online learning. In fact, widely used online learning systems do not allow users to directly use VMs online, and the use of VMs often has to rely on physical laboratories which provide only limited resources to host the VMs and limited hours to access the VMs. This gap between online learning and VM-based education presents a serious hurdle to further adoption of these technologies. This paper presents a new approach to addressing this gap by integrating VM-based education with online learning. The result of this approach is CloudVLE, a new education system that seamlessly embeds the management and use of VMs in a widely used online learning system.

CloudVLE is among the first to enable instructors and students to flexibly and conveniently create and consume VM-based course materials in the same online environment that they are already familiar with and use for conventional course materials. Moreover, CloudVLE is also unique in its use of several emerging computing technologies for enhancing online VM-based education. It supports the use of cloud computing resources to provide scalable and cost-effective hosting of the educational VMs. It provides a mobile app to present an interface tailored for mobile devices, which further improves the accessibility and usability of the VMs. It integrates social networks with the traditional online forums used by online learning systems to enhance the interactions between instructors and students. Finally, this paper discusses the experiences from the use of CloudVLE for teaching an undergraduate operating systems course, and presents the highly positive feedback from the students taking this class.

1. INTRODUCTION

There are two key educational technologies that are increasingly used for teaching and fundamentally changing how training is delivered. On one hand, online learning systems (e.g., Moodle [1], Blackboard [2]) are widely used by teachers and students to conveniently create, distribute, and consume course materials such as slides, notes, homework, and quizzes. On the other hand, virtual machine technologies (e.g., VMware [3], VirtualBox [4]) are more and more used by educators from different disciplines to create self-contained and portable virtual educational appliances. Educational software can be conveniently packaged in their virtual machines (VMs) and transparently distributed and deployed to any computer. These two concurrent trends in educational technologies are also complementary to each other. An online learning environment can significantly enhance the accessibility of educational VMs. VMs enable fast distribution of educational software at scale, which is also great match to the model of online learning, e.g., for creating massive open online courses (MOOC).

However, in current practice, the use of VM-based education is disconnected from the use of online learning systems. Although instructors and students are already familiar with online learning systems and use them to post and access a variety of conventional course materials, they cannot distribute or use the educational VMs online. Consequently, the use of educational software still relies heavily on traditional physical computer laboratories which offers only limited computers and limited laboratory hours to both instructors and students. Hence, this gap between VM-based education and online learning prevents the users of educational VMs from benefiting from online learning, and in return seriously hinders the broader adoption of VMs for education.

This paper presents an approach to bridging the above mentioned gap by integrating VM-based education with typical online learning systems and providing users with versatile accessibility to both conventional course materials and new VM-based educational software in a coherent online environment. The result of this approach is CloudVLE, a new education system that seamlessly embeds the management and use of VMs in a widely used online learning system (Moodle [1]). Specifically, CloudVLE allows an instructor to create and customize VMs with a rich variety of educational software, e.g., applications, programming tools and libraries, and OSes, and make them available to students online; and it also allows students to access the instructor-prepared VMs, in the form of either shared VMs or independent private copies of the VMs, online for a rich variety of activities, e.g., watching a demo, conducting an exercise, and submitting an assignment. These activities happen entirely in the online learning system that the instructors and students are already familiar with.

CloudVLE significantly enhances learning experience by allowing instructors and students to enjoy the benefits of both VMs and online learning systems for education: VMs allow the educational software to be easily distributed and deployed, while the online learning system enables the VMs to be controlled and used at anytime from anywhere conveniently through the web. CloudVLE therefore completely removes the constraints of the traditional physical laboratories, allow the courses that still reply on them to be taken entirely online and free instructors and students from the limited computer lab resources such as space, equipment, open time, and support staff.

Moreover, CloudVLE is also unique in its use of several important emerging technologies, cloud computing, mobile devices, and social networks, for enhancing online, VM-based learning. First, CloudVLE supports the use of both private cloud (in-house virtualized computer clusters) and public cloud (Amazon EC2 [5]) resources for hosting the VMs employed for online education and benefiting from the scalability and economics of cloud computing.
Therefore, CloudVLE can deliver good performance to instructors and students for using the educational VMs at large scale and cost-effectively. Second, CloudVLE provides a mobile app to present an interface tailored for smart mobile devices such as smartphones and tablets which are becoming pervasively used and offer excellent mobility and performance. Therefore, CloudVLE’s integration with mobile devices further improves its accessibility, while providing great usability of the online educational VMs through its interface tailored for mobile devices. Third, CloudVLE integrates social networks and exploits their power of information dissemination to facilitate the communications between instructors and students. By linking the traditional online forums commonly used by online learning systems and the widely used social networks (Facebook), CloudVLE addresses the limited usability of online forums and further improves the learning experience that it offers to instructors and students.

Finally, this paper presents the experiences from using CloudVLE to teach a real undergraduate course on operating systems. In the past offerings of this course, the online learning system was used for only the conventional course materials, while VM-based kernel exercises were supported only through physical laboratories. With CloudVLE, everything required by the course is entirely supported online. The instructor can prepare VM templates for the exercises online, and students can make independent clones of the templates and use them for kernel development and debugging online. When the assignments are due, the students mark their VMs as submitted and the instructor can also easily grade the assignments using the VMs. An exit survey was conducted to collect feedback from the students, which shows highly positive responses regarding the use of CloudVLE. Most students responded that CloudVLE is better than the traditional physical laboratory model and they would like to use it for other courses as well.

The rest of this paper is organized as follows. Section 2 introduces the background and related work. Section 3 presents the design and implementation of CloudVLE. Section 4 discusses the experiences from using CloudVLE to teach a real course. Section 5 concludes the paper and outlines the future work.

2. BACKGROUND AND RELATED WORK

2.1. Integrating VM-based Education with Online Learning Systems

VMs (e.g., VMware [3], VirtualBox [4]) are powerful platforms for education. Educational software can be conveniently encapsulated in VMs and transparently distributed and deployed to any computer. For examples, to teach web development, VMs allow the students to set up their own web software stack and get hands-on experience with multi-tiered web architecture; and the availability of isolated VMs also avoids any potential conflict (e.g., port numbers) among the students. To teach operating systems, VMs allow students to conveniently and safely work at kernel space, hacking and restoring kernels at ease and with no danger of comprising the underlying system nor affecting other students. To teach big data, VMs allow an advanced distributed system such as Hadoop to be easily set up on a set of VMs on a single computer, without the need of multiple physical computers and the overhead of managing them. To teach a scientific application (e.g., weather forecasting modeling, wind tunneling simulator), VMs allow the application-required, complex execution environment (OSes, libraries, and tool chains) to be set up only once by the instructor and distributed to many students by simply copying the VMs. Many of such success examples of VM-based education can be found in the literature and practice [6][7].

Online learning software (e.g., Moodle [1], Blackboard [2]) is widely used by instructors and students to create, distribute, and consume course materials such as slides, homework, and quizzes through the web. However, in current practice, the use of VM-based education is disconnected from the use of online learning systems. Instructors have to come up with ad hoc ways to provide VMs to students, particularly by putting them on a website to allow students download or by hosting them in the workstations of a physical laboratory to allow students use. The former approach is always cumbersome due to the large size, typically at least a few gigabytes of VMs. If the VMs are used for assignments, the students also have to upload the VMs in order to submit the assignments, which further aggravates the problem. In the latter approach, the VMs can be only installed on a limited set of laboratory computers, while the students access them only during limited laboratory hours.

This disconnection prevents the users of educational VMs from benefiting from online learning, and in return seriously hinders the broader adoption of VMs for education. For example, an operating systems course that the author teaches makes intensive use of VMs to support Linux kernel based exercises. In the past, Moodle was used only for distributing conventional materials such as slides and notes, while the use of VMs still relied on the physical laboratory. Despite of the instructor effort of arranging additional laboratory hours, many students were still frustrated by accessibility and resorted to working on their own computers. Many also gave up on the idea of using VMs, because it was cumbersome to copy VMs back and forth between the laboratory computers and their own.

CloudVLE addresses the gap between VM-based education and online learning in current instructional technologies. It provides a new software system that seamless integrates VMs in widely used online learning systems and supports teachers and students to use them in a coherent online environment. There is related work that integrates educational VMs in an online portal [10][11], but to the best of our knowledge, CloudVLE is the first to support VM-based education in a widely used online learning system that users are already familiar with and provide a coherent, effortless user experience. CloudVLE is also among the first to exploit the power of cloud and mobile computing technologies for online learning, which will be discussed next.

2.2. Exploiting Cloud and Mobile Computing for Online Learning

Cloud computing systems emerge as important computing platforms for many disciplines because of their elasticity—the ability to dynamically grow and shrink the resources provisioned to an application on demand, and economics—the ability for users to run their applications at scale without up-front or long-term commitment to the resources. Public clouds (e.g., Amazon EC2 [5]) allow public users to rent resources and run a wide variety of applications; private clouds allow users from the same organization to run their internal applications on shared resources (e.g., a computer cluster virtualized using VirtualBox [4]). Education can also benefit from the elasticity and economics of cloud systems, as modern educational software become increasingly complex and demanding. For examples, to teach operating systems using real-world OSES, each group of students needs a physical or virtual machine with sufficiently provisioned CPUs and storage resources because kernel compilation is a CPU and I/O-intensive process. To teach big-data or high-performance computing, clustered physical
or virtual machines are required in order to allow the students appreciate the power of parallel programming.

Another important technology trend is that smart mobile devices such as smartphones and tablets become pervasively used and increasingly powerful. On one hand, today there are more smart mobile devices shipped than PCs and more users accessing Internet from their smart devices than from desktops. On the other hand, the performance gap between smart mobile devices and PCs is getting smaller, and smartphones and tablets are able to run a wide variety of tasks that any general-purpose computer can do while they also possess unique capabilities such as mobility, location awareness and new interfaces based on touch screens, gyroscope, and accelerometer. Therefore, an online learning system must be able to exploit the pervasiveness and power of mobile devices in order to adapt to the changing personal computing landscape and train our students more effectively.

There is related work on exploiting private cloud resources to support VM-based education [10][11]. In comparison, CloudVLE supports the use of both public and cloud resources for VM-based education in a widely-used online learning system. There are also related mobile apps that provide interfaces to widely used online learning systems [1][2], but none of them supports the use of VMs for education. In contrast, CloudVLE enables seamless use of VMs in familiar online learning environment and offers convenient interfaces to the VMs through both web and mobile app.

3. DESIGN AND IMPLEMENTATION

Figure 1 illustrates the architecture of CloudVLE which consists of four key components. 1) It integrates VM access directly in an existing, widely used online learning system. The current prototype implementation is based on Moodle, which is open-source and one of the most widely used online learning system. 2) It makes use of cloud computing resources to host the education VMs. 3) It provide online access to the VMs via a coherent web interface that users are already familiar with. 4) It also provides a mobile app interface that is tailored for smart mobile devices. The rest of this section will discuss these components in detail.

3.1. Integration with Virtual Machines

CloudVLE seamlessly integrates the use of VMs as an integral part of its online learning environment. Teachers and students are able to conveniently use VMs to distribute and consume course-required educational software in the online environment that they are already familiar with. Specifically, CloudVLE supports the following VM-based activities. First, it allows an instructor to create and customize VMs with the desired educational software, including OSes, libraries, and applications. These VMs can be easily created by cloning existing basic images preinstalled with commonly used OSes and libraries or from a VM that the instructor created offline and uploaded to CloudVLE. The instructor can then deploy these VMs for various purposes such as demos and assignments. Second, it allows students to access the VMs prepared by the instructor, e.g., to learn the software installed in the VM, or create their own copies of the VMs, e.g., to do exercises independently in isolated VMs. When the students are done with their exercises, they can submit their results in the form of their VMs for the instructor’s evaluation and feedback.

All of the above mentioned user activities are supported completely online through the web interface provided by the online learning system. The look and feel of the VM-specific features are consistent with the rest of the system. Here are some examples of what this interface provides. 1) A student can control only the VMs created by him/herself or his/her groups. An instructor can control all the VMs created by his/her students. 2) When the instructor needs to use VMs for a new assignment, he/she creates the assignment of the VM type, which brings thee VM configuration interface. 3) Once a VM is up and running, the user of the VM can access the VM via a remote desktop provided as part of the web interface of CloudVLE. 4) When a group of students works on a shared VM, they can choose to share the same desktop (e.g., for viewing the materials together) or have individual desktops (e.g., for independent code desktop). 5) The user of the VM can conveniently take snapshots from time to time to save the current state of the assignment and can roll back to a previous snapshot at any time to restore the earlier work. 6) The VM desktop displays a countdown till the assignment, and when the time is up, the VM can be automatically locked, if the instructor chooses to prevent late submission, till the instructor is ready to grade it.

Figure 2 presents a snapshot of the VM desktop provided as part of CloudVLE’s web interface. Overall, this project will substantially benefit faculty to allow them to conveniently distribute and deploy education software and benefit students to allow them to conveniently access the software anytime anywhere.

3.1. Integration with Cloud Systems

CloudVLE can employ a variety of private and public cloud computing systems to host the VMs for its users and benefit from the scalability and economics provided by cloud computing. The CloudVLE allows plugins developed for different VM software and different cloud systems. Currently, it supports both VirtualBox-based VMs hosted on a private cloud and the VMs hosted on

Figure 1. Architecture of CloudVLE

Figure 2. A snapshot of VM interface on CloudVLE
Amazon EC2, VirtualBox is an open-source, free VM product that can be installed on a variety of OSes including Windows and Linux. CloudVLE interfaces with VirtualBox through libvirt, an open-source library for VM management. Amazon EC2 is one of the most widely used public cloud providers which offer VM services based on high-performance VM platforms (e.g., Xen). It leases VMs with different capabilities and different price (including a free usage tier). CloudVLE interfaces with EC2 through its well-defined web-services interface, Amazon Web Services.

Virtual machines allow the educational software to be conveniently distributed and deployed, while the enhanced online learning system enables the virtual machines to be controlled and used anytime anywhere completely through the web. Therefore, these courses can be taken entirely online and do not have to be constrained by limited physical lab resources anymore.

3.2. Integration with Mobile Devices

CloudVLE Mobile is a mobile app that further allows instructors and students to use CloudVLE through mobile devices such as smartphones and tablets. CloudVLE Mobile was implemented as a native mobile app and it possesses a device friendly interface so the device screen, storage, and other features can be used efficiently. The prototype was developed in Android since it is the most widely used mobile OS.

CloudVLE Mobile provides a subset of the features available on the CloudVLE web interface, which are also tailored for the screen sizes of mobile devices in order to deliver the best user experience on the devices. Users can still conveniently manage their VMs on their mobile devices, and access their VMs via remote desktops with resolutions matching the screen sizes. But on smart phones with small screens, the remote desktops are often cumbersome to use. To address this issue, a command-line interface based on Secure Shell (SSH) is provided as an alternative way to access the VMs. This command-line interface is also useful when the mobile network connection is not fast enough to support the remote desktops on the devices. Figure 3 presents the snapshots of the VM’s control interface and command-line interface.

3.3. Integration with Social Networks

Traditional online learning systems still rely on online forums, chats, and messaging systems to support the interactions between instructors and students. Although these tools are straightforward to use, they are quite cumbersome in practice. When a user, either an instructor or a student, wants to post a question on the forum, the user has to go to the computer, log into the website, and open the forum, which often takes minutes even before the user is able to draft the message. When someone posts a response to the user’s earlier question, although most systems allow the user to receive the response via an email, the user still has to go through all the aforementioned steps in order to follow up on the response. Moreover, the other online user interaction tools provided by a traditional online learning system, such as chats and messages, are rarely used by users because the users do not stay in the system most of the time nor at the same time.

CloudVLE addresses the above mentioned limitations of traditional online learning systems by integrating social networks to support the interactions between instructors and students. Social networks are highly popular and widely used by students to interact with their friends in a variety of ways. The pervasive use of smart mobile devices and mobile apps tailored for such devices make social networks even more convenient to use. Hence, social networks can be a powerful platform for education by supporting effortless interactions between instructors and students. CloudVLE takes advantage of this emerging technology by integrating it to support the interactions between instructors and students.

The integration with Facebook is done in two folds. First, an app is created on Facebook to present the interface to CloudVLE. This app runs on both standard computers and mobile devices, and its look and feel are similar to Facebook and its other well-known apps (e.g., Quora, Glassdoor), so there is no learning curve for users who are already familiar with the social networks. The homepage of this app includes the news feed for recent events, a list of courses that the user is currently enrolled in, a list of trending tags, and a list of links to Facebook accounts of classmates. Users can select a forum from a course and navigate a list of threads within that forum. Users can view comments within those threads, post responses to the comments within a thread, and start new threads of discussions. Users can “like” a forum comment to signal satisfaction with the information provided, and tag a comment to associate it with the relevant categories. Users can search the forums for information using keywords and tags through a search bar. When there are new messages posted on a user’s forums, the user receives personalized notifications in the form of messages, icon change, sound, and vibration, in the same way as a typical app.

Second, the information presented on the CloudVLE’s Facebook interface is automatically synchronized with the information on the traditional Moodle forums so that users who access CloudVLE through these two different interfaces can see the same discussions and interact with users on Facebook. The Moodle forums do not support social-networks-specific information such as “likes” and tags, but the rest of the information on the forums is the same as on the Facebook app. The user accounts on Facebook are also linked with their accounts on Moodle so that only registered Moodle users can use the Facebook app and the comments posted by users through the Facebook app can be attributed to the corresponding Moodle users.

![Figure 3. Snapshots of CloudVLE mobile app: (a) VM control interface; (b) VM command-line interface.](Image)
4. EXPERIENCES

CloudVLE has been deployed for production use by several real classes at the author’s institution. This section summarizes the experiences from using CloudVLE to teach a core undergraduate course and the feedback from the students in this class.

4.1. Teaching Operating Systems on CloudVLE

This operating systems course is a required course for computer science undergraduates, taken by mostly senior students. It includes both a lecture component and a lab component. For the lecture component, in addition to allowing the lecture materials such as slides and notes to be distributed online like in a typical online learning system does, CloudVLE also enables the instructor, teaching assistant, and students to engage with one another through its unique social networks integration. For the lab component, CloudVLE’s support of VMs and cloud resources for teaching is a key differentiator from traditional online learning systems, which is discussed in detail in the rest of this subsection.

The lab exercises are a significant component of this operating systems course, in which students are required to apply the fundamental concepts and principles to the practical operating systems design and development. These exercises are centered on the main themes of the lectures, including OS structures, process management, memory management, and storage management, and they are assigned when the corresponding topics are taught in class. Each lab exercise typically requires a group of students (typically two to three) to complete in about a month.

The author was the first to introduce Linux kernel based lab exercises to this operating systems course at the author’s institution, and has been offering it for a few years. Traditionally, operating systems have often been taught using simulator-based educational OSes. However, as argued in the related work, using an advanced real-world OS to teach this course can help students appreciate how the fundamentals are applied in realistic settings and gain real systems development experience valuable to their future careers. Linux is chosen not only because it is a modern open-source OS, but also because of its importance and pervasive use in real-world systems, covering the entire spectrum of computing systems from mobile devices (e.g., Android, Ubuntu One) to supercomputers (e.g., Computer Node Linux, Kitten).

Taking one of these operating systems course offerings that the author made as an example, it included four lab exercises: 1) Linux and VM basics: the students are required to create a VM, install a Linux-based OS on the VM, install the most recent Linux kernel from source code, implement a new system call to the new kernel, and implement a new kernel module; 2) Process Management: the students are required to perform certain process management tasks using kernel threads, and use the kernel-space synchronization primitives to implement the synchronization among these kernel threads; 3) Memory Management: the students are required to walk page tables and use the collected information to implement a new page frame reclamation algorithm; 4) Storage Management: the students are required to implement a new file system tailored for flash storage under Linux virtual file system.

The traditional model for conducting these operating systems lab exercises is that the instructor arranges a physical computer laboratory where there are enough workstations for the entire class and a period of time every week when the students as well as the instructor and teaching assistant can attend. VMs are also used in this traditional model where they are stored in a centralized server and each student can access it from any workstation in the laboratory. However, there are several serious limitations of this traditional physical laboratory model to support the use of VMs.

First, the availability of the physical laboratory seriously limits students’ access to their VMs. The instructor has to request additional laboratory hours to provide students more time to work on their VMs, which is still insufficient and inflexible to the students. Second, the performance of the VMs is often bottlenecked by the networked storage commonly employed in the physical laboratories. To allow students to freely use any workstation in the laboratory, their VMs must be stored in a networked storage system such as the Network File System. However, in a physical laboratory, it is much more difficult and expensive to build a high-performance networked storage system compared to a consolidated environment. Storage performance is critical operating systems exercises which require frequent kernel compilations and has indeed been complained a lot by the students. Third, it is difficult for students to exploit both the physical laboratory resources and their own computers for the exercises. The students all want to work on their own computers due to the frustration with the physical laboratory availability. But the large size of VMs (typically at least a few gigabytes) makes it which unfortunately creates problems to VM reliability—students often lose their work due to problems with their computers, grading—students have to turn in their VMs, and many others.

The use of CloudVLE effectively addresses the above limitations of the traditional physical laboratory model. First, by exposing the VMs through an online learning system, the students can access their VMs 24/7 from anywhere there is Internet connectivity. Second, by hosting all the VMs on consolidated high-performance servers, CloudVLE provides excellent performance to the I/O-intensive operating systems exercises. The current CloudVLE prototype is set up on several rack-mounted commercial off-the-shelf servers where one of them serves as the storage server, storing the VMs accessed by the others via a commodity Gbps Ethernet. Third, based on the typical Internet connectivity to the students on campus and at their homes, the students generally do not need to do substantial work on their own computers for the operating systems exercises. Finally, by freeing the instructor and teaching assistant from the physical laboratories, they have much more flexibility to meet with the students face-to-face and discuss the exercises by
accessing the VMs through CloudVLE. Such physical interactions are in addition to the unique virtual interactions enabled by CloudVLE’s integration with social networks.

4.2. Student Feedback

We collected feedback from a class of undergraduate students who took the operating systems course and completed all of the four lab exercises using CloudVLE in spring 2013. A total of 28 students were in this class and 22 of them completed the survey. Figure 2 summarizes the results of several main questions asked in the survey. Based on the responses to the first four questions, we can see highly positive feedback to the benefits of CloudVLE. The majority of the students agree that CloudVLE is better than the traditional physical laboratory model, provides intuitive access to VMs, and helps them complete their assignments faster and more comfortably. The majority of them also would like to use CloudVLE for their other courses.

The survey also asked the students to describe the most favorite and least favorite features of CloudVLE. Most of the students expressed appreciation to the following features: 1) Working on their VM assignment directly in Moodle; 2) Accessing the VM at anytime from anywhere; 3) Sharing the same VM with their teammates; 4) Fast kernel compilation; 5) Working in the proper environment provided by the instructor (in the VM); 6) Easy and fast submission of assignment. In terms of the least favorite feature of CloudVLE, most of the concerns expressed by the students are about the VM’s remote desktop GUI interface is sluggish when their network speed is limited. We quickly realized this issue while offering this course using CloudVLE, and we provided an SSH-based command-line interface to access the VMs in CloudVLE. In the survey, when asked about this command-line interface, all of the students replied that they used it, in addition to the remote desktop GUI interface, and they were happy to have this alternative.

The survey also asked for feedback on the integration of CloudVLE with social networks and mobile devices, which is also summarized in Figure 2. By the time when this operating systems course was offered, the social networks component and the mobile app of CloudVLE were not completed yet, so the feedback to these two questions were based on the students’ experiences outside of CloudVLE. The responses are positive although there are still many reservations. However, the landscape is rapidly changing: social networks are gaining wider adoption and deeper penetration; and mobile devices are dominating the personal computing arena while getting more powerful with larger and better screens. Given that this survey was done back in early 2013, we believe that today’s students will be much more positive to these ideas. However, there is obviously still work to be done in order to make effective use of these technologies. We plan to offer this course again using the current version of CloudVLE, gather the feedback on our Facebook integration and Android mobile app, and use it to guide our future enhancements to CloudVLE.

5. CONCLUSIONS AND FUTURE WORK

Emerging computing technologies are indispensable to continue improving education, which in return helps train our workforce more effectively who will continue innovating in the computing field. This paper presents a new online learning system, CloudVLE, which seamlessly integrates several important technologies for enhancing education. First and foremost, CloudVLE embeds the use of VM-based education directly in a widely used online learning system, thereby allowing instructors and students to conveniently use VMs for a variety of course activities in a familiar and coherent online environment. Based on this framework, CloudVLE further enhances the use of online educational VMs by hosting them on scalable and cost-effective cloud resources, providing an app interface tailored for mobile devices to further improve the accessibility, and integrating social networks with online forums to improve the communications between instructors and students.

The prototype of CloudVLE was used in production for an undergraduate operating systems course, which was traditionally offered using a combination of online learning system and physical laboratories. CloudVLE has made it possible to take this course completely online, providing access to both conventional course materials and new VM-based materials in the same online environment. The exit survey conducted at the end of this course shows highly encouraging feedback from the students, most of whom responded that CloudVLE helped them complete their assignments faster and more comfortably and it was much better than accessing the VMs in physical laboratories. The students also wanted to have CloudVLE available for their other courses.

CloudVLE is only a first step towards making effective use of the emerging computing technologies to enhance education. Based on our current prototype and initial experiences, there are several aspects of CloudVLE that we intend to improve and new research questions that we plan to pursue answers to. For example, how to automatically optimize the remote desktop interface to the VMs based on the available network connectivity? How to use social networks more effectively to encourage students to communicate with instructors and help instructors understand the problems that the students have? We also expect to use CloudVLE to teach more diverse courses, e.g., courses that teaches non-system-level programming and courses from non-computer-science domain, and gather more feedback for improving the system. We will make the code of CloudVLE publicly available and elicit help from the community to improve it collaboratively. Indeed, there are related efforts from community that we can already leverage to enhance CloudVLE, e.g., the Apache VCL software [11] can be employed to improve the VM management aspect of CloudVLE.

6. REFERENCES


