

Managed Learning in 3D Multi User Virtual Environments

I. Perera, C. Allison, J. Ross Nicoll, T. Sturgeon, A. Miller

School of Computer Science, University of St. Andrews
{indika, ca, jrn2005, tommy, alan} @cs.st-andrews.ac.uk

Abstract

Maintaining student engagement is a major concern in higher education, especially when concepts become more sophisticated and coursework becomes more complex. Shared online virtual worlds are attractive in that they have the potential for supporting student engagement through novelty and intrigue whilst providing a programmable environment that can be tailored for educational purposes. This paper presents two case studies illustrating the use of such multi-user virtual environments (MUVE) for education, in the context of credit-bearing assignments in Human Computer Interaction (HCI). While there is considerable interest throughout academia in using MUVEs academics can experience many challenges when exploring various possible use-cases. The studies presented in this paper were conducted to identify some of the critical issues such as student perceptions, privacy, ownership, access to practical work for assessment purposes, maintaining an association between institutional and virtual world identities, and the achievement of learning outcomes through the use of MUVEs for teaching and learning. Through developing an awareness of the challenges that are encountered in MUVEs for managed learning, and describing how to address them, this paper contributes towards their significance as an educational resource which can inspire and engage students.

1. Introduction

Multi-user virtual environments and virtual worlds in general show significant potential for educational activities. They are particularly appropriate for educational use due to their alignment with the concept of experiential learning. Kolb [1] has developed a theory of experiential learning and has stated that the learners build a deep understanding and expertise by cycling through the four steps of the experiential learning cycle: concrete experience, reflective observation, abstract conceptualization, and active experimentation.

As noted in [2], learning through experimentation is a specific type of exploration. It is an approach we seek to inculcate in all systematic learning activity and scientific discovery. Learning processes with experiential activities help to provide student-centric, and more focused, student engagements [3]. Jarmon *et al.* have indicated that virtual worlds are likely to accommodate project-based experiential learning [4]. Dalgarno and others have described how researchers have argued that interactive 3D virtual environments demonstrate great educational potential due to their ability to engage learners in the exploration, construction and manipulation of virtual objects, structures and metaphorical representations of ideas [5]. Accordingly, many higher education courses which looking for novel and engaging approaches to conducting their practical coursework, are interested in the potential of virtual worlds in academia.

There are numerous potential advantages of using virtual worlds in education, either as a supportive tool or as the main platform for teaching. These include better demonstrations of complex scientific concepts, rich media content for learning, greater learning autonomy for students, and facilitating collaborations between physically remote learners [6]. These features are important for the progress of education, especially towards providing a better learning experience. They provide a new dimension in technology-enhanced learning.

Although the courses described in this paper do not inherently involve distance learning, the virtual worlds we chose (Second Life¹, and Open Simulator²) for the allowed students to work from any location in which they had unrestricted Internet access³ and a sufficiently powerful computer. Furthermore, while we have focused on their potential for providing rich educational environments, virtual worlds can also incorporate a games-like entertainment factor into the learning activities, which most other learning approaches lack. Student engagement can therefore be

¹ <http://secondlife.com>

² <http://opensimulator.org/>

³ Where there is an institutional firewall, which is the case on most campuses, Second Life requires many ports to be opened by the relevant authorities. OpenSim can be hosted within a firewall.

much higher with virtual worlds than with other hybrid learning methods [7]. Courses such as human-computer interaction (HCI) require more user-centric thinking than other areas of computing, which tend to be more analytically focused, and can arguably benefit more from the increased student buy-in from the game-like environment.

The remainder of this paper is arranged into 5 sections as follows: section 2 provides some background and related work information; section 3 describes the course environment in which the MUVE was used; section 4 presents our evaluation methodology and student feedback; section 5 elaborates our experience with the Open Simulator environment; section 6 concludes and lists some potential future work.

2. Background and Related Work

In a global context, Second Life (SL) is the most prominent and heavily used MUVE at present and is also the virtual world of choice for learning, teaching and research in UK academia [8]. It is owned and operated on a commercial basis by Linden Labs. The client-access software (the SL viewer) is free, but “land ownership” within Second Life costs real money, on a rental basis. In effect land owners are leasing the use of servers based at Linden Labs which provide a part of the global simulation. The pattern of network traffic differs significantly from massively multi-player online role playing games, such as World of Warcraft (WoW). Second life traffic uses an average of 200 Kbits/second bandwidth in contrast to WoW’s 7 Kbit/second. Furthermore, the downward traffic from the server to each client is the main consumer of this bandwidth, as shown in Fig. 1, copied from [9]. Further SL systems considerations are described and discussed in [9-11].

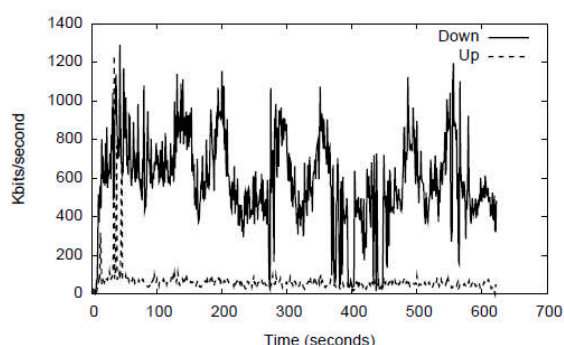


Figure 1: SL network traffic up/down

White [12] explains Second Life as an Internet based multi-user, 3D world construction which emphasizes creativity, collaboration, socializing and

self-government. The Second Life community is a group of residents who collaboratively create, live, and interact in the 3D world which is owned and operated by Linden Labs. The world (or “grid”) is composed of numerous different regions (also referred to as islands, sims or simulators).

Users in virtual worlds such as Second Life can use powerful modelling tools to create and modify world content that can then be shared between students, or even with the “outside world”.

Various educational projects at the University of St Andrews have used virtual environments in their course delivery. These include LAVA [13, 14] and WiFiSL [15]. MMS, the Module Management System, [16, 17], is an online learning management system which interoperates with Second Life in order to maintain an association of institutional and virtual world identities as one of its many features. The Laconia Acropolis Virtual Archaeology (LAVA) project [13, 14] allows students to engage with a simulated archaeological excavation, and then explore a recreation of the site in Second Life. The WiFi Virtual Laboratory in Second Life project (WiFSL) [15] aids teaching and learning about wireless networking by using virtual world interfaces to collaboratively explore and visualise simulations of wireless traffic.

Despite a variety of limitations, which are discussed later, the use of SL at St Andrews exemplifies a growing confidence from academia in utilising virtual worlds as a mainstream method of fostering knowledge construction in a student-centred and experimental environment.

3. The Course Environment

An HCI course is available to final year students as part of the BSc degrees in Computer Science and Internet Computer Science. Amongst other topics it is intended to teach students about user interface design, interaction design, evaluation, and the presentation of visual information. It is targeted specifically at final year students due to their breadth of experience and programming skills.

de Freitas and Neumann report that user interfaces need to become more intuitive following the requirements of the individual learner and reinforcing the drive towards more personalised learning and greater learner autonomy [7]. Concomitantly, since one of the major learning objectives for this course is to give students an opportunity to understand how to incorporate attractive and creative ideas in interface development, the use of 3D virtual learning environments for learning activities provides a suitable platform for their endeavours.

An effective learning strategy should give the students opportunities to practice with the learning environment and its tools. Accordingly, early in the course, students were given introductory sessions covering the basics of using Second Life and the Linden Scripting Language (LSL). LSL is a platform-specific language used to provide interactivity within Second Life, primarily used to allow objects to react to events occurring to/near them.

Although by default new users (residents) in Second Life “arrive” on a small island used for training them in using the client, we found the generic service was inappropriate for educational use and provided our own training in place of this. As this coursework was credit-bearing it was important that all the usual assessment safeguards were in place. In order to maintain a secure association between the student’s institution identity and their SL identity, the registration API (RegAPI) was used via an interface created within MMS [16, 17], the institutional learning management system, to allow students to pick a name and create an avatar. This also allowed us to ensure that the students arrived directly on the region “owned” by the Computer Science department, and had immediate access to their own parcels of land.

In addition to various pre-allocated parcels of land there was also a meeting/demonstration area for Computer Science (see Fig.2), a sandbox where students could experiment with content creation/manipulation outwith their own parcels, and the LAVA and WiFi projects.

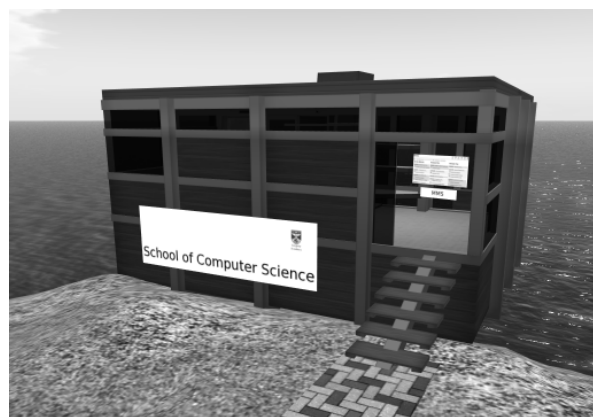


Figure 2: Computer Science's communal meeting space in Second Life

3.2 Project for Assessment

Usability is a major theme within HCI courses, and is also key to the efficacy of computer-based learning resources. As such, the final year undergraduates on the HCI course were asked to construct an interactive learning resource which could be used in 1st and 2nd year courses to aid learning about a specific algorithm.

Dijkstra’s shunting algorithm [18] is often taught in 1st or 2nd year Computer Science courses, but like many useful algorithms, can leave some students bored and disengaged when presented in an abstract context, so the goal was to demonstrate expertise in constructing a user-friendly learning resource to engage students who would otherwise lose interest. The algorithm is used to convert mathematical equations in infix notation to postfix notation. For example:

- “1 + 2” becomes “1 2 + ”
- “1 + 2 * 3” becomes “1 2 3 x + ”
- “(1 + 2) x 3” becomes “1 2 + 3 x ”

Postfix notation is frequently used for stack-based evaluation. An optional part of the HCI assignment was to show the stack-based evaluation of the postfix notation.

Students could build and script within their own private parcels of land with minimal restrictions imposed on them. This allowed them the ability to exclude some or all other residents, and many chose to do so to ensure they were not disturbed. Figure 3 demonstrates the appearance of the boundary of one parcel which prohibits others from accessing it. Note the “No Enter” labelled transparent surface that prevents entry.

Students were deliberately given limited guidance on the expected form or operation of their implementation, with the hope that this would encourage creative and novel implementations.



Figure 3: Restricted access – students cannot enter others’ work area

3.3 Coursework Issues and Considerations

There are several issues involved with assigning a practical on a new platform. Providing the students each with their own parcel of land on the Computer Science region meant that they had a safe area to work in, with the ability to block other residents from entering it. However, Second Life does not provide any functionality for hiding in-world content from other residents, so it cannot prevent them seeing their colleague’s works or creative concepts. This was considered a reasonable limitation for a creative course

environment, as the scripts themselves were still protected.

A significant opportunity with the Second Life virtual world is the ability to create 3D content with scripting for automation. If the same assigned task had been tried in another programming environment students would have had to limit their creative concepts due to GUI or API limitations. However, the students still had to learn Second Life scripting for their content automation, which takes a reasonable time. This was not a serious problem for final year computing students but could be troublesome for those with less experience in programming.

4. Feedback and Evaluation

At the end of the project most students had constructed an interactive version of the shunting algorithm that worked correctly and many had produced highly novel and engaging user interfaces. Figure 4 summarises the grades for the coursework. It shows that ten out of thirteen students obtained high grades, with eight of them achieving distinction level.

Their ideas of system design and interfacing were fascinating and demonstrated the potential of MUEs for educational resource construction. Examples are shown in Figures 5a and 5b.

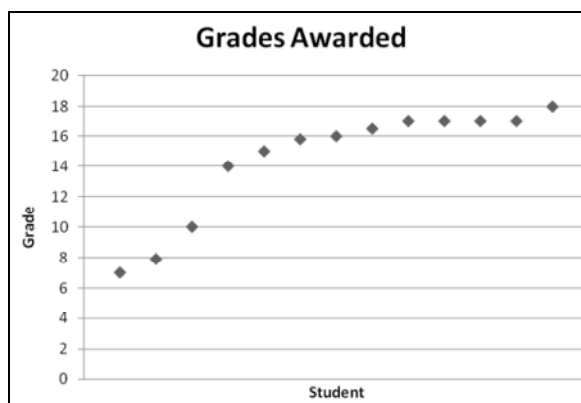
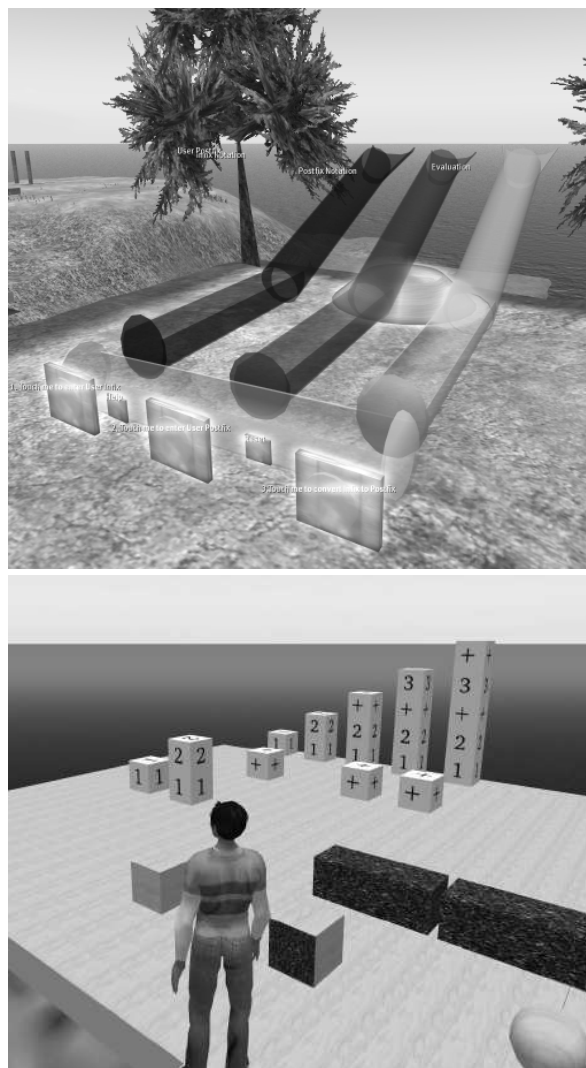


Figure 4: HCI class grades: ten out of thirteen students scored high grades, with eight of them achieving distinction level

4.1 Student Feedback

After the end of the course, students were asked to complete a simple questionnaire about their experience with Second Life. The questionnaire is based on the system usability scale [19] (SUS) as we were primarily interested in how comfortable the students were with Second Life. As with the SUS, it was delivered as ten statements. The statements were written in pairs, one phrased positively, one phrased negatively, to ensure balance, and students were asked to rate their "agreement" with each statement on a scale of 1-5.



Figures 5a and b: implementations of strange and wonderful machines for learning about Dijkstra's shunting algorithm.

The statements were:

1. I think that I would like to use Second Life frequently
2. I found Second Life unnecessarily complex
3. I would like to use Second Life for meetings with other students and staff
4. I think that I would need the support of a technical person to be able to use Second Life
5. I found the content in Second Life interesting
6. I think that Second Life would be unsuitable for assessed coursework
7. I would imagine that most people would learn to use Second Life very quickly
8. I found Second Life disorientating
9. I felt very confident using Second Life
10. I needed to learn a lot of things before I could get going with Second Life

Statements #1 and #3 were included to evaluate students' willingness to use Second Life. This helps us to decide whether the learning environment was attractive to students and whether they were willing to explore more usages in the future. The opposite statements were #8 and #6 respectively. Statement #6 was of particular importance as it helped evaluate student perception of the specific area of the course assessment. Assessment methods have to be acceptable to students to be effective [20]. A virtual world assignment can be a failure if the assessment is not effective and convincing even though the rest of the course is successful. In order to inform the use Second Life with other courses, questions such as those outlined above would indicate how feasible that is.

Statements #5, #7 and #9 were included with the objective of assessing student difficulties with respect to using Second Life. If the underlying learning environment is too complex students tend to learn more about the environment but not enough of the course content during their studies. Statements #5, #7 and #9 were paired against statements #2, #4 and #8 respectively.

Unfortunately, response to the questionnaire was too low to be meaningful on this occasion, due entirely to the date of issue, which was just after the final year students had completed their entire degree course. Without quantifiable results, we were forced to infer from the reflective feedback written directly into the student's reports (which were a required part of their coursework). Some example quotations:

"Creating a physical implementation in a virtual world is somewhat trickier than was first expected. As you have to account for the different timings and the unexpected occurrences of tokens leaving the machine."

"The hardest part of this practical was getting the second life physics engine to work properly."

"The visual aid which has been produced is good at displaying the way which Dijkstra's "shunting yard" algorithm works using a stack. It clearly shows the stages that are gone through from parsing the input to evaluation."

"...the Second Life physics engine is not very consistent or realistic."

"The most difficult aspect of this practical was programming in the Second Life environment and LSL. Although potentially quite powerful many of its features and functionality are poorly implemented and most of the practical time was spent diagnosing and correcting unexpected behaviour."

While we expected the students to avoid the physics engine for reasons of simplicity, the majority in fact successfully used it for their practical. It was a major source of issues however. Interestingly, they seemed mostly happy with the LSL language (which can be frustrating for its lack of language features such as complex data types or exception handling).

4.2 Assessment Issues

Assessing work performed in a virtual world provided several interesting challenges. The first and most obvious was how to associate students' real identities with their in-world identities, in order to assign marks for work submitted in-world to the correct student. This was done, as previously mentioned, by having students create Second Life accounts through a the MMS learning management system, which created the accounts through the regAPI provided by Linden Labs. All associations were recorded in the MMS learning management system.

Submission of work proved confusing and tricky for students. Although given instructions to give their work to an avatar account created specifically for only this purpose, several had to ask for help with this. They were uncertain how to transfer work to another avatar in-world, and how to convert items in-world into a form that could be "rezzed" and used easily elsewhere. This showed a need to either provide more sophisticated facilities for "handing in" work, or to teach the students more in-depth background on the Second Life platform and its underlying model of ownerships, permissions and inventory management.

Once work had been submitted, further problems arose. The lecturer in charge of assessing the coursework received e-mails from Second Life notifying them of assets being transferred to their account, but these never actually arrived. Second Life provides no delivery notification to the sender, so there was no way for students to be made aware of this. The markers also had to trust that they received notifications of every asset sent to them, as Second Life has no ability to prove an item was sent.

For the assets that were delivered, further unanticipated problems were encountered. Students had not realised some of the implications of the permissions system which restricts the ability to copy, transfer and/or modify assets. By default, assets were created with the owner able to do all of these, but others could only transfer the assets. This is intended as a copy-control mechanism, so that content purchased in-world cannot be copied or transferred freely, but resulted in several pieces of coursework arriving in a non-functional state. Specifically, a lot of them depended on "rezzing" assets from object inventory, which without copy permissions enabled meant the rezzed object was removed from inventory as a side-effect. Scripts that expected to be able to "rezz" multiple copies of the same object failed due to this.

In the end we abandoned using the default in-world means for transferring objects, and wrote our own coursework submission tool. Students "bought" this for free, and then put their object into the submission objects inventory for delivery. It then checked that the name of the submission was meaningful (to help the marker), and that it had copy, transfer and modify permissions enabled for the next owner. Once these checks had been passed successfully, it requested confirmation from the student and then delivered the

object to the lecturer. We would hope later versions could produce a receipt, so students can prove that they sent some form of object (and when), in case of hand-in problems.

5. Experience with Open Simulator

Having experienced the above mentioned issues with Second Life, we decided to create a virtual world using Open Simulator, to examine its appropriateness as a learning environment. The Open Simulator Project (OpenSim) is an open source project which can be used to support self-hosting virtual worlds with local administration and maintenance. OpenSim is of particular interest because it has a high degree of compatibility with the Second Life client, which is available on Windows, Linux and OS X (in comparison to most virtual world clients which are single-OS only).

We hosted an OpenSim based test 3D MUVE with an objective to use it for the HCI course work. We are also in the process of transferring existing projects (LAVA, WiFiSL) from Second Life to OpenSim for the purpose of experimentation. With the flexibility to have multiple regions, we created a dedicated region for the next cohort of students, simply named 'HCI'.

5.1 HCI Coursework in OpenSim

With the next student cohort for the same HCI course module, it was decided to give less weight to the virtual world component of their continuous assessment. Considering the reduced grading weight, students were asked to individually implement an interactive door system for an enclosure of their own creation. The enclosure system was expected to involve at least two doors - one from simple types and another from moderate/hard categories as listed in Table 1. Students were given the flexibility to incorporate their own creative design ideas, while meeting the course project requirements.

Students were assigned a parcel of land within the dedicated 'HCI' region in the OpenSim environment, giving them a space within which they could build and script with minimal restrictions imposed on them. Each parcel was a 2500 sq. m (50m x 50m) piece of land, with exclusive ownership for the assigned student. (This was approximately 10 times larger than could be afforded to them by the relatively heavily used Second Life region). There were 18 students registered for the course module; hence an area of 45,000 sq. m. was allocated for student parcels, and the remaining land (20,536 sq. m.), kept as a sandbox, to support student collaboration through a shared place.

Students could modify their parcel land and terrain to suit their implementation. Even though the assessment criterion does not consider the effort they have spent for the enclosure creation, almost all the students were keen to come up with an innovative and unique design for their work. Most of the projects

successfully incorporated two or more doors from the different types listed in Table 1. Two interesting door designs are shown in Figure 6a and b.

Type of Door	Difficulty	Explanation
Touch Door	Simple	A simple door that when touched moves or rotates a certain amount
Sense Door	Simple	The door senses objects in the vicinity and opens when detected
Password Required	Simple	When the user types in a password on a certain channel the door opens for a fixed period.
FOB Access	Simple	The door can sense objects in the vicinity, a SLID can be specified that allows the door to open
Guest List	Simple	The door can sense avatars in the vicinity, and only opens if the avatar's ID matches one in the list.
Force field	Simple	A Door that when access is granted sets itself to be a phantom object for a fixed period then back to physical.
Push sensitive & revolving	Moderate	The door can sense a collision (mimicking a person pushing a revolving door) and then opens
Keypad Entry	Moderate	The user must enter the correct code to gain access
Escape Door	Moderate	Designed to look like a fire escape door, when pushed the buildings fire alarm goes off with lights etc.
Double Sliding	Moderate	Creating two doors that can slide apart.
Garage Door	Moderate-Hard	Based on the same premise of garage doors, with a challenge and response.
Combination access	Hard	This would model a safe door, the user can turn the dial x right, x left etc. till the correct combination

Table 1: Different door systems and their difficulty for HCI project work

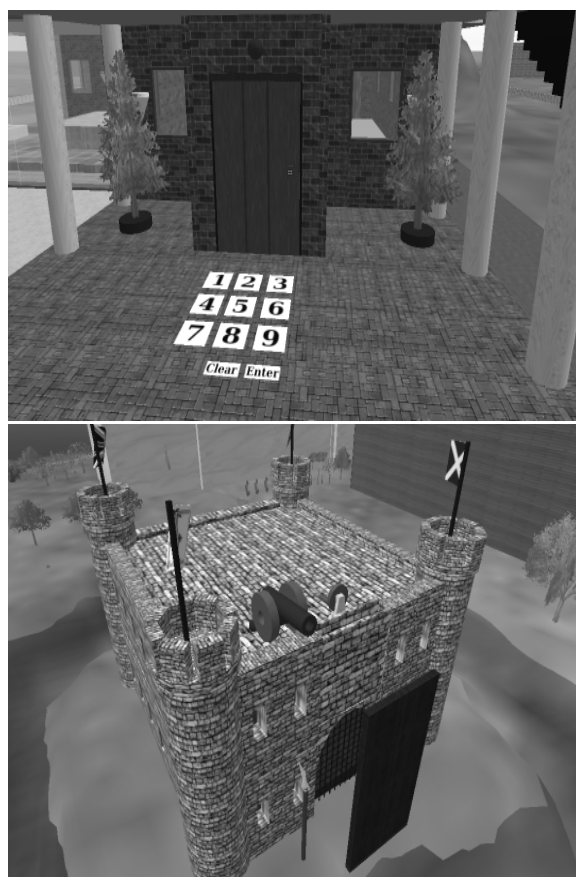
For this project assessment, we asked students to submit a report with a description of their project implementation and their experience. During the evaluation we found that some students were keen to develop more sophisticated designs to show their ability to implement complex use cases, instead of focussing more on HCI principles and best practices. On the other hand, this indicates students' motivation towards the advanced content creation over the stipulated minimum requirements for the projects. Overall, none of the students came up with poor or trivial work, and there were some outstanding efforts. All in all this reinforces our belief that virtual worlds have great educational potential through their ability to engage students.

5.2 Experiences with OpenSim

The Open Simulator software is still being developed, and the project is yet to reach the alpha state; the latest release version is 0.6.9. Being a developing project, the OpenSim has a reasonable amount of system implementation and testing to be done according to its development community. There

are known issues as well as emerging issues when it is used for educational activities.

LSL is the default scripting language in OpenSim, even though it is possible to use C#, J#, or VBScript through environment configuration. One advantage with using one of these languages is that we could use advanced data types and exceptions, which are not available with LSL. Before the students started their work, we did some tests to compare the benefits and issues with using C#, instead of LSL. We found that some of the LSL functions are yet to be implemented in C#, and that the access to static method calls such as 'System.IO.File' is available, creating a large security hole. With this observation, we decided to use LSL for the student projects.



Figures 6a and b: implementations of mechanisms for interactive door systems

As indicated earlier, land parcel access protection is a requirement for an assessed learning environment. OpenSim seems to support the access restriction for land parcels through land management. Once the public access restriction is set, the parcel shows a virtual fence with text "No Entry", as we experience in Second Life. However, other users still could go through this virtual fence without any difficulty! As an alternative solution, a more restrictive approach was followed, i.e. the *banned residents* methods. In the banned residents method, we set a list of users who are banned from a

parcel. This was done for all the parcels and tested for each user, individually using their logins. Even though it worked during the test cases, once everybody entered the region during the induction session, it failed! Fortunately, none of the students were concerned about allowing access to each others' parcels. However, there are learning activities where the parcel access restriction is essential.

5.3 Student Feedback

The questionnaire detailed in section 4.1 was given to students for their feedback. 15 students responded. The mean and standard deviation values for each statement are shown in the Table 2.

Question	Mean	St. Dev
1. I think that I would like to use OpenSim frequently	2	0.93
2. I found OpenSim unnecessarily complex	3.27	1.22
3. I would like to use OpenSim for meetings with other students and staff	1.93	0.7
4. I think that I would need the support of a technical person to be able to use OpenSim	2.33	1.11
5. I found the content in OpenSim interesting	2.4	1.12
6. I think that OpenSim would be unsuitable for assessed coursework	3.2	0.94
7. I would imagine that most people would learn to use OpenSim very quickly	2.27	0.8
8. I found OpenSim disorientating	3.33	1.05
9. I felt very confident using OpenSim	2.73	1.03
10. I needed to learn a lot of things before I could get going with OpenSim	3.07	1.1

Table 2: Student feedback summary on usability of the OpenSim HCI project work

An initial observation is based on the report in [21] which states that it is important to provide an easily useable technological platform for virtual learning activities. The responses for statements #1 and #3 indicate that students' willingness to use OpenSim is below the average norm (3) of the scale. This helps us to know that the environment should be more attractive to students for their future learning activities. The opposite statements, #8 and #6 respectively, show above average values. As described above, Statement #6 was of particular importance since the response indicates that there should be some form of student confidence building of the course assessment. Responses to the statements #5, #7 and #9 are below the average norm (3) indicating students are more towards to disagree with the statements. Even though the statements #2, #4 and #8 are the counter statements respectively, only #2 and #8 shows agreeing response mean values. The statement #4 has a disagreeing mean value; main reason for this is that the students have enough technical knowledge to understand the system without extra help as they are CS honours students. However, for a non IT/CS student sample, extra technical assistance could be essential.

The model behind the SUS specifies how a “quick and dirty” system usability figure can be calculated, in a range from 0 to 100, where the best usability would be 100. For this analysis, the system usability value was 40.3, which indicates that we need to improve the system usability aspects of the OpenSim based learning activities for a better learner experience.

Through discussions with the students the problems with their experience were all traced back to one main reason: there were system crashes which they perceived as frequent. These in turn were found to have been caused by stack overflow exceptions when the physics engine was used. As OpenSim is being still being developed, we can reasonably expect more stability and usability enhancements with future releases, which will provide educators better options.

6. Conclusions and Future work

Students were successful in producing useable, interactive learning resources for Dijkstra's shunting algorithm, and rose to the challenge the Second Life platform presented as a programming environment. However, Second Life left an impression with them of being unpredictable and unreliable, and project work had to be submitted two or three times due to problems with the ownership and protections model. So, while the MUVE concept proved popular and successful, the use of Second Life per se left a lot to be desired. Accordingly, used the OpenSim environment for the next cohort of students. Although students were able to freely develop their content in an attractive and interactive manner OpenSim needs more maturity and reliability to provide an effective MUVE environment for learning. We think this can realistically be achieved with future releases.

One further study would be to research how students' perception of using 3D MUVE, in this case the Second Life or the OpenSim, for their post-University activities. As the learning suggests permanent knowledge occupancy in the participants mind, it would be a worthy effort to examine whether students used the MUVE for the sake of course requirements or if they had built up their own interest and will tend to use it for their future requirements.

A further route for evaluation would be to provide 2nd year algorithms students with access to one or two of the interactive learning resources built by the 4th year students.

In summary we fully intend to continue exploring the use of virtual worlds in education. Going forwards we are enthusiastic about looking at platforms such as OpenSim that can be developed to suit educational requirements.

6. Acknowledgements

Second Life region rental was supported in part by the University of St Andrews *Fund for Innovations in Learning, Teaching and Assessment* (FILTA). The *Higher Education Academy for Information and Computer Sciences* (HEA ICS) supported part of the work on OpenSim.

7. References

- [1] D. A. Kolb, R. E. Boyatzis, and C. Mainemelis, "Experiential Learning Theory: Previous Research and New Directions," in *Perspectives on Thinking, Learning and Cognitive Styles*, J. Sternberg and L.-F. Zhang, Eds. Mahwah, NJ: Lawrence Erlbaum, 2001, p. 227.
- [2] C. Allison, A. Miller, K. Getchell, and T. Sturgeon, "Exploratory Learning for Computer Networking in ICWL 2007," *Lecture Notes in Computer Science*, vol. 4823/2008, pp. 331-342, 2008.
- [3] S. Barab, M. Thomas, T. Dodge, R. Carteaux, and H. Tuzun, "Making Learning Fun: Quest Atlantis, A Game Without Guns," *Educational Technology Research and Development*, vol. 1, pp. 86-107.
- [4] L. Jarmon, T. Traphagan, M. Mayrath, and A. Trivedi, "Virtual world teaching, experiential learning, and assessment: An interdisciplinary communication course in Second Life," *Computers & Education*, vol. 53, pp. 169-182, 2009.
- [5] B. Dalgarno, A. G. Bishop, W. Adlong, and J. Danny R. Bedgood, "Effectiveness of a Virtual Laboratory as a preparatory resource for Distance Education chemistry students," *Comput. Educ.*, vol. 53, pp. 853-865, 2009.
- [6] E. Prasolova-Førland, "Analyzing place metaphors in 3D educational collaborative virtual environments," *Computers in Human Behavior*, vol. 24, pp. 185-204, 2008.
- [7] S. d. Freitas and T. Neumann, "The use of 'exploratory learning' for supporting immersive learning in virtual environments," *Comput. Educ.*, vol. 52, pp. 343-352, 2009.
- [8] J. Kirriemuir, *"The spring 2009 snapshot of virtual world use in UK higher and further education,"* Bath, UK: Eduserv Foundation, 2009.
- [9] I. Oliver, A. Miller, and C. Allison, "Virtual Worlds, Real Traffic: Interaction and Adaptation," in *ACM Multimedia Systems 2010*, Scottsdale, Arizona, 2010.
- [10] Sanjeev Kumar, Jatin Chhugani, Changkyu Kim, Daehyun Kim, Anthony Nguyen, Pradeep Dubey, Christian Bienia, and Youngmin Kim, "Second Life and the New Generation of Virtual Worlds," *Computer*, vol. 41, pp. 46-53, September 2008.
- [11] J. Waldo, "Scaling in games and virtual worlds," *Communications of the ACM* vol. 51, pp. 38-44, August 2008.
- [12] B. A. White, *Second Life: A Guide to Your Virtual World*: Que, 2007.

- [13] K. Getchell, A. Miller, C. Allison, C. Kerbey, R. Hardy, R. Sweetman, V. Crook, and J. Complin, "The LAVA Project: A Service Based Approach to Supporting Exploratory Learning," in *IADIS International Conference WWW/Internet*, Murcia, Spain. 2006, 2006.
- [14] K. Getchell, J. Nicoll, C. Kerbey, A. Miller, C. Allison, R. Sweetman, and R. Michaelson, "Evaluating exploratory learning in LAVA," in *Proceedings of the sixth conference on IASTED International Conference Web-Based Education - Volume 2* Chamonix, France: ACTA Press, 2007.
- [15] T. Sturgeon, C. Allison, and A. Miller, "802.11 wireless experiments in a virtual world," *ACM SIGCSE Bull.*, vol. 41, pp. 85-89, 2009 2009.
- [16] C. Allison, A. Bain, B. Ling, and R. Nicoll, "Addressing Academic Needs in Managed Learning Environments," in *4th Annual LTSN-ICS Conference*, Galway, Ireland, 2003, pp. 227-232.
- [17] C. Allison, A. Bain, B. Ling, and R. Nicoll, "MMS: A User Centric Portal for eLearning," in *14th int. workshop on database and expert systems applications*, Prague, Czech Republic, 2003, pp. 292-297.
- [18] E. W. Dijkstra, "Making a Translator for ALGOL 60," *ALGOL Bulletin Supplement*, vol. 10, pp. 21-32, 1961.
- [19] J. Brooke, "SUS - A quick and dirty usability scale," 1996.
- [20] J. Macdonald, "Assessing online collaborative learning: process and product," *Computers & Education*, vol. 40, pp. 377-391, 2003.
- [21] G. I. U. S. Perera, "Key Success Factors for e-Learning Acceptability: A Case Based Analysis on Blended Learning End-User Experience," in *IEEE International Advance Computing Conference, IACC'09*, 2009, pp. 2379-2384.