Growing the Use of Virtual Worlds in Education: an OpenSim Perspective

Colin Allison, Anne Campbell, C.J. Davies, Lisa Dow, Sarah Kennedy, John McCaffery, Alan Miller, Iain Oliver, Indika Perera

School of Computer Science, University of St Andrews

Abstract. The growth in the range of disciplines that Virtual Worlds support for educational purposes is evidenced by recent applications in the fields of cultural heritage, humanitarian aid, space exploration, virtual laboratories in the physical sciences, archaeology, computer science and coastal geography. This growth is due in part to the flexibility of OpenSim, the open source virtual world platform which by adopting Second Life protocols and norms has created a de facto standard for open virtual worlds that is supported by a growing number of third party open source viewers. Yet while this diversity of use-cases is impressive and Virtual Worlds for open learning are highly popular with lecturers and learners alike immersive education remains an essentially niche activity. This paper identifies functional challenges in terms of Management, Network Infrastructure, the Immersive 3D Web and Programmability that must be addressed to enable the wider adoption of Open Virtual Worlds as a routine learning technology platform. We refer to specific use-cases based on OpenSim and abstract generic requirements which should be met to enable the growth in use of Open Virtual Worlds as a mainstream educational facility. A case study of a deployment to support a formal education curriculum and associated informal learning is used to illustrate key points.

Keywords. Virtual Worlds, Education, Open Simulator, XMRM, MUVE

Introduction

Immersive 3D Multi User Virtual Environments, hereafter referred to simply as Virtual Worlds, are currently used in a number of different sectors including training, games, entertainment, social networking and education. They have been of research interest to educationalists for over ten years since the availability of Second Life as a service on the global Internet. Virtual Worlds for learning have been created to support topics including Internet routing [1], WiFi experimentation [2], electromagnetic theory [3], programming algorithms [4, 5], HCI [6], archaeology [7], space science [8], cultural heritage [9], humanitarian aid [10] and linguistic and cultural competence for American forces [11]. Part of the motivation for using Virtual Worlds is that students and pupils readily engage with them, often more so than with conventional learning materials and contexts [5, 12]. Second Life was pioneering in its global reach but it was not designed for education and several commentators have highlighted problem areas that arise when using it for that purpose [13, 14]. These include: commercial cost, code size restrictions, age restrictions, unwanted adult

1 Corresponding Author.
content and other distractions, difficulty of coursework marking due to the permissions system, quality of experience due to remote servers, firewall blocking by campus computing services, a lack of facilities for copying and sharing content and backing up work outside of the virtual world. Other issues arise – those of ethics, trust, privacy – and the technological barriers to running a class with software which does not scale. In recent years OpenSim [15] has increasingly displaced Second Life (SL) as the platform of choice for developing immersive learning environments. OpenSim is an open source project which uses the same protocol as Second Life so is compatible with any SL compatible viewer/client including the SL viewer itself, Hippo [16], Meerkat [17] and Phoenix [18]. This software compatibility has resulted in a de facto standard for Virtual Worlds and has meant that OpenSim offers a natural progression from Second Life for educationalists. While OpenSim offers solutions to many of the significant drawbacks encountered with Second Life - commercial cost, age restrictions, land constraints, content sharing and backup - there are features (or the lack thereof) inherited from Second Life which act as barriers to a wider adoption of Virtual Worlds in education. This non-exhaustive set of challenges includes Management Guidance, Network Infrastructure, the 3D Web and Programmability. These are followed by a case study which involved deploying an immersive historical reconstruction as part of a national curriculum for high schools.

1. Management Guidance for Educators

Virtual Worlds bring a novel set of management challenges to the stakeholders at every level of MUVE use: educators creating a new learning environment, instructors using a ready-made learning environment, and learners. Due to the intrinsic characteristic of 3D simulations that Virtual Worlds are capable of defining unique use cases for different application domains based on their core set of system and environmental functions. As a result, managing Virtual Worlds can become a significant challenge, not only because they are new when compared to Web-based platforms, but also because they have a radically different nature which varies across application domains and underlying goals of the use. To support users for their MUVE management needs, we are researching suitable mechanisms with a particular focus on educational applications.

Virtual Worlds follow system and environment controls through Role Based Access Control (RBAC) [19]. OpenSim has three different categories and implementations of RBAC to cover the Virtual World functions, i.e., Land related RBAC, Content related RBAC, and Group related RBAC. These three different RBAC implementations are not mutually exclusive from one another; in fact, varying levels of interdependencies can be seen with prominence often given to the Land related RBAC. Further, when combined with the possible avatar activities and user management functions, all these create a complex and challenging scenario to manage Virtual World at the system level, let alone the complexity added by different Virtual World application domain requirements. However, interestingly, OpenSim does not enforce a specialised RBAC at server administration, but relies on the hosting server’s operating system user access management. Therefore, any user with access to the server (directly or remotely) can perform administrator functions without any restriction; this can be a challenge for mainstream uses that require higher trust and security enforcement.
Briefly, Land or the spatial management of Virtual Worlds are based on three different levels with related roles: Estate (Estate Owner), Region (Region Owner), and Parcel (Parcel Owner). An estate can have one or more regions, and a region can have one or more parcels; similarly, an Estate Owner has the highest user privilege for that estate, and then the region and parcel owners have decreasing levels of control corresponding to their land ownership. The land privileges can affect the group and content management since the Virtual Worlds are designed with a focus on the 3D environment. Content management involves 6 different roles: Creator, Owner, Next Owner, Group, Everyone and Administrator (God), with functions, Move, Copy, Modify, Transfer and Delete. Combinations between these roles and functions can make Virtual World content management a difficult task. Moreover, complex scenarios such as composite permission, cyclic permission loss and fair ownership model were identified as essential factors for effective Virtual World content management [20]. Group management is relatively straightforward at the basic level with two roles, Group Owner and Member. However, the flexibility to define new roles for a given group and delegate owner privileges can create unexpected group management problems if not properly handled. Avatar activities are the main form of interaction with the 3D environment that users can practice. Indeed, these functions such as fly, teleport, gesture, terrain editing, content creation, etc. are the reasons why Virtual Worlds become so engaging and intuitive when compared to the 2D Web; therefore, restrictions on these functions for managing the Virtual World can result in a negative impact on their usability and attractiveness.

Considering the generic nature of these Virtual World functions and their core role in determining the success of the management of Virtual World usage in different educational domains, we have proposed a mechanism based on policy considerations. The advantage of this approach is it takes into the account the need for flexibility at the use case level to customise policy considerations. Therefore, it allows the users to practice unique policy considerations based on a selected Virtual World function, depending on their requirements at the application level.

We have defined a process for policy based management practice relating with the Kolb’s experiential learning model [21], which has been widely considered to be successful with Virtual World supported teaching and learner support [22, 23]. As part of this management process development, an extensive user study with two student cohorts from undergraduate and postgraduate (taught) programs was performed to identify key policy aspects of Virtual World management. The study revealed that the policies should be based on promoting student self-regulatory behaviour and Virtual World system function management [24]. In view of this outcome, a policy taxonomy was introduced to avoid ambiguities at the policy making stage since there can be multiple ways of using Virtual Worlds for different educational needs. A further study with a sample of PhD students from Scottish universities stressed the fact that there should be user guidance and training for managing Virtual Worlds, especially on the functions and their complex interrelationships. This initiated our work on exploring different methods to train academics and students for their Virtual World interaction and management activities; an overview on the training environment which was developed follows.
1.1. User Training for Virtual World Management

To facilitate the need of user training for managing OpenSim we developed Training Island, with training centres to cover major policy areas related to Virtual World functions. A campus like architecture was used to make the academics feel at home. This region was accompanied by Introduction Island, for generic student training for rapid familiarisation with the Virtual World, implementing the SL concept of Help Island for new users, but in a more comprehensive manner focusing on educational uses. Training content and support materials were expressed in natural language with suitable figures to depict the Virtual World management task scenarios and related User Interface functions. Since the management and interactive tasks are done through user actions at the application level, it was decided to express the content in natural language; moreover, it adds flexibility and ease of training. Fig. 1 shows an aerial view of Training Island.

In addition to the policy specific training centres, four generic virtual spaces and constructs were provided to increase user engagement and facilitate different types of training activities and media formats. These places are: the Discussion Rooms, the Cinema, the Open Forum and the Sandbox. The Cinema provides the virtual space to train users through a set of video content on managing learning activities in virtual worlds, the Open Forum is a dedicated place to summarise most of the Virtual World interacting functions with possible hotkey combinations available in the viewer programs; this enables a rapid familiarity of, and efficient interactivities with the environment. The Discussion Rooms are places where group discussions can be hosted for training sessions on using Virtual Worlds. Finally, the dedicated Sandbox lets the users practice their learnt environment engaging and management functions without restriction; it helps to make the main training environment to be kept tidy and enforced with some restrictions on terrain alteration and object creation.
2. Networking Infrastructure Considerations

OpenSim’s networking protocols and their parameter constraints are largely inherited from Second Life. These were originally developed over ten years ago, and although well studied [25, 26], have not been seriously revised despite advances in network infrastructure and computer capabilities over the past decade [27]. There are two general problem areas: security-motivated port blocking and use of bandwidth on the Internet.

In the first case a recurring problem encountered in the common educational settings of the School or the Campus is that the authorities who control institutional firewalls are reluctant to open the large range of ports (53, 3478, 3479, 5060, 5062, and 12000-29999) required for client-server communication, due to security concerns. This can be overcome by placing a server inside the institutional firewall but this is a poor solution in two respects: i) it removes the efficiency of running a particular learning environment as a single service from one or more servers across the Internet; and ii) there is no opportunity to mix and meet with users outside of normal institutional cyberspace boundaries – friends, fellow travellers and domain experts for example. Another solution is to lease an ISP connection, as ISPs deliver unfiltered IP, but in addition to the extra expense involved this approach may contravene institutional rules.

In the second case it is necessary to understand that the Internet’s good health is due in large part to TCP’s self-regulatory congestion control algorithms [27], which also provide a type of fairness [28] in bandwidth usage amongst competing streams of traffic. So, for non-TCP based traffic, OpenSim for example, it is important that these applications dynamically manage their traffic transmission rates to be TCP Fair. In a nutshell, TCP takes loss as an indication of congestion and reduces its transmission rate. In the absence of loss it increases its transmission rate to take advantage of available bandwidth.

OpenSim implements its own adaptive traffic management mechanism, which involves a “throttle” mechanism to limit the maximum rate. This has three shortcomings: i) it fails to reduce its transmission rate in line with the TCP Fair algorithm in the presence of loss, so can waste bandwidth and contribute towards congestion; ii) it fails to take advantage of available bandwidth to improve the quality of experience for the user/avatar; iii) it has a default absolute rate limit (throttle max) of 1.5Mb/s with a user settable maximum limit of 3Mb/s, regardless of available bandwidth, such as the 100Mb/s plus found on local area networks – a typical campus scenario. The Mongoose client [29] addresses problems (i) and (ii).

With respect to (iii), a particular problem caused by the conservatism of the inbuilt possible maximum throttle values is the significant delay encountered when initially downloading a virtual world from a server to the client. Figure 4 shows a range of initial download times for Cathedral Island, a 3 * 2 OpenSim mega-region for a range of throttle settings.
When set to the lowest likely to be encountered using an ISP (500Kb/s), it takes over 4 minutes; at the best available throttle value of 3Mb/s it takes 1.5 minutes. It would clearly be beneficial to develop OpenSim clients and servers to allow for significantly higher bandwidth to be used where it is available, as is usually the case on local area networks and wide area academic networks.

3. Towards the 3D Immersive Web

Unlike Second Life, OpenSim is very much focused on the idea of the 3D Immersive Web: “OpenSimulator lacks support for many of the game-specific features of Second Life (on purpose), while pursuing innovative directions towards becoming the bare bones, but extensible, server of the 3D Web” [15]. Emerging standard efforts to support interoperability of virtual worlds include MPEG-V [28] and the IEEE Virtual Worlds working group P1828 [29]. Over the next few years as the computing stock is upgraded the infrastructure to support immersive 3D applications will improve. Virtual Worlds have the potential to play an important role in facilitating the creation of 3D applications and in making them accessible as an extension to the existing World Wide Web. For this to happen they need to be seen not as a 3D alternative to the 2D Web but rather as an extension of the existing web. If Virtual Worlds are seen as a component that contribute to the extension of the web into three dimensions, then the walled garden approach adopted by Second Life appears less appropriate than the OpenSim approach of developing a toolkit for serving 3D applications. It should be possible to move seamlessly from the Web to a virtual world environment and back again. Getting from a Virtual World to the web is one of the strengths of OpenSim. Links to web pages can be embedded on the surface of objects and launched from scripts. They can be viewed in an in-world browser or an external browser. Thus the 3D world can be an interface to 2D Web resources. Moving from the 2D Web to the 3D world is more problematic. It can work in two ways. Either the browser opens a separate 3D client and communicates details from the web page to the client to allow it to connect to the appropriate virtual world or the browser has a plug-in which handles virtual world content. There have been projects for both approaches however to date they have either been incomplete or proprietary, so there is no readily available way to move from the
web to a virtual world, and achieving this must be a research priority for wider adoption of virtual worlds in education.

4. Programmability

As part of its compatibility goals OpenSim has carried over support for Linden Scripting Language (LSL [30]) from Second Life. LSL is very good at the job it was designed for: adding basic interactivity to individual objects within a scene. If more complex behaviour is required LSL can be used but the process is not simple. For example, LSL scripts can only communicate with each other via chat requests. As such all processing for one script must be done in one single text file. This makes writing and understanding complex LSL code hard to do. There is also no support for referencing third party utility libraries. When creating more substantial software for interactive learning environments such as virtual laboratories LSL is far from adequate. In addition to lacking the expressive power of object-oriented or declarative languages there is no support for development environments, debugging, libraries, code packaging, and IDE facilities such as dynamically loading new code into a live environment for testing.

External Mini Region Modules (XMRMs) are a scripting extension for OpenSim. Their purpose is to give developers a powerful way of programming content that exists within the virtual world. Scripts are written as external libraries in C# and loaded via short in-world scripts pointing to their location. The external libraries can be re-compiled and re-loaded in-world without having to shut down the server. Scripts can control the appearance, characteristics and position of objects as well as interacting with avatars and the terrain. One of the limitations of LSL is that it is designed so that scripts only affect the object they have been placed in. Manipulating several objects at once is difficult using traditional LSL mechanisms. XMRMs do not suffer from this limitation; the entire scene graph is available to them. They have all the powers of more traditional, in-world, scripting mechanisms but the fact they can be written externally using an IDE opens their scope.

Routing Island illustrates what can be achieved by programming with XMRMs [1, 31]. It is a configurable, interactive simulation of either Link State or Distance Vector routing. Once in-world users can create network topologies, run their chosen algorithm across the topology and alter their topology and watch the results in real time. Figure
5a shows an Internet core conceived of as a hypercube; Figure 5b shows a situation where students have modelled part of their local metropolitan area network in routing island.

These simulations also allow students to view the forwarding table for nodes and save and load topologies. Different configurations allow the simulation to run as a personal sandbox, a shared, collaborative sandbox, a static demonstration or as an interactive, pre-recorded demonstration. It was created entirely using XMRMs. It uses external libraries to read configuration files and do dependency injection late coupling for different configurations. It takes XML configuration files which allow dozens of separate control variables to be set to control behaviour. The simulation itself is split into multiple separate libraries. It is extensible and features its own plug-in mechanism allowing new algorithms to be written and tested on the system without having to alter any other part. All this development is significantly outside the scope of standard LSL. Even if using an external HTTP server to do most of the processing the complexity of the system would be prohibitive. XMRMs made it possible to develop Routing Island using standard software engineering methodologies to manage the complexity.

5. Case Study: Linlithgow Palace

Considerable effort is needed to deploy an immersive learning environment compared with an educational web site, even within the confines of an educationalist’s own institution. Deploying a substantially crafted virtual world into a different institutional setting, where there are other educational contexts in terms of curriculum, classes and age groups raises further challenges. We give an account of an example - Linlithgow Palace - that involved multiple inputs from various contributors, aimed at the primary and secondary education sectors in Scotland.

An OpenSim-based 3D reconstruction of Linlithgow Palace was installed in a computer lab at the local school, Linlithgow Academy, effectively combining cultural heritage, educational activities and virtual worlds. Linlithgow Palace was the birthplace of Mary Queen of Scots and was once lived in by successive Stewart kings. The preserved monument of the Royal Palace can be visited in a picturesque park setting overlooking Linlithgow Loch. The development of the reconstruction (Figure 6) was a collaboration between historians, computer scientists, educationalists and graphics designers.

![Figure 6: Reconstruction of Linlithgow Palace in the 16th Century](image)

The motivation behind the Linlithgow deployment was to help to support the forthcoming teaching of Kings and Queens of Scotland within the history department at
Linlithgow Academy. This fits well with the new national Curriculum for Excellence and its aim to explore topics in a local context to bring them to life. The Curriculum for Excellence [32] promotes exploratory learning and aims to move away from the traditional learning environment to incorporate planned and spontaneous play, which are well supported by the immersive environment offered by virtual worlds.

The original plan, based on previous experiences of adequate graphics capabilities in PCs, and blocked ports on firewalls, was to take an OpenSim server and clients into the school. However, it was found that the school systems were able to work effectively with the Virtual World technologies and the reconstruction. An OpenSim server was installed on the school application server and modified clients were installed on 20 PCs in a computer laboratory. Whilst the wide area network infrastructure and local government firewalls normally prevent access to virtual worlds, social network sites and a host of other web resources, the flexibility of the OpenSim architecture meant that the local school infrastructure could be used.

Pupils from Linlithgow Academy had the opportunity to interact with the virtual reconstruction of Linlithgow Palace in June 2012. Linlithgow Academy has over 1,600 students. The reconstruction was advertised to all classes, and a PowerPoint presentation was displayed on the school information board. Over 100 students actively interacted with the Virtual World during scheduled and drop-in sessions. On the day members of staff from the School of Computer Science and representatives from Education Scotland and Historic Scotland [33] engaged with the students during the sessions.

Pupils produced artwork prior to the event. This had been organised and scanned by the school’s History Department, and then placed within the reconstruction in the form of a web gallery to include user generated content and engage them with the forthcoming event. During scheduled activities more than 60 pupils completed an optional task sheet to reveal a key date and a special location within the palace. They achieved this by locating facts in-world and interacting with historical characters in locations throughout the virtual palace. Finally they voted for their favourite artwork in-world and used three imaginative words to describe the reconstruction.

Another exercise in student engagement made use of chroma keying. It was discovered during the deployment that some students from the school don historical costumes and work as volunteer tour guides at the Palace as it is today (see Fig.7).

![Figure 7: School students working as tour guides](image-url)
A number of these students subsequently took part in creating video images. These volunteers wearing authentic costumes were filmed against Education Scotland’s green screen during a visit to the present day monument (see Fig. 8). The resulting images were then imported into the reconstruction, which placed the guides in their correct context of the Palace as it was in 1567, albeit as rather ghostly presences (see Fig. 8).

![Figure 8: Filming the tour guides using chroma keying and the resulting import into the virtual world](image)

Outwith the scheduled sessions pupils and staff had the opportunity to drop in for free form explorations. The sessions proved very popular and created much excitement within the Library where the event was taking place. Although scheduled events took place in an enclosed area within the Library, two additional areas were provided with more open space allowing staff and pupils, who were not part of the scheduled sessions, an opportunity to use the reconstruction. These areas made use of monitors and projectors displaying the virtual reconstruction where users could explore the Palace using Xbox 360 controllers, and Xbox 360 Kinect (see Figure 9). Several interesting observations were made. The Xbox controllers and Kinect were very popular and proved more intuitive than using the keyboard and mouse. Students were clearly already used to and adept at using such game controllers. Students were very interested in interacting with objects and other users in-world.

![Figure 9: Digital native with Xbox controller; Calibrating Kinect interface](image)
In summary, this immersive educational activity provided a focal point which energized pupils in investigating and exploring events, life and personalities that would have filled the Palace some 500 years ago. In doing so, they in turn enriched the reconstruction as an educational resource and through the medium of film and images were able to communicate with their peers. We found the existing technologies were able to support these activities and therefore this pilot study will be extremely useful in providing insights for use in future educational settings.

6. Conclusion

Virtual Worlds have great potential for educational use, as is evidenced by the growing range of subjects they have successfully been used for. In particular we have noted how OpenSim has come a long way from Second Life in terms of its capability for supporting educational activities. From the point of view of more sophisticated content creation and interactivity better programmability is needed as is illustrated by the use of XMRMs in Routing Island. We have focused on some further considerations as to how virtual worlds could be adopted more widely in education: user guidance for management, revised communication protocols and bandwidth constraints and integration with the Web. We have featured a case study in the area of cultural heritage and history: Linlithgow Palace. The young learners were digital natives and readily engaged with the Virtual World, without thinking of it as an exploratory learning resource, supporting Kolb’s higher order learning activities [21], although that is exactly what it is intended to be. However, considerable effort was needed to support the deployment, especially when compared with using an educational web site. Teachers and local ICT technicians at Linlithgow relied on the expertise of the project team to install the server software, connect the user interfaces, import the pupils’ art work into the Virtual World and configure user names and avatars. In some ways, the question is: how do we make the support team superfluous? A means of accessing the Palace across the Internet would remove the need for local installation – hence the comments made earlier about the poor suitability of the current OpenSim communication protocols in terms of reachability on the global Internet. Teachers and lecturers could learn to create users with avatars and allocate land through advice from the management user guidance methodologies described in section 2. Future work includes streamlining support for deployments in educational settings and encouraging educators at all levels of education to bring Open Virtual Worlds into the learning technology mainstream.

References


