AN EVALUATION OF USER SUPPORT STRATEGIES FOR MANAGED LEARNING IN A MULTI USER VIRTUAL ENVIRONMENT

Galhenage Indika Udaya Shantha Perera

A Thesis Submitted for the Degree of PhD at the University of St Andrews

2013

Full metadata for this item is available in Research@StAndrews:FullText at:
http://research-repository.st-andrews.ac.uk/

Please use this identifier to cite or link to this item:
http://hdl.handle.net/10023/4049

This item is protected by original copyright
An Evaluation of User Support Strategies for Managed Learning in a Multi User Virtual Environment

Galhenage Indika Udaya Shantha Perera

PhD Thesis

A thesis submitted for the degree of Doctor of Philosophy
School of Computer Science
University of St Andrews

July 2013
Abstract

The management of online learning environments so that they are effective and efficient presents a significant challenge for institutions and lecturers due to the complexity of requirements in the learning and teaching domain. The use of 3D Multi User Virtual Environments (MUVEs) for education introduces a novel set of management challenges. MUVEs were designed to cater for entertainment and commercial needs and as such do not intrinsically support managed learning. When MUVEs are used for educational purposes, forming 3D Multi User Learning Environments (MULEs), user support for learning management becomes an important factor.

This thesis highlights the importance of managed learning in MULEs. It proposes a coordinated approach which accommodates the existing education institutional infrastructure. The research has focused on two very widely used and closely compatible MUVEs, Second Life (SL) and OpenSim. The thesis presents system and user studies that have been carried out on these selected MUVEs. The findings reveal the challenges that academics and students can experience if they do not have sufficient knowhow to manage learning activities in SL/OpenSim. User guidance and training tools were then developed for supporting learning management strategies in the context of SL/OpenSim and demonstrated in exemplar use-case scenarios.

The user support models and tools which were developed have been extensively evaluated for their usability and educational value using diverse participant groups. The results validate the efficacy of these contributions, defending the research thesis. These contributions can be used in future research on managing MUVE supported education.
Acknowledgement

I owe my gratitude to many people who have made this research possible.

My deepest gratitude is to my supervisor, Dr. Colin Allison for the opportunities and the support he gave. I was very fortunate to have him as my supervisor. His valuable input and the flexibility in shaping the best path for my research interests have helped me a lot for this work. I really appreciate his support extended to develop my research and academic competencies.

I would like to thank my second supervisor, Dr. Alan Miller, and my reviewers Dr. Mark-Jan Nederhof and Dr. Alex Voss, for their support and feedback.

I highly appreciate the financial support received from the Commonwealth Scholarship Programme (UK) and the Scottish Informatics and Computer Science Alliance (SICSA) to complete this work.

I am thankful to Dr. Iain Oliver, Dr. Thomas Sturgeon, Dr. Ying Zhai, Dr. Meng Yu, Dr. Ittipong Khemapech, Mr. John McCaffery and Mr. Ross Nicoll for the collaborative research, inspirational discussions and knowledge sharing.

I would like to thank my friends and the staff at the School of CS. It was a wonderful experience to work with them.

I am grateful to my family for their love and kind support throughout.

Thank you.
Declarations

I Galhenage Indika Udaya Shantha Perera, hereby certify that this thesis, which is approximately 66,000 words in length, has been written by me, that it is the record of work carried out by me and that it has not been submitted in any previous application for a higher degree. I was admitted as a research student and a candidate for the degree of Doctor of Philosophy in September 2009; the higher study for which this is a record was carried out in the University of St Andrews between 2009 and 2013.

Date
Signature of candidate

I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate for the degree of PhD in Computer Science in the University of St Andrews and that the candidate is qualified to submit this thesis in application for that degree.

Date
Signature of supervisor

In submitting this thesis to the University of St Andrews I understand that I am giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. I also understand that the title and the abstract will be published, and that a copy of the work may be made and supplied to any bona fide library or research worker, that my thesis will be electronically accessible for personal or research use unless exempt by award of an embargo as requested below, and that the library has the right to migrate my thesis into new electronic forms as required to ensure continued access to the thesis. I have obtained any third-party copyright permissions that may be required in order to allow such access and migration, or have requested the appropriate embargo below.

The following is an agreed request by candidate and supervisor regarding the electronic publication of this thesis:

Access to printed copy and electronic publication of thesis through the University of St Andrews.

Date
Signature of candidate

Signature of supervisor
Contents

1. Introduction ........................................................................................................................................... 1
   1.1 Research Overview .......................................................................................................................... 1
   1.2 Thesis Statement ............................................................................................................................... 1
   1.3 Research Objectives and Methodology ............................................................................................ 2
       1.3.1 System Environment for the Research ..................................................................................... 2
   1.4 Research Contributions .................................................................................................................... 3
       1.4.1 Overview of Research Contributions ......................................................................................... 3
       1.4.2 Selected Publications .................................................................................................................. 4
   1.5 Thesis Structure ................................................................................................................................. 6
   1.6 Summary ........................................................................................................................................... 6

2. Review of Literature ................................................................................................................................. 7
   2.1 Introduction ....................................................................................................................................... 7
   2.2 Multi User Virtual Environments ...................................................................................................... 7
       2.2.1 Game Environments - MMORPGs ........................................................................................... 7
       2.2.2 MUVEs – an Overview ............................................................................................................. 8
       2.2.3 Related Terminology of MUVEs .............................................................................................. 11
       2.2.4 Management Challenges in MUVEs ....................................................................................... 12
   2.3 Technology Enhanced Learning ......................................................................................................... 13
       2.3.1 E-Learning ............................................................................................................................... 13
       2.3.3 Other Approaches ..................................................................................................................... 14
       2.3.4 Blended Learning ...................................................................................................................... 15
   2.4 Teaching and Learning with MUVEs ................................................................................................. 16
       2.4.1 Related Educational Projects .................................................................................................. 19
   2.5 Educational Theories for MUVE based Learning ................................................................................ 20
       2.5.1 Generic Learning Taxonomies and Models .............................................................................. 20
       2.5.2 Experiential Learning .............................................................................................................. 23
       2.5.3 Exploratory Learning ................................................................................................................. 24
       2.5.4 Constructivist Learning ............................................................................................................ 25
       2.5.5 Collaborative and Corporate Learning ...................................................................................... 26
       2.5.6 Constructive Alignment ............................................................................................................ 27
       2.5.7 3P Model .................................................................................................................................. 27
   2.6 Management of Learning Environments ............................................................................................. 28
       2.6.1 Security and Trust in Learning Environments ............................................................................. 28
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.6.2</td>
<td>Management Approaches in e-Learning</td>
<td>30</td>
</tr>
<tr>
<td>2.6.3</td>
<td>Summary</td>
<td>31</td>
</tr>
<tr>
<td>2.7</td>
<td>Behavioural and System Usability Aspects for Learning</td>
<td>31</td>
</tr>
<tr>
<td>2.7.1</td>
<td>Learning Environment Usability</td>
<td>31</td>
</tr>
<tr>
<td>2.7.2</td>
<td>Self-regulatory Learning</td>
<td>33</td>
</tr>
<tr>
<td>2.8</td>
<td>Concluding remarks</td>
<td>34</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
<td>35</td>
</tr>
<tr>
<td>3.2</td>
<td>Second Life</td>
<td>35</td>
</tr>
<tr>
<td>3.3</td>
<td>Open Simulator</td>
<td>37</td>
</tr>
<tr>
<td>3.4</td>
<td>Browser Applications</td>
<td>39</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Second Life Viewer</td>
<td>39</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Hippo OpenSim Viewer</td>
<td>40</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Phoenix and Firestorm Viewers</td>
<td>40</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Other Viewers</td>
<td>41</td>
</tr>
<tr>
<td>3.5</td>
<td>Overview of SL and OpenSim System Models</td>
<td>41</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Access Control and User Management</td>
<td>42</td>
</tr>
<tr>
<td>3.5.2</td>
<td>Land Management Models</td>
<td>42</td>
</tr>
<tr>
<td>3.5.3</td>
<td>Content Management Models</td>
<td>43</td>
</tr>
<tr>
<td>3.5.4</td>
<td>Group Management Models</td>
<td>49</td>
</tr>
<tr>
<td>3.5.5</td>
<td>Avatar Activities</td>
<td>52</td>
</tr>
<tr>
<td>3.6</td>
<td>Challenges for Learning</td>
<td>53</td>
</tr>
<tr>
<td>3.6.1</td>
<td>An Approach to Manage Avatar Appearance Change</td>
<td>56</td>
</tr>
<tr>
<td>3.7</td>
<td>MUVEs in Student Learning – User studies</td>
<td>57</td>
</tr>
<tr>
<td>3.7.1</td>
<td>User Study in Second Life</td>
<td>58</td>
</tr>
<tr>
<td>3.7.2</td>
<td>User Study in OpenSim</td>
<td>59</td>
</tr>
<tr>
<td>3.8</td>
<td>Concluding remarks</td>
<td>62</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>63</td>
</tr>
<tr>
<td>4.2</td>
<td>A Model for Blended Learning with MULEs</td>
<td>63</td>
</tr>
<tr>
<td>4.2.1</td>
<td>The Model – Strategic View</td>
<td>63</td>
</tr>
<tr>
<td>4.2.2</td>
<td>An Analysis of Use Cases for a Blended MULE</td>
<td>65</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Use Case Mapping Challenges – A Case Study</td>
<td>71</td>
</tr>
<tr>
<td>4.2.4</td>
<td>A Case Study on a Learning Aid within Different Domains</td>
<td>75</td>
</tr>
<tr>
<td>4.3</td>
<td>Important Management Aspects for a MULE</td>
<td>78</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>7.7.1 Study Limitations</td>
<td>192</td>
<td></td>
</tr>
<tr>
<td>7.8 Concluding Remarks</td>
<td>192</td>
<td></td>
</tr>
<tr>
<td>8. Conclusion</td>
<td>193</td>
<td></td>
</tr>
<tr>
<td>8.1 Introduction</td>
<td>193</td>
<td></td>
</tr>
<tr>
<td>8.2 Research Review and Contributions</td>
<td>193</td>
<td></td>
</tr>
<tr>
<td>8.2.1 Thesis Review</td>
<td>193</td>
<td></td>
</tr>
<tr>
<td>8.2.2 Objective Achievements and Overview of Contributions</td>
<td>194</td>
<td></td>
</tr>
<tr>
<td>8.3 Future Work</td>
<td>195</td>
<td></td>
</tr>
<tr>
<td>8.4 Concluding Remarks</td>
<td>197</td>
<td></td>
</tr>
<tr>
<td>Reference</td>
<td>199</td>
<td></td>
</tr>
<tr>
<td>Appendices</td>
<td>213</td>
<td></td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Training support for MULE management – Management Island</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>SL and OpenSim region growth</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>LAVA project in OpenSim</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>The virtual reconstruction of the St. Andrews Cathedral in OpenSim</td>
<td>20</td>
</tr>
<tr>
<td>2.4</td>
<td>Bloom's Model - Cognitive Domain</td>
<td>21</td>
</tr>
<tr>
<td>2.5</td>
<td>MUVE learning affordance model proposed by Gamage et al.</td>
<td>23</td>
</tr>
<tr>
<td>2.6</td>
<td>The Kolb’s model of experiential learning</td>
<td>24</td>
</tr>
<tr>
<td>2.7</td>
<td>Zone of Proximal Development model of learning</td>
<td>25</td>
</tr>
<tr>
<td>2.8</td>
<td>The 3P Model of Teaching and Learning (adapted from Biggs, 2001)</td>
<td>28</td>
</tr>
<tr>
<td>2.9</td>
<td>Content development session in Second Life</td>
<td>36</td>
</tr>
<tr>
<td>2.10</td>
<td>OpenSim system architectural modes: Standalone and Grid</td>
<td>38</td>
</tr>
<tr>
<td>2.11</td>
<td>Second Life viewer - version 3.3.4 (262321)</td>
<td>39</td>
</tr>
<tr>
<td>2.12</td>
<td>Hippo OpenSim Viewer – version 0.6.3</td>
<td>40</td>
</tr>
<tr>
<td>2.13</td>
<td>Phoenix Viewer 1.6.1 (1691)</td>
<td>41</td>
</tr>
<tr>
<td>2.14</td>
<td>World Map of several regions (SL): A square is a region of 65,536m²</td>
<td>43</td>
</tr>
<tr>
<td>2.15</td>
<td>Client permission settings options for different roles</td>
<td>47</td>
</tr>
<tr>
<td>2.16</td>
<td>Cyclic Permission Model</td>
<td>48</td>
</tr>
<tr>
<td>2.17</td>
<td>Fair Ownership Model</td>
<td>49</td>
</tr>
<tr>
<td>2.18</td>
<td>System architecture of Groups in OpenSim</td>
<td>50</td>
</tr>
<tr>
<td>2.19</td>
<td>Standard Phoenix Viewer (left) and Modified Phoenix Viewer without user options to change appearance or remove clothes (right)</td>
<td>57</td>
</tr>
<tr>
<td>2.20</td>
<td>Completed student projects in SL - Dijkstra’s shunting algorithm</td>
<td>58</td>
</tr>
<tr>
<td>2.21</td>
<td>Completed student projects in OpenSim – Interactive Door Systems</td>
<td>60</td>
</tr>
<tr>
<td>4.1</td>
<td>High level model to analyse MUVE integration for blended learning with MULEs</td>
<td>64</td>
</tr>
<tr>
<td>4.2</td>
<td>The Comparison of the key use cases of the student role in Moodle with available OpenSim/SL use cases</td>
<td>66</td>
</tr>
<tr>
<td>4.3</td>
<td>The comparison of the key use cases of the teacher role in Moodle with the available OpenSim/SL use cases</td>
<td>68</td>
</tr>
<tr>
<td>4.4</td>
<td>The comparison of the available key use cases in Moodle and OpenSim/SL for the academic role module coordinator</td>
<td>70</td>
</tr>
<tr>
<td>4.5</td>
<td>Creating a rich text entry for a Moodle course using the in-world web browser without Sloodle mediation (direct access to Moodle through URL)</td>
<td>73</td>
</tr>
<tr>
<td>4.6</td>
<td>An operational instance of the interactive web based OSPF algorithm simulation learning resource</td>
<td>75</td>
</tr>
</tbody>
</table>
Figure 4.7: A collaborative activity in the Routing Island – the construction of hypothetical Internet core as a hypercube of degree 4 ................................................................. 77
Figure 4.8: Wireless Island overview and a simulation ........................................... 79
Figure 4.9: The map of experiment regions: left - the world map, right - a region ........................................... 80
Figure 4.10: Varying levels of avatar customisations: ............................................. 82
Figure 4.11: Student content creation and content alteration in the learning environment ........................................... 82
Figure 4.12: A region compared at the end of the session: left – original layout and right - the terrain modified and content altered region ........................................... 83
Figure 4.13: The Component Plot in Rotated Space and Rotated Component Matrix ........................................... 85
Figure 4.14: Taxonomy for classifying different management practice scenarios based on management aspects ........................................... 92
Figure 4.15: Process to be followed for developing policy considerations ........................................... 94
Figure 5.1: The distribution of the study participants ............................................. 99
Figure 5.2: OpenSim function framework for policy implementation ........................................... 103
Figure 5.3: An example graph topology written in DOT ........................................... 105
Figure 5.4: The graph representation of Gephi for the topology script in Fig. 5.3 ........................................... 105
Figure 5.5: The complex network of OpenSim functions for educational uses ........................................... 106
Figure 5.6: The selected Avatar Fly ego-network and the legend for users ........................................... 107
Figure 5.7: Hypothesis test output (PASW) ............................................................... 110
Figure 5.8: The map of the designed VHD training environment ........................................... 113
Figure 5.9: Completed VHD training environment ................................................. 114
Figure 5.10: A VHD researcher referring to the guidance document while practicing policy implementations on VHD islands ........................................... 115
Figure 5.11: Dilemma situations presented in Notecards for disaster management decision making ... ........................................... 116
Figure 5.12: Directed learning paths and Sloodle activities for VHD learning scenario 1 ........................................... 117
Figure 5.13: The ego-network of avatar activity function - Terraform ........................................... 118
Figure 5.14: Role-play character avatars – a refugee female (left) and a male (right) ........................................... 119
Figure 5.15: Disaster management official avatars – British Red Cross Aid Worker (left) and Refugee Camp Officer (right) ........................................... 120
Figure 5.16: The ego-network of Avatar Teleport obtained from the tool ........................................... 122
Figure 5.17: An example usage of an allocated land parcel with content and land alteration ........................................... 124
Figure 5.18: A selection of a complex ego-network for land management ........................................... 125
Figure 6.1: Pre-test Prototype Island and a training guide at the arrival point ........................................... 131
Figure 6.2: Design map of Test Island for the pre-test evaluation ........................................... 133
Figure 6.3: Completed Test Island with parcel settings and sensors ........................................... 134
Figure 6.4: Boxplots of walking times to each sensor for the island users ........................................... 135
Figure 6.25: Aerial view of completed Introduction Island .................................................. 157
Figure 6.26: Aerial view of completed Management Island ...................................................... 157
Figure 7.1: Experiment Island ........................................................................................................ 165
Figure 7.2: Land border walls for parcel creation for individual and group projects ................ 165
Figure 7.3: Content objects and empty content holders for evaluation tasks .............................. 166
Figure 7.4: Sloodle setup with user registration and MCQ quizzes on Quiz Chair .................... 166
Figure 7.5: Average performance scores (%) for Subtask 1 ......................................................... 168
Figure 7.6: Average performance scores (%) for Subtask 2 ......................................................... 169
Figure 7.7: Average performance scores (%) for Subtask 3 ......................................................... 170
Figure 7.8: Average performance scores (%) for Subtask 4 ......................................................... 171
Figure 7.9: Average performance scores (%) for Subtask 5 ......................................................... 172
Figure 7.10: Average performance scores (%) for Subtask 6 ....................................................... 172
Figure 7.11: Average performance scores (%) for Subtask 7 ....................................................... 174
Figure 7.12: Average performance scores (%) for Subtask 8 ....................................................... 175
Figure 7.13: Average performance scores (%) for Subtask 9............................................................................. 176
Figure 7.14: Average performance scores (%) for Subtask 10......................................................................... 177
Figure 7.15: Average percentage score for the task scenario................................................................................ 178
Figure 7.16: Number of correct responses for each question.................................................................................... 179
Figure 7.17: Percentage score differences per each MCQ ...................................................................................... 179
Figure 7.18: Total score distribution box plots for both groups.................................................................................. 180
Figure 7.19: SPSS (v19) output of ANOVA for testing hypothesis............................................................................. 181
Figure 7.20: SUS value distributions of the two groups............................................................................................ 183
Figure 7.21: SPSS (v19) output for ANOVA to test hypothesis .................................................................................... 183
Figure 7.22: The distribution of educational value scores of the two groups............................................................... 185
Figure 7.23: SPSS (v19) output for ANOVA to test hypothesis..................................................................................... 186
List of Tables

Table 1.1: Research objectives and the corresponding discussions ............................................ 5
Table 3.1: A Comparison of land units and ownerships in SL and OpenSim .................................. 42
Table 3.2: Roles and task types in content management ............................................................ 44
Table 3.3: Content Ownership and Permission Model of SL and OpenSim ................................... 46
Table 3.4: Content types available for composite permissions in SL and OpenSim ......................... 46
Table 3.5: Composite permission model for objects with scripts ................................................. 47
Table 3.6: Abilities available in Groups and their function categories ........................................ 52
Table 3.7: Modified SUS Questionnaire items and relevant mean and std. dev. ......................... 61
Table 4.1: Sloodle tools and corresponding Moodle functions ................................................... 72
Table 4.2: The summary of experiment groups and module details ............................................ 79
Table 4.3: Questionnaire items .................................................................................................. 81
Table 4.4: Questions on student engagement and the descriptive statistics ............................... 86
Table 4.5: Questions on student self-regulation and descriptive statistics .................................. 87
Table 4.6: Questions on system environment management and descriptive statistics ............... 88
Table 4.7: Summary of the analysis variables and their internal consistency measures ............... 88
Table 4.8: Regression analysis summary .................................................................................... 89
Table 5.1: An example management policy with brief details on possible conflicts .................. 98
Table 5.2: Questionnaire items and descriptive statistics of the responses ............................... 100
Table 5.3: Analysis variable summary with consistency measures ............................................. 101
Table 5.4: A sample of EVC ranks and corresponding PLoS values ........................................... 108
Table 5.5: Descriptive Statistics for the two groups of PLoS ....................................................... 109
Table 5.6: A selected sample of Betweenness Centrality measures of functions ....................... 111
Table 5.7: A set of policies for the management of VHD learning Scenario I .............................. 117
Table 5.8: A set of policies for VHD use scenario 2 management .............................................. 121
Table 5.9: A set of policies for VHD use Scenario 3 management ............................................. 123
Table 6.1: Average time comparison for the four task phases ................................................... 140
Table 6.2: Descriptive statistics of the entire task for the two groups ....................................... 140
Table 6.3: Key observations of each user on the task engagement ........................................... 140
Table 7.1: The Task Scenario given to the experiment participants ........................................... 162
Table 7.2: Sloodle MCQ set with the answers ............................................................................ 163
Table 7.3: Allocated score points for each subtask and its component activities ....................... 164
Table 7.4: The comparison of subtask score means between the groups .................................. 177
Table 7.5: Average response values for the SUS questionnaire .................................................. 182
Table 7.6: Average response values for educational value questionnaire ........................................ 184
Table 7.7: SUS questionnaire response per each participant .............................................................. 190
Table 7.8: educational value responses per participant ................................................................. 190
Table 7.9: SUS and educational value scores for each participant ............................................... 191
1. Introduction

1.1 Research Overview

Immersive 3D Multi User Virtual Environments (MUVEs) designed for education, referred to as 3D Multi User Learning Environments (MULEs), have shown sufficient success to warrant their consideration as a mainstream educational paradigm [1, 2]. With the use of MUVEs for supporting learning and teaching, educational institutions are facing a novel set of management challenges that come alongside the immersive and engaging learning experience. Although MULEs have been extended to various innovative and attractive use cases, their management can be challenging: lecturers can find that the underlying system functionalities and use cases are less cohesive than conventional online facilities, while students can be overwhelmed by the rich and engaging nature of the 3D immersion and might focus more on environment features rather than the Intended Learning Outcomes (ILO). In order to overcome this challenge the research discussed in this thesis has developed a set of user support models and tools for managed learning in MULEs.

Existing online learning environment management practices have been specialised for non 3D environments, such as 2D Web and static learning content, user collaboration and associated learning objectives. MUVEs however are designed to cater for entertainment [3] and commercial needs; hence, the system models used are not explicitly designed to support educational requirements. MUVEs at present are not adequate in themselves as a complete learning environment; there must be additional support for formal course module management and student guidance [4]. However, in order to perform these MUVE management tasks and student support, academics need to be aware of the functions and mechanisms available in them. Since MUVEs have a steep learning curve [5-7], suitable user support strategies on MUVEs can help teachers and students to overcome this challenge. This research investigates problems in this domain using two widely used and closely related MUVEs: Second Life (SL) [8] and Open Simulator (OpenSim) [7]. The research provides its recommendations and contributions with respect to these MUVEs.

1.2 Thesis Statement

The thesis of this research can be stated as follows:

_The provision of user support for enhancing user self-regulation and contextual environment management can improve the management of learning in a Multi User Learning Environment._

Evidence in support of this thesis is shown in:

_Theoretical Underpinning –_ The thesis is supported by educational theory, which points towards the use of blended learning with supporting technological environments. These support the thesis and
subsequent research studies. The research rationales are supported by related studies and standard theories on learning with MUVEs and e-Learning.

**Domain Feasibility** – The system studies were performed on the selected MUVEs, Second Life [8] and Open Simulator [7], and have been incorporated into the design of user support models and tools. The design and implementation of user guidance tools are compatible with the selected MUVEs; these tools are shown to be capable of provisioning effective user support for managed learning.

**Validity and Acceptance** – This research includes the evaluation of the thesis by way of an intervention strategy of prototype implementations of user support tools and the evaluation of those for usability and educational value. The solutions are examined with quantitative analyses. Case studies on working projects and artefacts in the selected MUVEs are also used to validate the thesis statement.

### 1.3 Research Objectives and Methodology

This research aims to provide user guidance for the effective use and management of the selected MUVEs. It therefore seeks to achieve the following objectives:

**Objective 1** - Identify appropriate system models of the selected MUVEs for learning

**Objective 2** - Identify productive use cases for teaching and learner support in these MULEs

**Objective 3** - Identify the most significant management aspects for MULEs

**Objective 4** - Develop user guidance models for MULE management

**Objective 5** - Design and implement prototype user guidance and training tools for the selected MUVEs

**Objective 6** - Evaluate the guidance and tools and make recommendations for learning management in the selected MUVEs

### 1.3.1 System Environment for the Research

Second Life is one of the leading MUVEs available. It is owned by Linden Labs and was first launched in June, 2003 for public use. SL follows the client-server model of system implementation; the servers remain the property of Linden Labs under their exclusive control. SL has a large user community making it one of the most populous MUVEs at time of writing. The University of St Andrews had rented a virtual region (Minerva Island) for its educational and research activities but had moved to OpenSim between 2008 and 2010. OpenSim is a freely available open source MUVE platform. The close similarity between SL and OpenSim is due to the design considerations made by reverse engineering the SL Viewer source code and the SL communication protocols. With an architecture that supports virtual world development and plugin support for additional functions, OpenSim has become a popular choice for academics, either to start their MUVE based teaching or to shift their rent-based educational regions in SL to locally hosted OpenSim installations [9, 10]. For example, the University
of St Andrews has an OpenSim grid that hosts virtual regions for education and research. Further details on SL and OpenSim will be discussed in Chapter 3.

MUVEs in general and OpenSim in particular leave the management of educational environments on these platforms as the sole responsibility of academic institutions [9, 11]. The development model for OpenSim is a generic MUVE platform [7] catering for a plethora of application domains, without domain specific tools or user support. This high degree of platform generalisation offers many positive features such as flexibility, exclusive ownership and autonomous sessions, yet introduces a wide gap between existing mainstream educational practices and MUVE education potential [12], creating significant challenges for managing the learning infrastructure. A possible solution might be to change the core design of OpenSim to support formal educational requirements. However, such an approach would introduce a critical problem of maintaining modified server compatibility with future architectural changes in OpenSim as an on-going open source project. Moreover, server modification to align with institution specific academic needs could result in compatibility issues with other academic MUVEs that are based on the standard OpenSim server; this can make the modified server based academic activities isolated, which may not be an productive way to address the problem. At the same time, SL allows very little opportunity in this prospect because of its proprietary nature.

This research addresses this challenge through a set of user support models and tools. It is intended that the recommendations based on the selected MUVEs can help academics and students explore and exploit these environments for their teaching and learning needs while fulfilling their formal educational requirements.

1.4 Research Contributions

1.4.1 Overview of Research Contributions

The major contributions of this research can be summarised as follows:

- This dissertation highlights the need for managed learning in MULEs and proposes an approach based on a set of user support strategies.
- A review of related MUVE supported education and applicable pedagogical theories has been carried out, summarised and presented.
- Key system and functional attributes of the selected MUVEs (SL and OpenSim) that affect educational use have been identified and reviewed.
- This work has presented a model in which MUVEs play a central role in defining blended MULEs for effective learning. The model is examined for possible integrations with e-Learning through user functions and application scenarios.
- Important areas for managing MULEs were identified through quantitative and qualitative analysis. User support models were developed for managing learning in the selected MUVEs.
• An interactive network topology was developed to demonstrate the complex and interrelated SL/OpenSim functions to helping users in their management tasks. This tool support was validated for its accuracy through statistical analysis. The tool was further evaluated by using a case study of module teaching at a university.

• A training environment catering for basic and advanced SL/OpenSim management needs was implemented utilising the study findings (Fig. 1.1 shows a section of one of the training islands). The training regions were evaluated for usability and educational value. The results were used to inform revisions to the tools and support methods. The completed training environment is now hosted in the university MULE (OpenSim grid).

• Evaluation of the user support models and training tools showed a positive and significant response from participants; the peer-reviewed publications demonstrate an interest from the wider academic community.

![Training Island for 3D MULE](image.png)

**Figure 1.1:** Training support for MULE management – Management Island

### 1.4.2 Selected Publications

A selected set of publications representing the contributions of this research is shown below. A complete list of the research publications associated with this work is in Appendix 1.


The achievement of research objectives and the related discussions for each objective with contributions can be summarised as shown in Table 1.1.

<table>
<thead>
<tr>
<th>Objective</th>
<th>Discussed in Chapter</th>
<th>Related Publications</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2, 3</td>
<td>P1, P6, P9</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>P4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>P3, P5</td>
</tr>
<tr>
<td>4</td>
<td>4, 5</td>
<td>P2, P7</td>
</tr>
<tr>
<td>5</td>
<td>5, 6</td>
<td>P2, P8, P10,</td>
</tr>
<tr>
<td>6</td>
<td>5, 7</td>
<td>P2</td>
</tr>
</tbody>
</table>

Table 1.1: Research objectives and the corresponding discussions
1.5 Thesis Structure

The thesis starts by establishing the value of MUVE supported education and the importance of providing user support for managing these learning environments.

Chapter 2 reviews related work. The chapter discusses MUVEs and educational uses, technology enhanced education, relevant educational theories, management and usability of learning environments and self-regulatory learning practices.

Chapter 3 describes system studies on SL and OpenSim with respect to their educational use while highlighting the challenges that ensue. The chapter initiates the discussion about managing these MUVEs for learning activities.

Chapter 4 discusses the integration of MUVEs with the existing online learning infrastructure. It presents a user study which identifies key management aspects for MUVE supported learning. The findings are used for developing user support models for managing learning using MUVEs.

Chapter 5 presents a user study evaluating the need for user support on MULE management. The chapter then presents the network developed to depict the complex and interrelated nature of the functions of the selected MUVEs. Statistical analyses are performed to validate the accuracy of the network. The final section presents a case study on evaluating this user support tool for managing a MULE used in a course module.

Chapter 6 describes a user study conducted to select a suitable form of user training for MUVEs. The analysis of data and user feedback suggested developing a set of user training islands to help perform introductory and advanced functions in-world. The design considerations for these islands are presented.

Chapter 7 presents the evaluation of user training islands (Introduction Island and Management Island) for managing an OpenSim MULE. It describes the revisions made and relevant further evaluations carried out before deploying the training environments in the university OpenSim grid.

Chapter 8 concludes with a review of the research contributions before presenting potential future research areas linked to the outcomes of this study.

1.6 Summary

This chapter has initiated the discussion on the need for user support for managed learning in MULEs. The goal of this work is to understand how MUVEs can be used in formal educational settings and to explore suitable user support strategies for these new learning environments. The main work contributes to developing a set of user support models and tools to facilitate the management of the selected MUVEs. This chapter gave an overview about the research, the thesis statement, research objectives and the study contributions. Finally, the chapter outlined the thesis structure.
2. Review of Literature

2.1 Introduction

This chapter reviews related literature and previous work. Section 2.2 gives a brief overview about MUVEs including details about some MUVEs for learning. Section 2.3 discusses Technology Enhanced Learning, e-Learning and blended learning; Section 2.4 presents related studies and previous projects on using MUVEs for education; it includes educational projects developed using Second Life and OpenSim at the University of St Andrews. Section 2.5 elaborates a range of educational theories and learning models with respect to the learning and teaching in 3D MUVEs; details about successful applications of theoretical models are also included. Section 2.6 discusses aspects of the management of learning environments. Finally, Section 2.7 presents related work on users behavioural and usability aspects.

2.2 Multi User Virtual Environments

Virtual environments with a 3D space that allow multiple users to immerse and explore are referred to as 3D Multi User Virtual Environments (MUVEs). The term MUVEs can be used to describe any system that allows multi user interactions. However, this work refers to the common understanding of MUVEs that allow users to interact and explore the 3D environment without predefined goals or story plots. Therefore, Massively Multiplayer Online Role Playing Games (MMORPGs) and other types of multi user games are not considered; nevertheless, a brief overview of MMORPGs is provided acknowledging the shared attributes and common ancestry of the development of MUVEs.

2.2.1 Game Environments - MMORPGs

Games have been a popular application type for entertainment throughout the history of computing and the Internet. Although today’s very sophisticated, graphic intensive multiplayer games substantially differ from the early types of casual games, certain core concepts, such as a predefined plot of story to follow, fixed game environment functions, goal achievements, etc. have been preserved throughout. In comparison to the traditional games (casual games) 3D virtual environments that allow multiple users to play different roles of a game (MMORPGs) have been quite popular and share some common attributes with MUVEs.

Many MMORPG systems feature a character progression method in which players earn rewards for their actions and reach higher levels that make them better at their play. Combat, often with Non-player Characters (NPCs) or other players and completing quests are the primary ways to earn rewards (points, weapons and resources) and is part of the game progression. The highly competitive nature of the user interactions and the roles defined in the game plot, MMORPGs are often associated with aggressive and
quest based user actions and cultures. MMORPGs require significantly lower bandwidth in comparison to MUVEs[13]. Moreover the simulation strategies result in a fixed environment with no flexibility for user content creation, which is often noted by academics who have tried using MMORPGs for teaching. World of Warcraft (WoW) [14] is a prominent MMORPG, developed by Blizzard Entertainment. It follows a series of game plots and releases with respect to the Warcraft universe, the simulated game virtual world. WoW has been used for a range of educational activities [15, 16]; however, WoW and MMORPGs in general have been criticised for their lack of flexibility for designing required educational spaces [17, 18].

2.2.2 MUVEs – an Overview

MUVEs (also referred to as 3D virtual worlds or Metaverses) show unique characteristics, which distinguish them from other systems and environments in the widely accepted Reality-Virtuality (RV) Continuum [19]. MUVEs fit into the virtual environment end of RV Continuum, and can be associated with augmented virtuality and augmented reality spaces to define immersive, mixed reality environments for educational purposes as in [11, 20]. MUVEs follow a client-server system architecture. A relatively thick client application [3], known as the viewer or MUVE browser, is used as the client, depicting the virtual reality according to the server responses. MUVEs provide a seamless, persistent virtual environment where users can transparently roam around without predefined goals or story plot [21]. Users interact with the virtual environment through special 3D virtual persona called avatar; avatars can be in the forms of abstract shapes, an animal or mythological creature, or the regular human body shape. The simulated world is generated following a realistic physical law of motion and dynamics through a physics engine. MUVEs allow users to create and develop customised content; users mostly design, develop and modify their content objects to populate their virtual space [21, 22]. To support these user activities MUVEs incorporate a range of utility functions and collaboration tools for community building. MUVEs provide a mechanism to enhance environment and content behaviour through user programming known as scripting (in SL and OpenSim). Second Life and OpenSim were used as the MUVEs and further details are discussed in Chapter 3.

Some MUVEs support virtual economies and commercial transactions within the MUVE as well as between the real world and the MUVE. For this feature, virtual currencies are often used with respect to the MUVE: for example Linden $ in SL[23], and ThereBucks in There[24]. To prevent artificial manipulation of these economies and virtual currency values, MUVEs follow a tight monitoring and controlling process regulating their commercial transactions. These controls sometimes can introduce additional challenges limiting the required control and flexibility for learning (in SL to form groups for group collaboration activities users must pay a fee).

MUVE systems can be categorised into two groups according to the way in which they offer their solutions: first category of MUVEs provides a virtual environment that supports for a larger user community of active participation (e.g. Second Life). The second category of MUVE systems provides a solution platform as a generic toolkit to develop and implement a user defined virtual environment for unique needs (e.g. OpenSim, Open Wonderland). These systems can be either free and open source or proprietary based on a business model. Categorisation of MUVEs according to different parameters has
been researched previously; Messinger and others [18] have proposed a five scale taxonomy on purpose (orientation), place (location of interaction), platform (synchronous or asynchronous activities), population (demographic data) and profit model to group MUVEs; a classification of popular MUVEs according to these factors is presented in [18]. Another factor to consider for classifying MUVEs is the accessibility of the virtual environment, i.e., public access or limited to a specialised group that meets a condition (institution specific). SL and Active Worlds [25] are examples of publicly accessible large community based MUVEs whereas certain private installations of OpenSim or Open Wonderland based MUVEs impose access restrictions.

There are a large number of MUVEs at present and a good snapshot overview on MUVE industry is provided by the quarterly virtual world universe reports of KZero[26]. There are many kinds of virtual worlds but all of them share the following features [27]: Shared space, Avatar, Immediacy of actions (realtime diffusion), Immersion, Persistence and Socialization with creation of social groups. An elaborated and extended view of these MUVE characteristics are given in [28]; it identifies seven central characteristics of MUVEs as follows:

- Realistic: the environment captures essential aspects of the real world which allows navigation and communication to be intuitive.
- Persistence: the environment persists over a long period of time as do changes that are made to that environment.
- Mutability of environment: users are able to create content for and edit spaces within the environment and not simply arrange or rearrange objects provided by the system.
- User control over objectives: There are no predefined set of goals, these can be created by users of the system.
- 3D: The world is 3D providing a realistic environment
- Avatar: the world is experienced through the proxy of an Avatar.
- Distributed: MUVEs are of necessity distributed allowing humans in diverse locations to interact with each other through their avatars.

These MUVE characteristics make them a unique type of application. Moreover, when MUVEs are used for education they offer unique learning affordances and opportunities to overcome challenges associated with other types of learning, which will be discussed later.

The MUVE industry is dynamic and growing; new system features and visually appealing virtual environments regularly emerge. At the same time, some of the MUVEs that perform well can disappear. For example, Teen Second Life [29], There (old version) [24], Google Lively [30] are a few of the MUVEs that could not be sustained. The free and open nature of OpenSim has attracted many innovative approaches from academics and researchers to develop their educational environments in a cost effective manner while accommodating the desired educational theoretical support.

Although a comprehensive analysis of SL and OpenSim is presented in Chapter 3, the following details give a brief overview on why these systems are a good fit for this study. SL has a large user community which includes many with an interest in education and research activities [18, 31-33]. Fig. 2.1 (left image obtained from [34]) shows the growth of SL grid regions to-date. OpenSim on the other hand
provides a platform and a set of tools to implement MUVEs; there are many publically hosted OpenSim grids placing the total region count in the same range as SL.

Fig. 2.1 (right image, obtained from [35]) shows the growth of OpenSim regions in the most popular 40 grids. The cross compatibility of these two environments (SL and OpenSim) enables users to experience a similar environment and features. This allows them to decide which is better for educational hosting based on the required flexibility, technical expertise, cost concerns and the need for accessing a large community. Further to the objectives of selecting these two environments for this study, these data also indicate that the study contributions can help a large user community in these two MUVEs, which is likely to represent a considerable portion of MUVE supported higher education at present.

![Figure 2.1: SL and OpenSim region growth](image)

With respect to the focus of this study apart from SL and OpenSim, a few of the popular MUVEs that are widely used for education are briefly discussed as follows to give an overview about other available alternatives.

**Active Worlds**

Active Worlds [25] is one of the earliest MUVEs, which started in 1995 and made the service publicly available, from 1996. It provides a 3D environment with several separate worlds either owned by Active Worlds or the users. Initial system designs of Active Worlds universe had orientations towards being the 3D equivalent of 2D Web browser, where users can create 3D virtual places instead of web pages.

**Open Wonderland**

Open Wonderland [36], initially known as Project Wonderland, is another extensible MUVE platform maintained by the open source community, the Open Wonderland Foundation. With the initial support by Sun Microsystems the project has been developed using Java and can run on multiple operating systems. A modular architecture with client-server approach allows Open Wonderland developers to extend and package the content and 3D environment through services. Strong security implementations allow fine-level access control on content and the concept of Federation allows users to navigate
between virtual worlds easily. There are a number of successful educational projects for immersive learning with Open Wonderland such as the SIMiLLE project [37].

**RealXtend**

RealXtend [38] is another open source MUVE platform which was developed by the RealXtend Association and the RealXtend Foundation following the OpenSim initiative. RealXtend extends the capabilities of OpenSim incorporating different features that are not in the OpenSim platform. Its unique viewer, Naali, supports OpenSim’s ROBUST architecture [7]; the RealXtend server and viewer applications are closely aligned with the conceptual implementations of the OpenSim architecture.

### 2.2.3 Related Terminology of MUVEs

There are some unique terms specific to MUVEs, in particular to Second Life and OpenSim. This section provides a brief overview about a set of terms that are frequently used for this thesis.

**Avatar** – A virtual persona where the users are represented in-world. It can be of any form such as human, animal or an abstract shape. The appearance of an avatar can be changed according to user’s preference.

**Estate** – The highest level of land entity that groups several regions together to support delegation of land management privileges. An estate has an estate owner and one or more assigned estate managers.

**Inventory** – A persistent virtual folder that keeps references to the different types of content objects belonging to an avatar. Users can take objects or content types from their inventory or save the developed or copied content in the inventory for future use.

**God** – The in-world administrator role to perform admin tasks through the client viewers.

**Island** – A virtual region surrounded by sea.

**Parcel** – The smallest unit of land that a virtual region can be divided into. At the maximum size a parcel is the same size as a region (65,536 m²) and at the smallest size it is 16 m².

**Physics Engine** – is the software component that accounts for simulating the physics. It mainly simulates the gravitational forces and relevant dynamic properties. Physics can either be enabled or disabled at various levels depending on the desired simulation.

**Prim** – the smallest unit available for content object creation. Different virtual worlds provide different shapes; however, every system ensures a set of basic 3D shapes required for construction.

**Region** – an area of land with fixed boundaries. Often, surrounded by sea (i.e. an Island), if not adjacent to other land.

**Rezzing** – the action that creates content in-world or taking a content object from an avatar inventory to in-world.
Sandbox – an openly accessible land area within a virtual region with no (or minimum) restrictions to practice content creation and to get familiar with the virtual environment. Students can be allowed to access it freely to experience the virtual environment before engaging in educational tasks.

Teleporting – an action that instantaneously moves an avatar between two regions or between two places within a region.

2.2.4 Management Challenges in MUVEs

Research on MUVE management challenges often takes the perspective from the original entertainment and commercial objectives. Moreover, a common view towards considering MMORPGs and MUVEs together treating them as 3D virtual worlds can be seen. Although these findings are mainly about managing these environments to facilitate game play and financial stability, but not the educational challenges per se, these can provide a useful insight for this study.

The European Network and Information Security Agency (ENISA) published a report on 3D virtual world security and management challenges [39]. A similar study and congruent list of challenges have been published by [40]. The key challenge areas presented in these studies can be summarised as follows.

Age verification – This is a common challenge for any application that has an age limit to fulfil for its service access. Usually, this is addressed through Terms of Service (ToS) or End User Licence Agreement (EULA), but it is not fool-proof.

Identity Theft, Authentication and Authorisation challenges – These are common areas of concern for users who have accumulated wealth or invested real money in the system. Games tend to report a higher rate of incidents associated with this.

Intellectual Property Theft, Copyright issues – These challenges can be at different levels relating purely virtual content or properties that have a close association with an entity or content in the real world.

Confidentiality and privacy of user actions/data – Eavesdropping on other avatars actions and communications, access to sensitive information relevant to real life, and exposure of such information without the notice of the victim.

Anti-Social Behaviours – This is the broadest category of misbehaviours that an avatar can practice. It includes vandalism, user harassment and stalking, defamation, cybersquatting and Griefing attacks (a harmful action similar to bullying[41]). These actions can affect the user experience to varying levels depending on the management strategies implemented in the environment.

The culture of the environment and the user traits are significant contributors for successful management [39]. In particular, certain game virtual worlds and MMORPGs allow cheating in their culture [42], which is seen as academic misconduct if practiced in education. Most of these management challenges can be addressed by having a known community of users and suitable user support strategies. By having a known group of students with suitable identification policy for avatars we can overcome the challenges of age verification, certain types of identify thefts and some of the anti-
social behaviours. Plagiarism and confidentiality challenges can be addressed through institutional academic policies. In order to transform these generic MUVEs into managed learning environments for education this work has researched and developed appropriate user support strategies and methods which exploit the technical and behavioural aspects of the MUVE technologies.

2.3 Technology Enhanced Learning

Technology Enhanced Learning (TEL) can be described as the support of learning and teaching activities by associating suitable technologies with formal education requirements. TEL aims to improve learning opportunities and student experiences by extending the boundaries of traditional learning in terms of place, time, cost and styles of learning. TEL hopes to provide opportunities which enhance learning practices by opening up multiple forms of learner participation, student centred course activities, efficient and effective learning processes and higher levels of learner collaboration to name a few. TEL enables more convenient ways to construct learning materials and can optimise the efficiency and effectiveness of how information is conveyed to the learner [43]. Moreover, TEL provides opportunities to associate a range of suitable assessment and feedback methods supported through technology, thereby letting the students and teachers communicate clearly about learning objectives and student achievements. TEL can be facilitated in various educational environments that may or may not involve computers [44]. Widely used and critical technologies which are used to enhance education include: web based e-Learning, complex gaming and immersive technologies, mobile learning platforms, interactive videos, electronic blackboards and presentation tools [43]. In fact, although the associated technology plays a significant role by providing the infrastructure for learning, it should be directed by the relevant pedagogical and educational processes to make learning a formal activity. According to Boytchev (et al.,) [44] using sophisticated equipment and/or software in education may incur a high, sometimes unacceptable, burden that has little relevance to educational goals; appropriate user support for the technology in TEL is often essential for users to avoid such burdens or overcome the challenges they experience.

In summary, TEL is widely used in today’s education; Internet based e-Learning is one of the most prominent forms.

2.3.1 E-Learning

E-Learning offers the advantages of TEL by incorporating Information and Communication Technologies (ICT). Several views and ideas can be found explaining what e-Learning is. Welsh (et al.) have indicated e-learning as the use of computer network technology, primarily over an intranet or through the Internet, to deliver information and instruction to individuals [45]. E-Learning practices and learning styles indicate a transformation of individual experiences in learning and knowledge through a process of knowledge construction in active participation in the given activities [46].

Clark and Mayer have elaborated the definition in a generic manner explaining: what can be taught (learning content), how it can be taught (learning processes) and why e-Learning is needed (advantages).
Whatever the focused characteristics and the learning instruments and tools, e-Learning eventually provides an improved learning opportunity for individuals [46]. E-learning practices can be used for either synchronous or asynchronous modes of learning: in the synchronous methods more opportunities are given for group collaboration, interactive participation with others and virtual simulation of a classroom whereas the asynchronous approaches let the students learn at their own pace.

ICT services in e-Learning has provided significant gains in operational efficiencies and cost reductions with respect to the demanding educational needs [48]. E-Learning solutions often provide more sophisticated and seamless module management functions with required pedagogical alignments. E-learning is no longer seen as a purely technical and administrative tool; academics can also get the benefit of this pedagogically sound, learner focused approaches of e-Learning [49]. With the advantages gained from the Internet and Web technologies e-Learning has been able to become a mainstream teaching and learning practice in almost all universities and academic institutions. We can find a large body of literature and related work on case studies, pedagogical analysis, management and successful use of e-Learning, which indicate the wide extent of its use and its acceptance. Moreover the recent technological trends of the Internet and Web such as Web 2.0 and social network platforms have introduced new paths for enhanced and attractive learning offers with shared virtual spaces as part of e-Learning. Furthermore, e-Learning helps to overcome the traditional learning barriers (physical, time and resource) supporting students to participate in collaborative group activities, certain types of simulation for learning, contribute to shared learning spaces and active engagements in module activities.

The high demand and wide acceptance of e-Learning practices in universities and HEIs has resulted in a number of commercial, institutional specific, as well as free and open source e-Learning solutions being developed during the last decade. These solutions provide a generic set of teaching and learner support activities, multiple forms of user participation and collaborative activities, resource and knowledge sharing functions and content management tasks incorporating learning management into the e-Learning solution through a blended approach of certain traditional learning activities. In this study Moodle [50], a globally popular and widely used e-Learning solution, is considered for the proposed blended MULE models discussed in Chapter 4.

2.3.3 Other Approaches

Although the popular e-Learning practices with the Internet support account for a larger portion of today’s TEL, there have been certain practices for supporting student learning through different technologies and devices. Draper and others have successfully used an electronic device based student response tracking method for Multiple Choice Questions (MCQ) session with a large group of students [51]. Several schools in the University of St Andrews also use such devices, called “Clickers” to engage students in interactive lecture sessions. However, these approaches tend to be diminishing with the wide use of e-Learning and immersive technologies with pedagogical underpinning.

Learning using mobile phones, commonly known as m-Learning, has been a debated topic in TEL for its technological offer and its pedagogical support. However, the reasons that caused the origin of the practice have a certain merit; portability and the reachability for students in proximity regions [52], in a
context where internet penetration and access to a computer is a significant challenge. With the increased use and popularity of advanced portable devices such as tablets and smart phones, it can be seen that mobile learning will be incorporated into e-Learning, forming an enhanced e-Learning offer. This stance was taken considering e-Learning with portability for the model development of MULEs in this work; however, the research did not consider other TEL approaches.

No doubt, the portability and ubiquity presented by hand held devices add a significant value to existing learning activities. Many researchers tend to consider it as a sub category of broad e-Learning application domain. Lefrere [53] has examined activity based scenarios for ubiquitous mobile e-Learning. In his findings, he concludes that prevailing resource constraints of mobile infrastructure and the complex nature of existing e-Learning use case scenarios has prevented mobile learning being considered as a broad and independent learning technology, but rather as a rich extension of existing e-Learning practices. Moreover, learning with mobile device support does not replace existing learning practices, but it offers a way to extend the support of learning outside the classroom, to the conversations and interactions of everyday life, as a value addition to e-Learning [54].

2.3.4 Blended Learning

The vision that underpins technology enhanced teaching and learner support methods has a broad conceptual richness beyond the specific application types of teaching such as e-Learning. Importantly, e-learning gives many advantages, but there have been criticisms on using it alone as a mainstream method of education. In fact, this concern was examined by Graf and Kinshuk in their work on e-Learning adaption to standard learning styles [55], which suggested the benefits of a blended approach. The abstract view of blended learning including suitable technologies to fulfil educational requirements allows the use of multiple paths to achieve the same learning objectives. With this opportunity, educationalists and teachers have been utilising a combined approach of e-Learning techniques and established traditional classroom practices.

A definition of the term ‘blended learning’ is not generally agreed upon [56, 57]. Colis and Moonen [58] have emphasised the flexibility gains of blended learning approaches and stated it as a combination of traditional face-to-face and online learning components where online activities are an extension to traditional classroom teaching. Rovai and Jordan [59] have defined blended learning as “a flexible approach to course design that supports the blending of different times and places for learning, offering some of the conveniences of fully online courses without the complete loss of face-to-face contact”. Blended learning is characterized as maximizing the best advantages of face-to-face and online education practices [60]. Blended learning can offer more robust educational experience than the offer one can get from either traditional or complete online learning solution, alone [59]. Davis and Fill believe that blended learning, the combination of face-to-face teaching methods with authentic online learning practices, has the potential to transform student learning experiences and learning outcomes [61].

Because of this difficulty of having a specific definition some researchers have followed a different strategy by promoting the benefits of blended learning as the basis for their work. Littlejohn and Pegler introduced a view with 12 considerations [62], and Sharpe (et al.) [63] have introduced 8 dimensions
that embrace blended learning possibilities. Their view on focusing the key attributes and possibilities of blended learning rather than defining the concept seems more realistic for the purpose of practical application. In the context of improving learning practices, blended learning can be considered as the attempt of overcoming the challenges of e-Learning and traditional learning so that teaching and learning activities meet the stakeholder expectations without introducing difficulties.

Gomez and Duart [64] claim that there is a subtle difference in the way blended learning practices are implemented; well-known blended learning practices allow the distinction to be made between component parts, whereas the other possibility, which they call hybrid learning, “is a cross between the elements from different sources that become fully integrated and inseparable” [64]. With this view, they also see the challenge of compulsory redesign of the courses from top to bottom, i.e., from learning objectives to the teaching and assessment tasks; this specialised hybrid concept may not be universally possible or convenient to practice in the myriad of learning and teaching support arrangements, contrary to general blended learning practice.

Littlejohn and Pegler considers blending of MUVEs for learning as “immersion in online multi-player gaming or multimedia role-playing, using extended, authentic simulations to explore real-life problem solving” [62]. However, with the recent advancements in MUVEs broader use cases and application scenarios have become available for today’s learning needs. As a result, it is fair to consider that MUVEs may go far beyond the narrow scope of role-playing or just exploring for solving real-life problems. MUVE supported learning addresses the core requirements of course delivery goals by covering all major phases of learning; in particular, Kolb’s [65] suggestions for experiential learning [33] through its unique learning affordances [66].

No two blended learning designs are identical; this results in a great complexity of blended learning [67]. This is another challenge that teachers have to face when they design and deliver course modules in blended environments. In spite of the advantages from blended learning, university teachers often find it difficult to adopt new online techniques [61]. Suitable user support strategies can help to overcome these challenges.

### 2.4 Teaching and Learning with MUVEs

The unique characteristics of MUVEs that we have discussed previously make MUVEs an important candidate for enhancing education. A large number of successful uses and related research studies on using MUVEs for educational purposes can be found in the literature. A good overview of educational applications of early MUVEs (pre SL era) with several example MUVEs is presented by Dieterle and Clark [68]. Mikropoulos and Natsis have provided a good source of reference by reviewing the research on educational virtual environments (EVE) in the period of 1999-2009, which refers to a number of MUVEs used in education [69]. Hew and Cheung indicated that by March 2008 there have been 470 research articles published on the use of MUVEs in education [70]. The findings of these sources reinforce the views of [2, 71] [1, 18, 72] expecting MUVEs to play a central and essential role in future education.
MUVEs are highly interactive and provide dynamic feedback, learner experimentation, real-time personalised task selection and exploration [73]. Beard et al., [74] indicate that MUVEs offer unique didactic experiences to users through educating them on various topics of importance. All of these have a strong relevance to this research. Jarmon et al. have used SL for their study of experiential learning [33]. Moreover, appropriate pedagogical studies and student learning approaches have been discussed in [75-77], [33, 78]. More details are discussed later.

Immersive approaches can prompt users to employ well learned interaction patterns from the physical world in the virtual world, in contrast to e-learning [44]. Limited features available in webpages may not provide intuitive user experiences for field work training; compared to 2D web interfaces MUVEs provide more engaging simulated environment with user immersion for learning [12]. Using MUVEs for training is advantageous as it decreases the training budget, giving the flexibility for training schedule, and improves trainees’ motivation [79]. It provides richer interactions whereby face to face communication is replicated more closely than in other mediums and users are allowed to replicate body language and gestures [80]. 3D virtual worlds are suitable environments for conducting Role-plays as avatars can demonstrate various types of human characteristics, including communication expressions and gestures [81]. In relation to conventional forms of education in the ‘real world’, virtual worlds allow participants to experience roles and to do things safely, which can be difficult or impossible to do in the physical world [82-84].

Virtual worlds facilitate engaged learning through their dynamic and collaborative environment features [85]. There are unique advantages of using MUVEs for teaching and learner support, which we may not achieve from the other methods, as described above. Dalgarno et al. believe that interactive MUVEs 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 [86]. A number of similar educational uses can be found from literature and related projects such as [78, 87, 88]. Incorporating these unique benefits of MUVEs for learning, Dalgarno and Lee, have defined the learning affordances of 3D virtual learning environments (3D VLEs) in a framework with five affordances [66], as:

- **Affordance 1**: 3D VLEs can be used to facilitate learning tasks that lead to the development of enhanced spatial knowledge representation of the explored domain.
- **Affordance 2**: 3D VLEs can be used to facilitate experiential learning tasks that would be impractical or impossible to undertake in the real world.
- **Affordance 3**: 3D VLEs can be used to facilitate learning tasks that lead to increased intrinsic motivation and engagement.
- **Affordance 4**: 3D VLEs can be used to facilitate learning tasks that lead to improved transfer of knowledge and skills to real situations through contextualisation of learning.
- **Affordance 5**: 3D VLEs can be used to facilitate tasks that lead to richer and/or more effective collaborative learning than is possible with 2D alternatives.

Most of the literature on MUVE learning discussed above is covered by one or more of these affordances. It is important to consider these affordances since they give a good overview about
effective uses of MUVE for learning that gives unique advantages. We will discuss these affordances further when MUVE learning and user support methods are analysed later in this thesis.

With respect to the managed learning with MUVEs: De Freitas and Veletsianos [72] present their view that MUVEs for education are a new interest but most of the studies so far lack discussion on the serious user support required for making MUVE based education a sustainable mainstream practice. They highlight the importance of identifying and filling the gaps in the research. In future, a higher education institution without a presence in virtual worlds will be like an institution without a web presence today; however, the academics will be faced with the challenge of selecting, designing and managing their educational environments [71].

Hendaoui (at el.) [22] have expressed a set of research issues and challenges with respect to education with MUVEs; authors believe that there should be differentiated teaching and assessing approaches considering the learning environment, technology, avatar (user) behaviours, and management of MUVEs. Moreover, the design of learning content and tasks as well as the strategies followed in teaching and learning in MUVEs influence student learning and engagement, which require further research [89]. Some students may find the MUVE so engaging that they get distracted from their module objectives and given learning tasks; at the extreme this may result in lack of participation for the tasks or inappropriate behaviour [17]. Academics must be aware that the students need to become well acquainted with the virtual world environment before they start learning the course content [4]. These views strengthen the importance of this research to provide user support for managed learning in MUVEs.

Second Life and OpenSim are widely used for these educational purposes and research. Petrakou [4] has examined MUVEs as context for online education using Second Life. His findings have included research suggestions such as user training, designing learning activities and their management and competencies of the academics to plan suitable student interactivities. Second Life was the prominent choice for developing educational spaces; however, many studies reported the challenges faced due to Linden Labs exclusive control and the costs associated with SL. With the growth and stability of improved releases of OpenSim, a recent trend of selecting it over SL can be seen in educational projects; NASA moved from Second Life to OpenSim as the choice of virtual world platform [10]. Moreover, the open learning in 3D can be facilitated by using MUVEs such as OpenSim [9]. The University of St Andrews currently hosts an OpenSim based educational grid with a number of unique projects scaling up its MUVE learning activities from a single rented region of SL at the beginning.

In line with the University of St Andrews MUVE based education and research orientations, this study used OpenSim as its main platform of interest and for the training environment development. Nevertheless, the close association of SL and OpenSim makes the findings and contributions are cross compatible and equally applicable to both systems. Depending on the available resources, expertise and the required level of usage, academics may choose either SL or OpenSim to best fit their needs. The key technical and management details about these two systems are discussed in Chapter 3. For an initial comparison Allison et al. [90] provide a complete view on the relevant important factors for educational use of SL and OpenSim includes cost, content and environment development, accessibility and network challenges, learning and assessment tasks and avatar management with respect to teaching and learning [90].
2.4.1 Related Educational Projects

A number of successful educational projects with OpenSim have been developed and used in the University of St Andrews. These projects have provided invaluable educational support for various modules (undergraduate and postgraduate courses) and research at several schools of the University. This research also performed several evaluations and case studies with some of these projects. Further details are discussed in relevant chapters.

**LAVA Project** – The Laconia Acropolis Virtual Archaeology (LAVA) project [91] is a co-operative archaeological learning environment developed first in Second Life and then moved to OpenSim (Fig. 2.2). The project aimed at bringing archaeology fieldwork activities into the virtual environment and let the students practice virtual excavations as part of their module learning.

![Figure 2.2: LAVA project in OpenSim](image)

**Wireless Island** – Wireless Island [87] is a research project developed for teaching wireless communication to Computer Science undergraduate and postgraduate students. It is hosted in OpenSim and provides exploratory learning and research opportunities to the students [92]. Several studies were conducted in Wireless Island for this research and it provided a valuable platform for the findings of this research [93]. More details about the island will be discussed later.

**Routing Island** – This is a collection of educational regions in OpenSim dedicated for teaching network routing. Routing Island [94] provides facility to develop and explore network topologies in an intuitive manner with a range of functions simulating the network traffic. Moreover, it provides valuable educational support for learning different routing algorithms associating with the user created dynamic topologies. It uses Mini Region Module (MRM) feature with C# backend. This tool was used for research evaluation and comparison of proposed models [12]. Further details are presented in Chapter 4.

**VHD Project** – The Virtual Humanitarian Disaster (VHD) Project [95] is a joint work between the schools of Computer Science and Management for supporting teaching and learning about humanitarian disaster management. It has been used in related research as well. This research utilised VHD Project as a main case study environments for evaluating one of its user training tools [96]. More details about the project and the case study are discussed in Chapter 6.
St. Andrews Cathedral – The project has made a virtual reconstruction of the St. Andrews cathedral [97] from 1318 and used it as an educational tool for University and public use. The virtual construction of the cathedral spans across several regions and consists of a large number of prims. Fig. 2.3 shows an aerial image of the cathedral (left) and the internal view of the choir (right).

![Figure 2.3: The virtual reconstruction of the St. Andrews Cathedral in OpenSim](image)

Other Educational Projects – Several other educational projects have been developed in OpenSim such as Linlithgow Palace virtual construction [11], ViStA project for virtual construction of St. Andrews, and teaching Human Computer Interaction (HCI) through student experiential developments [88, 98]. There have been few other projects and research studies done on using OpenSim (and SL) with different objectives: OpenSim/SL traffic management [13], using OpenSim in cloud infrastructure [99] and cross reality user experiences [11] with OpenSim and real-world. Although, these projects are not directly associated with a particular educational goal, they indeed have provided valuable contributions for the development of educational projects in the University of St Andrews.

2.5 Educational Theories for MUVE based Learning

Pedagogical models and theories provide a necessary theoretical underpinning for formal educational requirements. There are a number of related theories and models that have been used for MULEs. These are essential for this research as the learning activities in MULEs often adhere to these theories. Therefore, the user support models and tools for learning management should cohesively support the norms and practices of relevant pedagogies. A particular interest in identifying suitable learning theories and pedagogical models including example applications with MUVEs has been made to support the thesis of the research.

2.5.1 Generic Learning Taxonomies and Models

This section includes some of the widely used and relevant taxonomies and models for learning. These, and most of the other learning pedagogies that are discussed here, were developed before MUVEs became widely used in education; however, they are generic and have been presented in abstract forms,
letting them be applied to modern TEL practices and incorporate their advantages with the learning activities.

**Bloom’s Taxonomy**

This is a classification of learning objectives to be achieved in the learning process. It was proposed by Bloom and considers three major domains of learning: affective, cognitive, and psychomotor [100]. Each of these major domains is defined with hierarchical levels ranging from lowest order processes to the highest. At the lowest level the student can only recall and re-quote the acquired knowledge with no understanding. At the higher levels, students can expand and apply their understandings to new problems. In brief:

- **The Affective domain** includes the skills that define the way people react emotionally and their ability to feel others. Affective objectives target the awareness and growth in attitudes, emotion, and feelings. It has five levels: Receiving, Responding, Valuing, Organising, and Characterizing.

- **The Cognitive domain** is the central aspect of Bloom’s taxonomy. It covers knowledge acquisition, comprehension and expansion giving the student a complete understanding of the problem domain. It has six levels, from lowest to the highest: Knowledge, Comprehension, Application, Analysis, Synthesis, and Evaluation (Fig 2.4).

- **The Psychomotor domain** describes the ability to manipulate a tool or instrument physically. Bloom’s original work did not include detailed levels within this domain; various suggestions and extensions have been proposed to the original taxonomy to cover this domain.

Later, Anderson proposed a modified version [101] following the critique on the strict hierarchical nature of the original taxonomy. Applying this taxonomy to design a game based 3D immersive learning environment is explained in [102].

![Bloom's Model - Cognitive Domain](image)

**Figure 2.4: Bloom's Model - Cognitive Domain**

**SOLO Taxonomy**

The Structure of the Observed Learning Outcome (SOLO) is an education taxonomy proposed by Biggs and Collis [103]. It describes a five level taxonomy to analyse the student capabilities in providing answers to a given task or question topic. The five levels can be briefly described as follows:
• Pre-structural – The lowest level of student learning: student has failed to provide meaningful answers. Shows misunderstanding of the topic.

• Uni-structural – A basic form of student answer showing superficial understanding of the learning topic. Often, it addresses only one aspect of the question.

• Multi-structural – The response addresses several relevant aspects in a correct manner, but the student fails to develop a coherent and consistent answer. The answers are often treated as independent correct answers. Usually subject to quantitative assessment.

• Relational – A response in this category shows the ability to put different aspects together and develop a coherent and accurate answer for the question. However, the skills are still limited to the familiar domain of the question.

• Extended Abstract – The highest level of achievement of the learning outcomes. In this level the student has an in-depth understanding of the subject area and is capable of extending the knowledge into a generalised form for applying to unfamiliar domains and new settings.

Lister et al., have successfully used the taxonomy for designing effective teaching and learning tasks to educate novice programmers [104]. In [105] the researchers developed a 3D virtual environment and examined the student learning achievements according to the SOLO Taxonomy. The results of the students’ interaction with their tool showed a shift from lower to higher SOLO levels, indicating an improvement in students’ understanding, while showing a greater benefit for students classified at the pre-structural and uni-structural knowledge level [105].

Learning Affordances

Affordance theory was introduced by Gibson (1979) referring to the functional properties that determine the possible uses of an object or the environment [66]. In the educational context learning affordances generally indicates the functional relationship between the learner and the learning aids. Conole and Dyke [106] suggest the importance of exploring the ways in which the ICT affordances can be utilised for teaching and learner support. Studies with the specific interest in e-Learning affordances as part of ICT based learning affordances can be seen.

As discussed previously, Dalgarno and Lee defined MUVE learning affordances in a five scale framework [66] summarising the educational benefits. Based on this framework in [66] an improved and empirically supported model for MUVE learning affordances was developed by Gamage et al. [107]. The model shown in Fig. 2.5 summarises the important learning theories and models applicable for MUVE learning. The authors have used their own terms, authentic 3D experiences and artificial 3D experience to represent, simulations representing the real-world and simulations on hypothetical constructions, respectively.
2.5.2 Experiential Learning

Kolb presented the concept of experiential learning in his Experiential Learning Theory (ELT)[65]. Kolb views learning as a process and designed his model to show the stages of the process. The model is based on four stages of learning: Concrete Experiences (CE), Reflective Observation (RO), Abstract Conceptualisation (AC), and Active Experimentation (AE). These four stages are connected in cyclic manner as suggested in Kolb’s model, shown in Fig. 2.6. It is important to mention that a learner can start from any stage by entering the cycle, but has to follow the order of the stages and the direction [110]. Kolb’s ELT model has been used in many successful studies of learning in MUVEs. MUVEs provide an opportunity to use simulation in a safe environment to enhance experiential learning, and learn through new ideas, skills, and mistakes [74]. Jarmon et al., [33] have examined the application of Kolb’s experiential learning theory in MUVEs (Second Life). The authors strongly recommend to use MUVEs for experiential learning practices while suggesting a revision of practices and learning management approaches to deliver the best educational offer from virtual worlds with experiential learning [33]. Stieglitz and Lattemann on using experiential learning in Second Life suggested the need for particular designs and management of learning content and activities to make the learning successful; they claim both the students and teachers are responsible for such special arrangements and learning activities [111].
Figure 2.6: The Kolb’s model of experiential learning

Not only has the Kolb’s model shown positive results with learning in MULEs, but also the structural merit of it for managing MULEs was observed. This training and related practices also follow the experiential learning process as the lecturers have to plan, experiment, experience, and reflect on their management practices depending on the usage context from module to module. This model was considered for developing user support models for MUVE management process to be followed; further information of the use of this model is discussed later.

2.5.3 Exploratory Learning

Exploratory learning theory suggests the provision of learning activities and materials that allow students to explore further into the subject matter without a strict guided process to follow. The minimum use of didactic teaching methods gives the students an open learning path that they can explore and understand the related concepts and knowledge. In exploratory forms of learning, although there can be supportive forms of student guidance and certain levels of constraints to be followed, students often get the opportunity to discover unexpected lessons \([92]\) and follow unique paths to reach the expected outcomes of the learning. Rieman has suggested that the task-oriented learning explorations are the best form to use in exploratory learning since they provide intrinsic constraints making learners focus on the learning objectives, as non-task oriented explorations may not be effective for orienting the learners towards the learning outcomes \([112]\). This is quite important to learning in 3D virtual environments since the rich and attractive features in the environment can be exploited for entertainment but not necessarily for the learning.

De Freitas and Oliver have studied exploratory learning practice in the context of immersive environments and indicated that there should be suitable evaluation methods \([75]\). They have proposed a framework to be followed for this purpose. Later, extending previous work, they presented the Exploratory Learning Model (ELM) specific to immersive education; in fact, as they have mentioned, it is a combination of the experiential learning ideas of Kolb with exploratory learning and put into the
context of MUVEs [76]. Exploratory learning theory has been successfully used for teaching wireless networks in Second Life and OpenSim [87, 92].

2.5.4 Constructivist Learning

The constructivist approach to learning proposed by Dewey, Piaget and others suggests that the students develop knowledge through their experience with the learning environment supported by their existing knowledge. The learners tend to use their existing knowledge and explore the extended knowledge with the support of effective input from the teachers. Teachers shape the learner experience through using better environments to promote positive learning experiences.

The work of Vygotsky on this model of learning introduced the idea of Zone of Proximal Development (ZPD) [113]. The idea behind this specific constructivist approach is that the students are not necessarily being taught by didactical methods from teacher to the student, but letting them to develop and extend their knowledge in the proximal areas beyond their existing knowledge space. Fig. 2.7 demonstrates the ZPD model. Students expand their knowledge towards the unfamiliar territory of learning through the ZPD area. Unknown territory essentially requires teacher support although students can reach ZPD with minimal guidance.

![Zone of Proximal Development model of learning](image)

Figure 2.7: Zone of Proximal Development model of learning

Instead of assessing what an individual learner can do unaided, Vygotsky proposed assessing what an individual was capable of with the help of a teacher [114]. This type of help provided by teachers in Vygotsky’s work was further elaborated and defined as *scaffolding* by Wood et al. [115]. There are several studies that have applied these theories following the constructivist learning approach in 3D immersive environments [116, 117]. Related literature notes that constructivist principles are fundamental and underlie learning in 3D virtual environments [118].
The constructivist learning theory allows learners to construct meaningful knowledge actively from individual experiences. It focuses on the learner’s control of learning processes and attempts to mitigate the knowledge gap [118]. An important characteristic of this theory is the student centred learning environment designed to scaffold their learning. Constructivists emphasize the design of learning environments rather than instructional sequences [119]. This view is relevant for this study since it also suggests user support for suitable design and development of learning activities for managed learning in a MULE.

2.5.5 Collaborative and Corporative Learning

The model of collaborative and corporative learning positions the learning task on students by giving them a feeling that they are part of a group. According to Janssen et al., in collaborative learning students actively contribute to the attainment of a mutual learning goal and try to share the effort to reach this goal [120]; cooperative learning allows students to pursue a shared goal through a clear and structured division of labour with the group but is assessed individually [120]. In both of these learning approaches students learn to share and learn through sharing [121]. In practice, the most distinctive feature between the two methods would be that group members are given the same mark for their task completion in collaborative learning, while individual students are given different marks for their individual contributions to the group work in cooperative learning. These two approaches were considered together for this work; it was observed that MUVEs support these approaches well.

The broader category of collaborative learning that fits with the research area of this study is Computer-Supported Collaborative Learning (CSCL). This includes all forms of collaborative learning methods that are facilitated by TEL. According to Wang [122] there are four major features for successful CSCL: individual accountability, positive interdependence, coordination and monitoring. CSCL through MUVEs support all these four features to a high degree. Following Wang’s conceptual framework for CSCL, Andreas et al. [123] have successfully designed and developed a collaborative learning environment in SL. They conclude that MUVEs provide rich environment for CSCL.

An interesting immersive educational project with the collaborative learning model is InterReality Portal, a holistic immersive mixed reality learning environment [124]. It has been designed to offer collaborative laboratory activities for geographically dispersed students with a mixed reality environment combining virtual and real components. Often, the collaborative learning practices benefit by having groups of students with similar socio-cultural traits. However, MUVEs can even support more challenging setups of collaborative learning with virtual student teams having a range of socio-cultural differences [125]. The study encompasses the collaborative efforts of students for problem solving and makes suggestions for region design.
2.5.6 Constructive Alignment

Constructive Alignment proposed by Biggs [108] shows how to infuse the ideas of constructivist learning models with the curriculum design. As the name suggests, the constructive aspect refers to the facility of using a student-centred process of learning and assessment allowing students to construct knowledge and meaning by following learning activities. The term alignment suggests the correct infusion of constructivist ideas from the learning environment design to the assessment of the students learning as the outcome. It has three major phases of activities in the form of formal education as follows.

1. **Intended Learning Outcomes (ILOs)** – these are the intentions of the course module expressed as learning outcomes. Usually, these outcomes are the objectives to be achieved by the end of the learning activity.

2. **Teaching and Learning Activities (TLAs)** – these are the activities designed and practiced for teaching and learning purposes. This phase is crucial for students and teachers since both interact with each other and with the environment through active participation in these activities.

3. **Assessment Tasks (ATs)** – these are the tasks that define the evaluation of learning. Often these are used to examine the students’ achievements with respect to the learning objectives.

Usually, these three major phases are intrinsic features in formal education. The significance of Biggs’ idea is that these should be put together as a seamless process linking each other with constructivist practices as the basis. Therefore, ILOs are defined, TLAs are designed and practiced according to the ILOs, and ATs are conducted to suit the ILOs and TLAs. Supportive pedagogical models can be associated while maintaining the alignment of the three phases.

The importance of this model was experienced with MUVE education. We see constructive alignment as a convenient method to practice the study recommendations accurately. Such practice can harness the benefits of the managed learning in MULEs with the constructive and collaborative learning practices together, while allowing academics to avoid conflicts and redundancies in their educational offers. Further details on using this model will be discussed later.

2.5.7 3P Model

The 3P model proposed by Biggs in [109] relates to Biggs’ idea of constructive alignment of teaching, learning and assessment. The model, as shown in Fig. 2.8, has been found useful in designing and structuring dynamic educational activities and student experiences [126].
According to the model, Presage is the stage that includes all the activities and conditions prior to a given learning activity. The student factors in this stage include the certain behavioural traits and knowledge levels that support for the considered learning activity, and the teaching context represents the academic environment. This can be taken as the MUVE based learning platform and the user support tools for the context of this study. The second stage consists of the teaching and learning activities, which is indicated as Process in the model. Biggs has used the concept of approaches to learning (deep vs. surface) to describe how students engage in the learning process. The Process stage is important for this research and will be further described in Chapter 4 with respect to the management of MULEs. The final stage is the outcome of the learning, i.e., Product. This generally represents the learning outcomes achieved by the students through the previous stage. Often, Product is quantified by a grade or feedback.

2.6 Management of Learning Environments

Managing a formal learning environment is a challenging task. Various approaches have been used through system architectures and usage patterns. This section discusses the related literature on different approaches that have been used for managing learning environments. First it provides an overview of the security considerations on technical and management aspects for teaching and supporting learning with trust. Then a brief review of literature on learning environment management is discussed.

2.6.1 Security and Trust in Learning Environments

Managing computer security is a broad area of interest with many sub disciplines. In today’s educational environment context security is achieved through technical aspects and managerial aspects. Technical aspects of the security of learning environments are undoubtedly essential and expected to be
associated with the design and implementation of the educational platform. On the other hand, the management aspects of security in an educational environment, in particular, focus on how users (students and academics) are provided with a trustful environment for their formal education and how to ensure the expected secure learning by the support of technical implementations of the security. In this study management aspects of security are focused and will be considered as an integrated part of the user support models and tools for OpenSim (SL) based learning management.

Students and academics always expect their educational activities and the learning environment to be trustful by having sufficient security implementations and appropriate management. To make the users trust the learning environment, it should implement confidentiality, integrity, and availability as the fundamental aspects from the technical point of view [127]. Technical implementations of these goals can be in the form of authentication, authorisation, access control and permissions, data encryption and attack prevention. These implementation strategies have been widely researched and well accepted in the present software and system development. Understanding on how different human, organizational and technological elements interplay could explain how different factors lead to security vulnerabilities within organizations [128]. Therefore, when systems are put into use there should be strong management practices to avoid security vulnerabilities.

Previous studies indicate the efforts on finding a generic security and management model for learning environments. Importantly, security and management requirements differ in learning parameters such as technological differences, organization objectives and practices, different learner focus, and content management approaches [129]. While acknowledging the flexibility required for defining institutional specific security and management policies, this idea is further extended by letting the autonomy and flexibility into the module level for learning in MUVEs. In such practice, it is essential that the individual users are strongly aware of the suitable practices in their MUVEs. Rezgui and Marks have indicated that the universities and academic institutions are among the least IS secured environments with poor level of security awareness, by citing a survey carried out by EDUCAUSE [130]. This supports the case for user training based on management considerations for MULEs as a goal of this research.

Few researchers have studied virtual worlds in general to identify possible security challenges. However, their focus was in particular to the generic entertainment and commercial uses [39]. Security management of academic misconduct, identity masquerading, performance evaluation, appropriate learning activities and their management are more important for learning in MUVEs. Lee and Warren [40] have provided an extensive overview of security issues they have identified in Second Life. The commercial motives of SL and its economy have been the centre of their analysis and the problem space they have defined. Although the findings are generic, a need of careful domain specialisation was realised when considering findings for educational environments. Focusing on virtual land use and privacy of avatars, Vilela et al., [131] have presented their work on identifying challenges for a virtual classroom in SL and OpenSim. They have mentioned certain solution practices through LSL programming as well. However, their focus on controlling the environment is not specific to educational requirements; most of the forced controls can hinder learner motivation and the advantages of the environment. It lacks the required pedagogical aspects of the environment control yet it provides valuable insights to this work.
2.6.2 Management Approaches in e-Learning

The management approaches used in today’s TEL in general and e-Learning in particular closely associate the education institution’s management goals with established teaching and supporting learning strategies. Consequently, a strong mutual relationship between the e-Learning processes and traditional practices can be seen; moreover, it has shaped the e-Learning system architecture and functionality designs to bring these systems into a common platform of management practices between the two educational approaches. As a result, a visible separation of the management approaches including institutional administration requirements and learning activity and course content management needs can be observed in most of the widely used e-Learning systems. Learning Management System (LMS) and Course or Content Management System (CMS) are the two major system types for these management approaches. University of St Andrews has its Module Management System (MMS) [132] for LMS specific functionalities and Moodle as the CMS of the university e-Learning infrastructure.

LMS usually provides the management functionalities specific to the business process of the learning activities such as student registrations, module enrolments, formal assessments etc. while CMS often provide specialised functions for educational content delivery, learning activities and student participation, informal feedback etc.[48]. The management of an e-Learning environment can be achieved through these two systems. Cukusic et al., in their study on e-Learning process management within a group of European educational institutes, have concluded that irrespective of the courses being taught or the technologies used in e-Learning systems, the success of e-Learning is determined by the systematic management of the learning tasks and content [133].

Weippl [134] has studied the security and management of generic e-Learning environments and presented a set of guidelines to be followed. In his work he has provided separate views and activities for key roles associated with the e-Learning management process. These include: academics, course managers, content authors and students. The suggestions are mainly based on management approaches of a technical (system administration) nature and user management through management policies. Policy based management is a common approach in universities and HEIs, such as in [135-137]; often the user management aspects are achieved through policies.

Although there are commonly accepted policies for e-Learning and associated learning activity management, quite often, different institutions tend to customise their policies to fit into their institutional and educational practices. The components of a management policy that a university needs may differ as well as the funding to implement [138]. This is an important view, which presents the intrinsic complexities in implementing policies for managing a university learning environment. Further details on user support for policy based management in OpenSim (SL) MULEs are discussed later.

An important factor that features in the management of learning environments is user awareness of the accepted practices and the support for them to refrain from making mistakes. Some studies have examined the need of a managed environment helping students to explore their learning. Citing the work of Carroll & Mazur, Reiman indicates that the novices attempting to explore a totally new system often made major and unrecoverable errors, even with the aid of basic manuals and tutorials [112].
On the other hand, even the academics and management staff of the learning environment may find it challenging to implement policies correctly to achieve the expected managed states due to the lack of their knowledge, as highlighted in [139]. This is a very important fact that suggests beyond the user manuals and help-notes there should be appropriate user support for management practices to prevent critical errors when new users (students and staff) experience OpenSim/SL for teaching and learning.

2.6.3 Summary

There are several research studies for managing e-Learning environments, yet a complete research on managing OpenSim (SL) supported learning environment for formal educational needs was not reported. An important further research on 3D virtual environments is establishing guidelines and best practices to identify and derive rules and principles that will guide and inform the design and development of 3D virtual environments for learning and associated tasks [66]. This suggestion indicates the need of user guidance for MUVE based education and in particular for managing OpenSim/SL supported educational activities which this research focuses on.

2.7 Behavioural and System Usability Aspects for Learning

For the success of any information system, it has to be usable and accepted by its users. Learning solutions based on virtual environments are also governed by this norm. This section includes related literature on system usability aspects, and user behaviours with specific considerations for the thesis of this work.

2.7.1 Learning Environment Usability

Usability of a software system is a well-accepted norm, which may even be considered as the most important attribute for certain application types. Nielsen developed usability heuristics [140] to define the main aspects of usability evaluation. There have been studies applying these generic usability concepts to develop educational environments expecting to achieve high usability. However, Jones et al., [141] argue that by directly following usability features without considering the educational and learning context can introduce challenges for teaching and learning. For example, highly usable user interface designed for a specific domain can still be used for educational purposes, yet can become overly complex when a particular learning and assessment session is designed within. This is particularly true and was evident in many studies of 3D game based learning; the rigid interface options and game plots introduced additional challenges unnecessarily when used for learning. The usability evaluation of an educational environment or tool should incorporate three measures with respect to the learning activity and the users involved: i.e., context, interactions and attitudes and outcomes [141, 142]. Suggestions for usability on teaching with 3D games in [142] indicate that the classification of usability characteristics has to take place in relation to the 3D virtual environment; complexities can be found with respect to UI, the available operations, and the strategies of the game.
Three factors were evaluated and empirically confirmed in [142]: User Interface Acquaintance that indicates the student’s knowhow of the user interface, Navigational Effort that indicates how well the user could navigate through the virtual world, and Environment Distraction that indicates disturbance to learning taking away the student attention disengaging them from the educational task are the main factors of usability for learning with 3D virtual environments. These suggestions were also observed in our teaching and learning support projects in OpenSim (SL) as part of this research. The view that usability aspects of an educational environment in MUVEs should include the convenient facility to engage in learning activities in-world and the ability to manage the environment to get the maximum benefits for the learning has been followed. User training for managing MUVE learning include the above three factors and substantially help to increase the usability of MULEs.

One of the widely used generic and efficient usability evaluation measures is System Usability Scale (SUS) proposed by Brooke [143]. It consists of 10 questions: alternatively arranged 5 supporting (odd numbered questions with positive wordings) and 5 opposing statements (even numbered questions with negative wordings) aimed at the examined system usability. It evaluates the responses through 5-point Likert scale from Strongly Disagree (1) to Strongly Agree (5). For positively-worded items (1, 3, 5, 7 and 9), the score contribution is the scale position minus 1; i.e., {0, 1, 2, 3, 4}. For negatively-worded items (2, 4, 6, 8 and 10), it is 5 minus the scale position {4, 3, 2, 1, 0}. To get the overall SUS score, first add all the score values and then multiply the sum by 2.5. SUS scores range from 0 to 100. The statements are:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

SUS based usability analysis has been confirmed for its validity and sensitivity [144, 145]. Also SUS has found to be reliable even at low sample size of 12 participants, while providing rapid convergence to the accurate conclusion [144]. On a comparative study with 5 widely used usability measuring questionnaires, SUS yielded among the most reliable results across sample sizes [146]. Because of these advantages, SUS has been widely used in e-learning and other forms of TEL to examine the usability of those learning environments [147-149]. The flexibility and the generic nature of the SUS statements that could be well fitted into most of the learning contexts have been commended in these studies. SUS
questionnaire has been used in MUVE supported educational projects as well. It was used for several educational islands developed in Second Life and OpenSim providing sufficient measures to evaluate the usability of those virtual regions through user studies [88, 92].

In overview, one of the major challenges of learning and teaching in existing MUVEs (OpenSim and SL in particular) is the usability of the environments; although the environment is attractive and engaging, the interface and the complex tasks associated with avatar and environment interactions can impede the motivation of students for learning as well as the academics for using those in their teaching, as highlighted by several previous studies. Moreover, the domain specific terminology used in viewer applications and the complex UI menu options further increase the difficulties for users. This is a significant challenge affecting the perceived usability of a MULE. The work presented in this thesis is pivotal in enhancing the usability of these environments for managed learning. Further details on usability evaluations are discussed in later chapters.

2.7.2 Self-regulatory Learning

Self-regulation is a widely discussed behavioural trait, which means individuals behave with self-control of their actions according to the context and environmental factors of their work. This concept of students behaving with a restrained approach imposed at their desire towards achieving learning objectives has been developed as a learning paradigm and widely researched in traditional and TEL environments.

Zimmerman and Schunk described self-regulated learning as a collection of self-generated thoughts, feelings, and actions, which are systematically oriented toward attainment of student goals [150]. Pintrich, defines self-regulation as “an active, constructive process whereby learners set goals for their learning and then attempt to monitor, regulate, and control their cognition, motivation, and behaviour, guided and constrained by their goals and the learning environment contextual features” [151]. This is important when promoting self-regulatory learning in virtual environments. The considerations for environment context and its attributes are quite significant for engaging and supportive learning in MUVEs.

Schunk [152] has suggested that there is a need for more research aimed at improving students' self-regulatory skills as they are engaged in learning and to examine how learning environment contexts affect the amount and type of self-regulation displayed. In a study with immersive learning activities authors reported distractions to learning when students behaved as if they had forgotten what their ultimate learning goal was; they repeated actions without any particular meaning[142]. The reason can be because the students have been absent-minded about their learning task [142]. Virtual environments provide better learning experience when students self-regulate as the activities are learner centric and subject to learner behaviour [153]. Students with higher self-regulatory skills tend to be more academically motivated and display better learning [154]; this is relevant to the MUVE support learning as well since the students often have to follow exploratory and collaborative forms of learning.
Usually, students expect the teacher to tell them what to do, how and when to do it, and when to stop doing it, yet too much external regulation (forced) can introduce negative results in their learning [155]. This was seen as a crucial factor for MULEs since the advantages of MUVEs that facilitate a range of student supportive modern pedagogies also require adequate levels of guidance and environment management and allow students to regulate their learning. Too much external regulation in the form of environment management can inhibit the advantage of MUVEs while too little can be prone to fail in achieving the learning goals. The research focus is pitched on this invaluable and yet to be explored area of study in the quest to support academics and students to have better and managed learning in MULEs.

A modified self-regulated learning model has been developed for augmented environments supporting adult learning [156]. The authors highlight the need of tailored self-regulatory approaches to fit with the domain of learning environment. Wan and Reddy [157], have examined self-regulated learning practices combining the idea of community of learning in 3D virtual environments (Second Life). Their views indicate the practice of self-regulatory learning within a community of avatars of the students in MUVE based learning. Moreover, they have argued that the conventional self-regulatory learning model that is based on the individual student should be extended for MUVE learning since student avatars are part of the community. Students are not only responsible for their individual learning goals but also for the community that they are part of [157]. This indeed supports the validity of our research focus where self-regulation should be part of the MULE management practices, not only in the form of learning but also for the environment interactions and avatar behaviours in-world. Further discussion on self-regulation is in Chapter 4.

2.8 Concluding remarks

The focus of this research is user support strategies for managed learning with MUVEs. This chapter therefore presented related literature on MUVEs (SL/OpenSim) and their educational uses, Technology Enhanced Learning with a particular interest in blended learning, learning theories with a particular focus on how they are applied in MUVE education, management practices of learning environments and the usability and behavioural aspects of learning activities. The reviews of literature discussed in each of these topics present the current state of those research areas. These theories and previous study findings are considered as relevant for user support models and tool development in this study, which will be discussed in later chapters.
3. A MUVE for Learning - System Study

3.1 Introduction

This chapter presents an overview of the system studies performed on the selected MUVEs, Second Life (SL) and Open Simulator. The objective of this chapter is to discuss how the SL and OpenSim designs impact users and learning activities. Sections 3.2 and 3.3 describe the relevant information about SL and OpenSim. Section 3.4 discusses a set of particular client applications. Section 3.5 presents the findings of the studies on SL and OpenSim. These system studies were performed with a particular interest in how their architectural norms and functions can affect educational uses; the resulting challenges identified are discussed in Section 3.6. Section 3.7 presents the findings from two studies on student learning from user feedback and usability evaluation.

3.2 Second Life

Second Life requires its client, e.g. the SL viewer, to be installed at the user machines. The environment simulated in SL is dynamic and shared between the active users in that instance; therefore, client programs have to rely on the SL simulation servers to get the simulation updates as all the simulations happen at the server side. User inputs through the client UI widgets are sent to the server and processed simulation data are sent to the client viewer to generate the rendered virtual world. Since SL follows distributed client-server architecture with multiple servers for various services such as user management, Instant Messaging, database services and simulation (Sim), and these are connected through the Internet the network latencies can affect the usability. Although this system model allows Linden Labs to maintain its exclusive control, end user experience and system usability can be affected due to network conditions. Linden Labs made its viewer code open source since it has no significant implementation details about the SL server environment.

Second Life is free to access by registering for an account through the user registration portal. Users are authenticated through the Viewer programs and provided with avatars which can be modified to an extent. There were originally two separate systems for different age groups: Second Life is for adults, and Teen Second Life was for users aged between 13 and 17. In December 2010, Teen SL grid was closed and Second Life reduced its minimum age to access the system to 16 years.

Free users do not own virtual land, but they can create content and keep those in their inventory. The 3D environment in SL is based on regions that are virtual islands with 256m x 256m (65,536 m² of virtual space). Virtual regions or part of a region can be purchased for a subscribed time period. Although a land area is subscribed by a user, it is treated as a part of the Linden Estate; only Linden Labs can overrule land owner actions. Second Life has one of the largest MUVE virtual economies at present. It operates through the virtual currency known as Linden Dollars (SLD). Economic transactions happen through the Second Life Marketplace [23] and users can convert their real money to SLDs.
Marketplace mediates the sales and purchases of land, content objects, and different services in-world, through the transactions based on $LDs.

Content objects in SL, also known as prims, can be any type from the available shapes: cube, cylinder, prism, sphere, torus, tube, and ring. Additionally, three derived versions from these basic shapes, i.e. tetrahedron, hemi-cylinder and hemisphere, are also included. For complex and organic shape creation SL provides the facility to build sculpted prims by uploading a UV map\(^1\) representing the 3D object shape. Content objects can be enhanced by associating textures, Notecards and scripts; scripts, which are written using Linden Scripting Language (LSL), determine the behaviour of the objects and how they interact with the users and the environment. Figure 3.1 shows a content building session in SL during the SL event “Learn Teach and Play in 3D Virtual Worlds” \([158]\), organised by the Centre for HCI, City University and sponsored by JISC EMERGE.

![Content development session in Second Life](image)

**Figure 3.1:** Content development session in Second Life

**Linden Scripting Language**

Linden Scripting Language (LSL) is the programming language used in Second Life. LSL programs (scripts) can control the behaviour of in-world objects. LSL has over 300 built in library functions available for users. Most of these functions provide basic forms of API (Application Programming Interface) for interacting with the system IO devices and in-world objects to create dynamic behaviour. Like many other programming languages, LSL allows users to define additional functions, but with a limited scope restricting the control flow to the content object the script is associated with. Second Life does not provide a localised persistent data storage facility. This means the users are not able to store virtual world information outside the Linden servers. Reasonably good documentation about LSL can be found at the LSL Portal \([159]\).

LSL is a state-event driven programming language. These scripts contain variables, function definitions and named states. Each state contains a description of how to react to events, which occur while the program control is within that state. States are defined on mandatory association with the object the script is embedded in. This approach of state based abstraction may be more suitable for simple

\(^1\) A UV map is a 2D texture representation of a 3D model. The texture contains necessary information to generate the 3D model as a sculpted prim.
structures such as 3D objects that simulate various states in-world, but many have reported the limited capabilities of LSL due to this approach, among other reasons, which prevented them from implementing advanced programmes for complex requirements. Cox and Crowther have investigated the LSL limitations, and the challenges faced by Second Life users when they use LSL for their in-world scripting needs. According to them, LSL generates considerable lag in response and user experience due to the significant number of listeners used in chat channels [160]. Moreover, the inability to call methods in other scripts has given many developers an unnecessarily difficult situation when they try to develop substantial programs. This was experienced in a number of educational projects we have been engaged in, along with the challenges such as limited script size of 16KB (including script, data, heap, and stack [160]), and comparatively limited support for scripting, testing and debugging. Because of the limited script size, quite often, users must have a large number of scripts in a given object if they need complex behaviour; this is not good programming practice.

3.3 Open Simulator

OpenSim is being developed using the .NET framework, and is available with a BSD public license [161] (the latest version is 0.7.4.1). With the recent architectural changes in the server side to increase scalability and to support different technologies OpenSim is now oriented towards being a more generic platform for MUVE development. Since OpenSim uses the same set of viewer applications that are derived from SL Viewer, users do not experience a difference in the simulated environment and content objects between OpenSim and SL. However, at the server side it provides two different options to be used as configurations: standalone mode and grid mode. The standalone mode is designed to cater for small loads and ideally for personal uses. On the other hand, grid mode allows higher levels of scalability on load and more autonomy on separating services and back-end data from the simulation. Fig. 3.2 shows the high level architecture of the two modes; with the grid mode several OpenSim simulations can be hosted on different hardware and connected to the common services.

OpenSim gives an excellent opportunity to researchers who are studying generic MUVE server architectures and essential services. In brief: i) Physics Engine handles the mathematical models and physical calculations for simulating a real environment, avatar and object behaviour; ii) Grid Service facilitates virtual world region layout and coordination between regions, including support for regional teleports; iii) User Service facilitates user authentication and authorisation to the simulated world; iv) Asset Service helps to maintain a reference copy of all asset types (content); and v) Inventory Service manages the in-world content objects by referring to the Asset Service. Support services such as Account, Profile, and Group, etc. are helpful for additional functionality and different applications. Simulated environment update happens with a timed clock mechanism with three main steps: object update based on user input; physics engine processing and update; and finally sends the updated object data to the clients. It usually happens 20 times per second and the lowest rate is 10 cycles per second [28]. This can cause a slight delay within an environment update cycle; however, this delay is acceptable for social virtual worlds in contrast to games.
Content types and available functions for content management, land ownership and available functions, and avatar activities and user interaction methods available in OpenSim show similar behaviour to the corresponding functions and tasks available in SL. Group activities are not included in the standard server but can be implemented using a separate module; it also shows the same functionality compared to groups in SL. Virtual currency (money) support is also not available in the standard server, but it can be implemented using a separate module. This module based plug-in development facility in OpenSim has opened many paths for advanced functionalities if a user wishes to implement them.

Scripting in OpenSim is similar to SL but provides more options to the users. LSL has been considered as the default programming language, although OpenSim also introduced a scripting language specific to OpenSim known as OSSL (OpenSim Scripting Language). LSL and OSSl often provide the same functionality; OSSL has added advanced functionalities and language supports which LSL does not, however. To enable OSSL, server administrators have to set `AllowOSFunctions = true` and decide the function threat level using `OSFunctionThreatLevel` at the configuration in the OpenSim.ini. For example, certain OSSL functions (such as `OSTeleportAgent`) operate only if the allowed threat level is severe (the maximum threat level out of 8 scale range, from None to Severe) [162]. Even unintentional use of such functions can be a danger to the region stability. The most important feature with scripting in OpenSim is the ability to integrate C# scripts as the direct approach of programming instead of translating LSL or OSSL scripts into C#. Because of this facility, programmers can comfortably code in C# and link it to the Simulation as a region module or Mini Region Module (MRM). Although the scripts are confined to a controlled environment, the ability to call native C# functions remains a major security concern. This was experienced within a test run where a series of successful calls on static method API, “System.IO.File” allowed for manipulating the server file space [163].
3.4 Browser Applications

To access SL and OpenSim servers, users need to have a dedicated client installed on their machines. For this research, browser applications were considered with special interest since those are the interfaces that students and academics use to interact with and manage their SL/OpenSim supported learning activities. In general, almost all clients that are used for accessing OpenSim today have been more or less based on the Linden viewer source code. The SL viewer functions were designed in line with Linden Labs commercial motives. Therefore, end users are excluded from their server administration. Because of this legacy situation other viewer programmes for accessing OpenSim that are based on Linden Labs codebase do not support user functions and UI widgets enabling users to manage the OpenSim server environment.

Brief overviews of three popular viewers follow.

3.4.1 Second Life Viewer

This is the official viewer application developed by Linden Labs to access SL virtual environment. It was made available to the public with source code under GNU Public Licence (LGPL 2.1) resulting in the open source community starting the project OpenSim as well as a series of client applications with enhanced features. SL Viewer does not provide a Grid Selector (Grid Manager) to select different grids.

Figure 3.3: Second Life viewer - version 3.3.4 (262321)

The initial SL Viewer versions (LL version 1.x – V1) had different UI widgets and menu layout compared to the recent versions (LL 2.x - V2 and LL 3.x -V3). Viewer version 2 and 3 increased the stability at the client end and included enhanced feature support according to the server improvements; a good archive of the release history is available at [164]. SL Viewer (and all other clients that are based on its code) lets users to alter their visuals and parameters that change the view of the simulated environment through preferences at the viewer. Users are allowed to customise key factors such as graphic quality, network settings, and performance parameters to suit their computer hardware and Internet link. A certain level of debugging and an advanced set of functions are provided to support land owners to manage their environment and the content inside the land area. In fact, these are the functions
that academics and students can use for learning and managing the learning environment, which are discussed later. Fig. 3.3 shows the login window and a simulated view from SL Viewer.

### 3.4.2 Hippo OpenSim Viewer

The Hippo OpenSim Viewer [165] was the first officially recommended complete client application for accessing OpenSim environments. Hippo Viewer was based on the Linden Viewer client (LL version 1.23.0) but provided further enhancements for content creation, environment modifications and a Grid Manager. With the Grid Manager facility, a single client can be configured to connect to multiple grids. Facilities to build prims up to 256 m in each direction, selecting and linking more than 256 prims together at once, building up to 10,000 feet, and transparency with 100% were included in this client. The latest version 0.6.3 (based on LL 1.23.5) released on April 30, 2010, supported Microsoft Windows (XP, Vista and 7 versions) with English as the default language. Fig. 3.4 shows the login screen and a simulated view from Hippo Viewer. However, the project, an open source project under GPL, appears to have become unmaintained.

![Figure 3.4: Hippo OpenSim Viewer – version 0.6.3](image)

### 3.4.3 Phoenix and Firestorm Viewers

The Phoenix Viewer [166] is based on the original Linden Lab Viewer version 1.23.5 and Snowglobe 1.5 codebase; the Firestorm Viewer [166] is based on the Linden Lab viewer 2 and 3 codebase. With the support for grid selection and added enhancements these two viewers are popular among OpenSim users. Apart from the user preference to have a client with features and UI widgets similar to Viewer 1 (LL 1.X viewers), Phoenix Viewer has been preferred for its stable functions when LibOpenMetaverse [167] (a collection of .NET libraries for interacting with 3D virtual worlds) is used. The Firestorm viewer with Viewer 3 (LL 3.x viewers), as the successor of Phoenix Viewer, showed a rapid increase in usage, and in September 2012, it became the top viewer in SL statistics based on usage sessions [166]. Both viewers are developed to support multiple operating systems (Windows, Linux and Mac at present) and remain as open source projects. Fig. 3.5 shows the login screen and a simulated view from Phoenix Viewer (v. 1.6.1).
3.4.4 Other Viewers

There are a number of viewers available. A good source of information about the client viewers along with a matrix comparing the key features of the viewers is available at [168]. Few client viewers are based on LL V1 only, but most of the others have been developed to support LL V2/3 incorporating recent trends in client development. These viewers have varying levels of features such as support for OSSL, Mesh, etc. [168]. Some of the popular viewers include RealXtend [169] Imprudence (LL V1), Kokua (LL V2/3) [170] and Singularity Viewer [171].

3.5 Overview of SL and OpenSim System Models

Due to its open nature most of our system studies were based on OpenSim. SL system behaviour is mainly observed through the viewer interface and available client side functions. Nevertheless, the ownership and permission models of SL and OpenSim are reasonably similar [163]. The research findings of these two systems can be summarised using the following categories.

- Access Control and User Management
- Land Management
- Content Management
- Group Management
- Avatar Activity Management
3.5.1 Access Control and User Management

SL and OpenSim use three different categories of Role Based Access Control (RBAC)\[172\]: Land related RBAC, Content related RBAC and Group related RBAC. These are not mutually exclusive; in fact, varying levels of interdependencies can be seen with prominence often given to the Land related RBAC. Furthermore, when combined with the possible avatar activities and user management functions, all these create a complex and challenging scenario for managing these MUVEs.

OpenSim does not enforce a specialised RBAC for server administration, but relies on the user access management provided by the hosting server operating system. Therefore, any user with access to the server (physically or remotely) can perform administrator functions without any restriction; this can be a challenge for mainstream uses that require higher trust and security. Since OpenSim is still in the pre alpha stage releases, the development community may have higher priorities than introducing server access RBAC with authentication and authorisation. User management and access control of SL/OpenSim partly depend on avatar naming: SL gives the option only to choose the first name of the avatar - the last name is restricted to a list of given names - whereas with OpenSim, users have more autonomy in deciding avatar names with varying levels of anonymity.

3.5.2 Land Management Models

MUVEs in general associate different levels of land management features depending on their system implementations and business models. For SL and OpenSim land management is the most important function category, however. User roles in SL and OpenSim are often defined considering different levels of land ownership and access to a given land. SL and OpenSim follow the same model of land management based on virtual regions (256m x 256m size of virtual land). As Table 3.1 summarises, the relationship between different land units in these MUVEs can be stated as: a virtual world consists of one or more estates; an estate consists of one or more regions, and a region consists of one or more parcels. Different owner roles associate with these units for SL and OpenSim are shown in the table.

<table>
<thead>
<tr>
<th>Land Type</th>
<th>Owner (SL)</th>
<th>Owner (OpenSim)</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual World</td>
<td>Linden Labs (LL)</td>
<td>System Administrator</td>
<td>The entire simulated MUVE</td>
</tr>
<tr>
<td>Estates</td>
<td>LL for Linden Estate / Estate Owner</td>
<td>Estate Owner</td>
<td>One or more regions</td>
</tr>
<tr>
<td>Regions</td>
<td>Region Owner</td>
<td>Region Owner</td>
<td>65,536 m²</td>
</tr>
<tr>
<td>Parcels</td>
<td>Parcel Owner</td>
<td>Parcel Owner</td>
<td>Min. Size 16m²</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Max. Size 65,536m²</td>
</tr>
</tbody>
</table>

Table 3.1: A Comparison of land units and ownerships in SL and OpenSim
The regions are located on a 2D Cartesian map with unique X and Y coordinates for each region. Fig. 3.6 shows the map view of a set of regions in SL. Within a region, position vectors are in 3D, i.e., \( \{x, y, z\} \). OpenSim offers a facility to combine regions together as a one region, which is known as a Mega-region. SL facilitates this feature by rescaling the estate land area if a land owner wishes to expand the land by purchasing more adjacent regions later. In Mega-regions, position vectors have one origin and therefore, it can be challenging for backing up and changing existing content and terrain.

![Figure 3.6: World Map of several regions (SL): A square is a region of 65,536m²](image)

Since land is treated as the core resource in SL and OpenSim, user accesses and various interactions with the environment are defined in the context of land. It is important to note that even for an avatar to access its inventory and friends it has to be in a land area (when users login, they are first located at the place they request or the nearest land available). This close relationship of simulation based on land is an important factor to keep in mind when defining various management practices for learning with SL and OpenSim.

### 3.5.3 Content Management Models

Content management is the second most important function category for SL/Opensim management. It involves the functions that are required for managing content objects and their permissions on different content related user roles. The Land-related RBAC model often gets prominence over content-related RBAC when manipulating and executing content objects, although these two RBAC models are defined in completely different contexts. As a result, land owners at different levels can manipulate the content related functions such as script execution, content creation, media streaming and content access inside their lands, provided that there are no conflicts between the content related functions at land levels (estate, region and parcel levels). The following content ownership and permission models were identified as part of this system study.
Content Ownership and Permission Models

SL and OpenSim follow similar ownership and permission models on content management. The administrator privileges in OpenSim are available for Estate Managers & Estate Owners and any user with allowed access, which is available only for Linden Labs employees in SL. The other content management roles and permission arrangements are identical in these two systems. The following roles and permission settings were identified (Table 3.2).

<table>
<thead>
<tr>
<th>Role Types</th>
<th>Task Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creator</td>
<td>Move (V)</td>
</tr>
<tr>
<td>Group</td>
<td>Copy (C)</td>
</tr>
<tr>
<td>Owner</td>
<td>Modify (M)</td>
</tr>
<tr>
<td>Next Owner</td>
<td>Transfer (T)</td>
</tr>
<tr>
<td>Everyone</td>
<td></td>
</tr>
<tr>
<td>Administrator</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.2: Roles and task types in content management

The possible tasks on content are defined in the context of 3D environment features. Moving a content object can be done in the 3D space within the region boundaries; it is denoted as Move (V) action on the content. Similarly, modification of a given content can be done within the allowed maximum dimensions of an object, which is usually 64 m in each direction in SL (i.e., $x<64\, m;\, y<64\, m;\, z<64\, m$) but can go up to 256m in OpenSim. Modification of a content object includes resize, reshape, retextrue and various other available activities including the ability to change the permissions of the content. This is denoted as Modify (M) in the permission model. Copying a content object automatically places a duplicate copy of the original object into the user’s inventory; the user can then take that copy into the region from the inventory. However, if a user wants to duplicate an object in-world dragging the object while pressing the Shift key duplicates the selected object, provided the user has permission to copy. This ability is denoted as Copy (C) in the permission model. Finally, the ability to transfer an object to another user is also considered as important since it nurtures the content-based SL economics. In Second Life usually the transfer function comes with the selling option where a price in $LD$ has to be set before making the content for sale. However, with OpenSim, content transfers can be conveniently done as content owners wish, without the effect of virtual currency. Content transfer is denoted as Transfer (T) in the permission model. These four permissible activities are known as the VMCT model of the permissions in SL and OpenSim. Apart from these four tasks, one of the default abilities is to delete content objects in-world. This can be done by anyone if the object is not locked, or by the owner of that object, which is discussed later in the section on Fair Ownership.

The role content Creator is the user that has created a content object in-world. This role represents the first owner of the object and quite often when an object (prim) is created, the roles Creator and Owner are represented by the same user. Owner (O) represents the role that owns the content object for a given
instance. To facilitate the current owners in setting their objects with correct permissions before letting others have those, a pseudo role, known as Next Owner (N), is also available. Content objects can be set for a group so that members of that group get access to that content, which is denoted as Group (G) in the permission model. Members of the group with the ability to manipulate objects set to that group can perform the tasks \{VMCT\} set for the Group permission role. Everyone else (i.e., any user either a resident or visitor) can be allowed to perform the VMCT tasks on a content object, in this case the permission role is represented as Everyone (E). Table 3.3 summarises the permission model with roles and states.

Another two unique states relevant to the SL (and OpenSim) permission model are Base Permission (B) and Effective Permission (F). Base Permission represents the maximum possible permission level that a content object has for a given instance. Often, it is equal to the full permission on VMCT when a content object is created; hence, for that instance Creator role is represented by B. However, as the content objects being transferred from one owner to another with permissions getting altered usually, the B can get decreased from its initial state. For example, at the beginning an object has \{VMCT, all-true\} and later becomes B = VCT (without the ability to modify). Once this happens, without the administrator functions (God tools) users cannot elevate the B state to its previous higher level. Effective Permission (F) is a dynamic state, which shows the effective permission level a user gets on a content object considering all the combinations and B for that given instance. Therefore, it was identified that \(F \subseteq B\), and at the instance where the object is created \(F = B\).

With these six possible roles and states, SL and OpenSim define their content ownership and permission roles as \{B, O, G, E, N, F\}. Also the possible abilities are defined as \{V, M, C, T\}. These 6 roles and 4 abilities create a complex ownership and permission model; SL does not explicitly present the complete model. With a series of test cases, this research identified the complete model behaviour and observed the same with OpenSim as well [173], which is summarised in Table 3.3. The default states are shown in bold, and the administrator role (only in OpenSim) is included to show the ability to change the model through God Tools. Since the Effective Permission (F) is a dynamic state that depends on the selected permissions (True/False) on VMCT and BOGEN combinations, it was not included in the model but can be easily found for a given combination of roles and abilities by referring to the model.

According to the identified content management model, without ownership \{O, N\} users do not get the right to transfer the object to another user. Also, unless M is set as true for G, G and E do not get permission to modify a content object. Furthermore, owners \{O, N\} always have the permission on V.

It was observed that the default permission arrangements are set to safeguard the owners if they accidently leave their objects in-world without setting the correct permissions. \{G, E\} always get VMCT - false while \{O, N\} always get VMCT - true as the default states, unless changed otherwise. Administrators (God) in OpenSim environments can use God Tools to force an object under their ownership irrespective of the VMCT settings, and with the ownership and God level access, they can then modify the MCT settings.
<table>
<thead>
<tr>
<th>Role</th>
<th>Move (V)</th>
<th>Modify (M)</th>
<th>Copy (C)</th>
<th>Transfer (T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creator</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>Owner</td>
<td>True</td>
<td>True/False</td>
<td>True/False</td>
<td>True/False</td>
</tr>
<tr>
<td>Next Owner</td>
<td>True</td>
<td>True/False</td>
<td>True/False</td>
<td>True/False</td>
</tr>
<tr>
<td>Group</td>
<td>True/False</td>
<td>True/False</td>
<td>True/False</td>
<td>True/False</td>
</tr>
<tr>
<td>Everyone</td>
<td>True/False</td>
<td>False</td>
<td>True/False</td>
<td>False</td>
</tr>
<tr>
<td>Administrator</td>
<td>Overrule</td>
<td>Overrule</td>
<td>Overrule</td>
<td>Overrule</td>
</tr>
</tbody>
</table>

**Table 3.3:** Content Ownership and Permission Model of SL and OpenSim

Further, three more content management models in SL/Opensim were identified; their impact on learning was observed in the system studies.

**Composite Permission Behaviour**

The economic model presented in SL shaping its system architecture, which was then followed by OpenSim as well, encourages users to create complex content objects in a composite manner. Usually, a content object is a prim as discussed previously. However, SL and OpenSim support 9 different content types as shown in the Table 3.4. Body parts, Clothing and Gestures are effectively useful only when they are worn or executed by avatars. Animations can be used to animate either an avatar or an object. Scripts, Sounds, Notecards and Textures are treated as supportive content types to enhance Objects. To increase the user engagement with the 3D environment content models always give Objects precedence in use; Scripts, Textures and Notecards have to be always associated with an object, in-world. However, any of these content types can be stored in the user inventory as individual items, but to use them in-world they have to be associated with an object or an avatar.

<table>
<thead>
<tr>
<th>Animations</th>
<th>Objects</th>
<th>Gestures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body parts</td>
<td>Scripts</td>
<td>Notecards</td>
</tr>
<tr>
<td>Clothing</td>
<td>Sounds</td>
<td>Textures</td>
</tr>
</tbody>
</table>

**Table 3.4:** Content types available for composite permissions in SL and OpenSim

To manage composite content objects SL and OpenSim allow users to practice finer granular permissions on each of the component in a composite object. Fig. 3.7 shows the user interface (SL viewer – version 3.3.4 (262321)), that allows content owners to decide permission settings on individual content types in a given Object on the role categories Group, Everyone and Next Owner.
When a user decides various combinations of permission settings for a composite content object, often, the permission settings on the Object (prim) have precedence. However, interestingly, when the permissions were set for Next Owner, as shown in Table 3.5, on a given object with an embedded script, a peculiar behaviour of composite object permissions was observed. In this case of permission arrangements for MC-false permissions, Script has the precedence in determining the MC permission of the composite object, i.e., false. For Transfer, Object gets the precedence. If the permissions are swapped (Object - CM false and Script - CM true) the composite object has CM false, as per the usual behaviour. This is reasonable since a script always has to be inside an object, and if the object has no permission given to Next Owner on CM (false) there is no effective meaning to make the composite object with CM true even if the script is allowed to be copied or modified.

<table>
<thead>
<tr>
<th>Next Owner</th>
<th>Copy</th>
<th>Modify</th>
<th>Transfer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>Script</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>Composite object</td>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
</tbody>
</table>

Table 3.5: Composite permission model for objects with scripts

Cyclic Permission Loss

This is one of the crucial phenomena relevant to the content management in SL/Opensim. When a user creates a content object, according to the permission model, that user has the full permission on that object; in that instance, the user is treated as the creator of the object (Creator role), as well as the owner (Owner role) by default. Effectively the Base permission of the object has the VMCT settings with all-true states. If the creator transfers this object without any modifications to the permission model, the user that receives the object should (i.e., Next Owner permissions are also VMCT with all-true states)
get the full permission; hence the Base permission is not affected. As the third step, let us say that the recipient of this object transfers the object back to its creator without altering any of the VMCT states. Upon the completion of this cycle of object transfer the creator of the object still has the full permission over the object since the Base permission was not reduced during the process (i.e., Base Permission = Effective Permission = VMCT).

In a similar scenario, let us consider if the creator of the object decides to alter the VMCT permission settings for the role Next Owner. The lowest level of permission settings that enable this scenario is for the VT settings to be true (i.e., Move – true & Transfer – true) for Next Owner. With these settings, Base permission has been irreversibly reduced to VT and that is the maximum possible permission state the object can then have during the forthcoming transfers. With this new permission setting (VT), let us say that the object recipient decides to transfer back the object to its original creator without modifying the permission settings. Essentially, while keeping the ability to transfer, this is the lowest possible permission state an object can be, therefore, the present owner cannot modify the object permissions further. When the original creator of the object receives it back he or she is treated as the owner (Owner role) but not as the creator for deciding permissions. Because of this reason, the creator gets only the VT permissions and cannot modify or copy that object (Base Permission = Effective Permission = VT). Fig. 3.8 shows the overview of Cyclic Permission Loss scenario.

![Diagram](image)

**Figure 3.8:** Cyclic Permission Model

**Fair Ownership Model**

This model ensures that the content object owners have permission to move or delete their content without restriction (Fig. 3.9). Delete can be performed either in-world or inside the inventory, whereas moving of an object in this scenario is defined only in-world. Object moving is often possible even when the owner is away from the object location; however, the object selection is possible only if the distance between the user and the object is less than 128m in the 3D space (i.e., Euclidian distance hence, \( \sqrt{(x_o - x_a)^2 + (y_o - y_a)^2 + (z_o - z_a)^2 < 128m} \); where \( O = \text{Object} \) and \( A = \text{Avatar, positions} \). Therefore, indirectly, users get a limit on the maximum distance that they can move an
object they own in a single move. Moreover, if the object is locked, to move or delete the owners first have to unlock their objects.

![Diagram](image)

**Figure 3.9**: Fair Ownership Model

### Content in Inventory

Another important content object management area in SL/OpenSim is the inventory of an avatar. The Avatar inventory is a personal repository that the user can control exclusively. Therefore, client applications do not provide administrator or land owner functions to access or control other users’ inventories. Users can decide the ways to structure their avatar inventories and move different content types within the inventory folder hierarchy. Content objects inside an avatar inventory enforce the fair ownership model; hence, the inventory owner can delete any content object in his or her inventory without any permission or ownership on that content. This is an important feature of both SL and OpenSim: content objects are treated in two different ways depending on whether those are inside a region or part of a user inventory. Locked objects have to be unlocked even if the owner wants to delete, in-world. However, inside the avatar inventory all content objects are treated as entries in the inventory, giving the owner of the inventory exclusive rights to decide the management options for their inventory. On the other hand, to avoid ownership conflicts, while within an avatar inventory, content objects can only be moved between folders and copied if permitted, but cannot be modified or change their permission levels.

### 3.5.4 Group Management Models

Support for groups in Second Life is a built-in feature of the application service. As part of the SL virtual economy promoting regime, users have to pay 100 $LD when a group is created; further if no user other than the group creator joins with the group within 48 hours from the group creation, the group automatically gets disbanded [174]. OpenSim does not provide a built-in group facility with its server, and the administrators must install a third-party group implementation module at the server end.
with database modifications. Unless the OpenSim based MUVE has a commercial motive, users can freely create groups and become members of other groups without any monetary involvement. This can be seen as an added advantage of using OpenSim for educational needs.

Groups in SL are managed by Linden Labs. SL users access the Group service through the client application. In contrast, if OpenSim is used, the system administrators can have two options to select for enabling a group facility with their OpenSim environments based on the implementation architecture: 1) XMLRPC groups (Flotsam Project) or 2) SimianGrid groups (SimianGrid project). For this system study and the OpenSim based activities, XMLRPC group implementation was used due to its reliability and wide usage. OpenSim group implementation has three major components: the Group Service, the Group Service Connector and the Group Module. Fig. 3.10 shows the high level system architecture of the OpenSim group implementation with these three components. This implementation architecture allows the flexibility to plug-in different implementations for the Group Service component; OpenSim, however, warns about possible protocol incompatibilities if such implementations are not based on OpenSim support protocols [175].

![Figure 3.10: System architecture of Groups in OpenSim](image)

Apart from this server side implementation and administration of the Groups module in OpenSim, SL and OpenSim group functions are identical to each other at the client end. There is no difference between managing groups in SL and OpenSim. In fact, if the same client is used to connect SL and OpenSim, a user would experience the same set of activities in group creation and becoming a member of a group.

Tay has examined social groups in Second Life and indicated that SL encourages the formation and development of avatar groups with members having similar interests; the group members can share space, objects and create an environment for group activities [176]. This view also highlights the fact that groups in SL (and in OpenSim) are mainly based on the 3D environment (land) and content objects, hence the nature of group related functions is that they are closely associated with land and content management.

Three default roles with respect to group management were identified: Group Owner, Officer, and Member. The Owner role represents the owner of the group; in fact, the user who creates the group automatically becomes the first owner of that group and they can add others as owners of that group. Group Owners have full permissions for managing the group, including all the functions the group
service offers at the client side. Users can be invited to become members of a group, represented by the role of Member (the lowest privileged category). Group Owners can make members Officers (a middle level privileged user category) to delegate a selected set of group management tasks. Through a study on viewer interface and widgets, 13 major categories of group management functions were identified. A summary of these functions is shown in Table 3.6.

<table>
<thead>
<tr>
<th>Function Category</th>
<th>Abilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Membership</td>
<td>Invite People to this group</td>
</tr>
<tr>
<td></td>
<td>Eject Members from this Group</td>
</tr>
<tr>
<td></td>
<td>Toggle ‘Open Enrolment’</td>
</tr>
<tr>
<td>Roles</td>
<td>Create New Roles</td>
</tr>
<tr>
<td></td>
<td>Delete Roles</td>
</tr>
<tr>
<td></td>
<td>Change Role names, titles &amp; descriptions</td>
</tr>
<tr>
<td></td>
<td>Assign Members to Assigner’s Role</td>
</tr>
<tr>
<td></td>
<td>Assign Members to Any Role</td>
</tr>
<tr>
<td></td>
<td>Assign and Remove Abilities in Roles</td>
</tr>
<tr>
<td></td>
<td>Remove Members from Roles</td>
</tr>
<tr>
<td>Group Identity</td>
<td>Change Charter, Insignia &amp; toggle member visibility in Group information</td>
</tr>
<tr>
<td>Parcel Management</td>
<td>Deed and buy land for group</td>
</tr>
<tr>
<td></td>
<td>Abandon land</td>
</tr>
<tr>
<td></td>
<td>Subdivide and Join Parcels</td>
</tr>
<tr>
<td>Parcel Identity</td>
<td>Set category and toggle ‘Show in Find Places’</td>
</tr>
<tr>
<td></td>
<td>Change parcel name and description</td>
</tr>
<tr>
<td></td>
<td>Set Landing Point and Set Teleport Routing</td>
</tr>
<tr>
<td>Parcel Settings</td>
<td>Change music and media settings</td>
</tr>
<tr>
<td></td>
<td>Toggle ‘Edit Terrain’</td>
</tr>
<tr>
<td></td>
<td>Toggle About Land Options</td>
</tr>
<tr>
<td>Parcel Powers</td>
<td>Always Allow ‘Edit Terrain’</td>
</tr>
<tr>
<td></td>
<td>Always Allow ‘Fly’</td>
</tr>
<tr>
<td></td>
<td>Always Allow ‘Create Objects’</td>
</tr>
<tr>
<td></td>
<td>Always Allow ‘Create Landmark’</td>
</tr>
<tr>
<td></td>
<td>Allow ‘Set Home to Here’ on group land</td>
</tr>
<tr>
<td>Parcel Access</td>
<td>Manage Parcel Access and Ban Lists</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Eject and Freeze Residents on Parcels</td>
</tr>
<tr>
<td>Parcel Content</td>
<td>Return objects owned by group</td>
</tr>
<tr>
<td></td>
<td>Return objects set to group</td>
</tr>
<tr>
<td></td>
<td>Return non-group objects</td>
</tr>
<tr>
<td></td>
<td>Landscaping using plants</td>
</tr>
<tr>
<td>Object Management</td>
<td>Deed objects to group</td>
</tr>
<tr>
<td></td>
<td>move, copy &amp; modify group owned objects</td>
</tr>
<tr>
<td>Accounting</td>
<td>Pay group liabilities and receive dividends</td>
</tr>
<tr>
<td>Notices</td>
<td>Send Notices</td>
</tr>
<tr>
<td></td>
<td>Receive Notices and view past Notices</td>
</tr>
<tr>
<td>Proposals</td>
<td>Create Proposals</td>
</tr>
<tr>
<td></td>
<td>Vote on Proposals</td>
</tr>
</tbody>
</table>

| Table 3.6: Abilities available in Groups and their function categories |

By default, Members only have the abilities to Create Proposals, Vote on Proposals, Receive Notices and view past Notices, Pay group liabilities and receive dividends, and to Set Home to Here on group land. The Officer role has all these permissions except the abilities to Create New Roles, Delete Roles, Assign Members to Any Role, Remove Members from Roles, and Assign or Remove Abilities in Roles, in the default conditions. Owners get complete abilities on all the functions. This scenario may appear straightforward to use; however, there is a provision to define new roles in a group, which can cause significant challenges depending on the context of use.

### 3.5.5 Avatar Activities

Avatar activities are the main form of user interaction with MUVEs. These activities are performed by the authenticated users through their selected client application. Although the tasks and various functions that were discussed in previous sections under land, content and group management topics can also be seen as a set of avatar activities, the activities that come under this category have a subtle difference in their nature. I.e., there is a limited facility to regulate most of these activities, and even if controlled, they can severely affect the free and attractive user interactions that are available for an engaging experience.

21 unique avatar activities were identified in OpenSim/SL for users to interact with the environment. Content creation & manipulation, and terraforming were also considered due to their significance in avatar interactions, which are part of the content and land management, respectively. The rest of the identified avatar activities are in the types of avatar communications, avatar mobility, and avatar presence and appearance related activities.
Avatar communication activities such as text chat including Instant Messaging, voice chat, and gestures are designed to provide collaboration opportunities with other users. Text chat messages cannot be restricted at the application level and are considered a primary method of avatar communication. Individual users get their preferred settings on how they see and manage their chat communications through the client application. Voice communication and voice streaming at the parcels are considered as optional ways of communication; therefore, land owners can restrict voice as an option. Gestures are another form of available avatar communication with the facility to have user defined gestures. By default, the gestures cannot be restricted at the application level; however, if the script executions and voice communication are disabled, the gestures that require these abilities would be indirectly blocked.

Avatar appearance change is unrestrictedly available to users to perform at their discretion - it is not expected to be managed by the land owners or the system administrators. Avatar mobility activities such as flying and teleporting can be controlled as part of land management. Extreme levels of control on avatar activities are also available such as freezing, which prevents all possible actions, or kicking it off a simulation.

### 3.6 Challenges for Learning

This section discusses how the previously presented SL/OpenSim system models can introduce challenges for managed learning requirements in SL/OpenSim.

**Access Control and User Management**

One of the major challenges is mapping roles between existing educational infrastructure and SL/OpenSim. This requires careful considerations on expected functionality and integration techniques between e-Learning systems and SL/OpenSim. This will be further discussed in Chapter 4.

Avatar naming is an interesting topic in the context of MULEs. Messinger (et al.) [177] have discussed the effect of avatar naming practices on user behaviour. “Individuals who “hide” behind their avatars cannot be easily identified, allowing virtual worlds to provide a certain degree of anonymity” [177]. The level of anonymity, i.e., de-individualisation [177], is higher in SL compared to OpenSim; Moreover, because of the user naming strategy of SL, it can be challenging to map avatar identities with student identities, if the avatar names do not represent sufficient information about the real user. A technical solution through linking SL avatars with the university’s Module Management System user identities was practiced using SL RegAPI; however, students have to follow a series of additional steps in this arrangement [88]. With OpenSim, the student first name and last name can be used as the First name and Last name of the avatar, respectively, making the user mapping more convenient [98]. In their study, Messinger [177] concluded that they did not find exceptionally different behaviours in the virtual-world compared to the real-world behaviour, but acknowledged the fact that avatars followed a less restrained practice from their normal inhibitions. Further investigations on using real names for avatars in shaping student behaviour during learning are discussed later.
If OpenSim is integrated with an existing e-Learning environment which has high security enforcement for accessing a server environment, the OpenSim server might be seen as a weaker entry point that compromises the security of the university learning infrastructure; it can be a concern for certain academic needs.

**Land Management Challenges**

The most significant challenge in managing land is to determine how land management affects the other management areas: user, content, group and avatar activity. It is therefore quite difficult to manage land in isolation from the rest. Moreover, users often have to implement such policies using the default land functions and option settings. This can create a significant complexity and high levels of uncertainty for non-expert academics; user support in this regard is essential.

Secondly, different levels of land ownership can create conflicts. Parcel Owners have a similar set of land functions and options settings to Estate Owners, except for the tasks specific for estate and region management. Similarly, Region Owners get the same set of functions as other owners except for functions required for estate level management. A region can be divided into one or more parcels (max. of 4096 parcels of 16m\(^2\) in size for a region and in case of Mega-regions multiples of 4096 will be the maximum number of parcels one can arrange depending on the number of regions in the Mega-region) with different users as parcel owners. There is no limit to the number of regions, and it depends upon the hardware capabilities of the server (SL supports this through adding more Sim servers, and OpenSim supports this through its ROBUST architecture). As the number of different land units grows it gives more opportunities to conduct various teaching and learner support activities for an educational institute by having more virtual space and unique control over the land. However, this brings an unprecedented scale of coordination requirements between the management practices among different land units for avoiding conflict and delivering a coherent learner experience.

The third challenge for education is due to the different objectives of land ownerships and the land functions. Mapping these abilities into the learning activities can be a significant challenge for academics. If web based e-Learning is considered, the 2D web page becomes the virtual space for integrating different web components and learning activities. In OpenSim/SL virtual land becomes this space and different learning constructs and content objects have to be located for simulating the learning platform. Moreover, the simulated environment is based on physical properties similar to the real world, where multiple learning and activity spaces can be defined within the virtual 3D space. Because of this, a considerable reshape of teaching and learning activities can be required along with transforming OpenSim/SL land settings into the educational domain; more details will be discussed in the next chapter.
Content Management Challenges

Content management becomes important when learning activities are linked to assessments. It is important to understand the composite permission model when managing SL supported learning activities. For example, the student activities that involve scripting on their created content can affect the composite permission settings without their knowledge. If they do not know about this special scenario they may submit the completed work for assessment with a different permission set. Lecturers can face difficulties in evaluating these objects without the permissions needed for their grading. This was observed during the system studies and will be explained in section 3.7.

In the Cyclic Permission Loss situation, in order to generate more value from user created content and to reserve the rights of the creators, SL allows the creators to reduce the Base Permission before they sell their content. This was not considered a significant problem by Linden Labs since the content sellers and buyers followed two price levels: a lower price without permissions and a higher price with full permission for their transactions. However, this research identified the significance of the challenge that educators can face due to Cyclic Permission Loss model [173]. OpenSim also follows the same permission model. In fact, this model affected the student learning sessions, which will be discussed in section 3.7. Another important observation was that early SL client application (version 1.xx) followed a default permission setting with Next Owner (VT) when an object is created, making Cyclic Permission Loss the default situation, unless the creators change VMCT to be true for Next Owners before they transfer their objects and expect a return of the same.

Fair Ownership can lead to accidental deletion of the objects by the owners. Since there is no verification mechanism at the client when a user presses the delete key or selects the delete menu after selecting an object, an irreversible object deletion happens. Furthermore, unless the user has a copy in their inventory there is no recovery method available from clients. This can prove critical for content based learning and assessment.

Group Management Challenges

To use the group facility for educational activities, the academics and students have to use the viewer-based group functions, which are quite complex; hence, the management of these functions becomes crucial for the success of SL/OpenSim supported learning. As Tay [176] observed, the functions available for managing groups are not designed to support group-based learning in formal education. Six of the function categories are land management (group owned) related, and another four categories are for content management. Roles and Memberships are the two generic areas which academics can use for supporting group based teaching. One possible option would be to define roles for the context of a learning activity and then delegate abilities to suite the requirements. However, there can be conflicting abilities at land, content and group levels, which would then need resolved.
Avatar Activity Management Challenges

Control on avatar mobility can be useful for specific learning requirements that require students to be restricted to a given area. Further functions such as sit-down and stand-up can be managed through land and content management options. One of the avatar actions in SL/Opensim often disabled is pushing others; it is a concept borrowed from MMORPGs and game character behaviours.

Schroeder [178] found that regardless of the technological constraints and virtual world usage norms, users prefer to have exclusive control over their avatar design and their appearance in-world. However, activities such as changing avatar body shape, body parts, appearance parameters, clothes and wearable items can affect learning. For example, an unlocked educational content object can be worn and moved but also be subject to conflicting behaviour through an indirect effect from the avatar appearance functions.

Extreme levels of avatar constraint can cause severe impacts on the usability of the system and degrade the student motivation to engage positively with learning activities. However, in an exceptional situation, as a reactive strategy to regulate the unwanted behaviour of an avatar that affects the learning environment and other students, these types of controlling mechanisms might be needed. Nevertheless, effective management practices that promote student self-regulation can help to avoid such incidents.

3.6.1 An Approach to Manage Avatar Appearance Change

Although the client applications to access SL and OpenSim are under the exclusive control of the user there can be possible workarounds to discourage users from trying to explore potentially disturbing activities within a learning session. These practices are temporal by nature and if the user decides to change or use another client without these settings, such change would nullify the expected control over the user behaviour. However, if a university managed laboratory of classroom based computers with the installed client applications are used for accessing the OpenSim server, then by removing the possibly harmful UI buttons and menu items through customisation of the skins of the client applications one can disable or remove certain avatar functions. A successful attempt was observed in the Linlithgow Palace project where the school children accessed the virtual reconstruction of the Linlithgow palace for their educational needs. A major concern that the school teachers had was the ability to remove clothes of an avatar and become naked while engaged in the classroom learning sessions. To overcome this challenge, a modified Phoenix Viewer without the menu options to remove or wear clothes was deployed on the school computers. Although, it is not a complete solution to prevent unwanted avatar appearance change, it successfully prevented accidental removal of clothes and intended naked presence since the students do not have the operating system administrator level privilege to change the client settings. Fig. 3.11 compares the standard Phoenix Viewer with the skin modified Phoenix Viewer: the UI wagon-wheel menu for user options of the modified client does not have the options that could change the user appearance, i.e., Take off (to take off the wearable including clothes), Appearance (to change the avatar appearance), and Texture (to modify wearable textures) options.
However, should they wish to modify their avatar or remove clothes they could use their own machines with standard clients installed if the institution allows them remote access the OpenSim environment. Further, they could have a locally installed OpenSim with the virtual reconstruction of Linlithgow palace by loading the .oar file of the project (which is available for public access at [179]). Hence students’ private use of a MULE with servers and clients installed on their personal computers must be distinguished from accessing the institutionally hosted MULE. The former option lets the student be the administrator of the personal MUVE while the latter option requires them to adhere to the institutional policies and academic practices. This research investigates and provides user support for the latter scenario; the study’s contributions may become ineffective if put into the context of private use of these MUVEs.

### 3.7 MUVEs in Student Learning – User studies

A preliminary user study was performed to evaluate user views of MUVEs (OpenSim/SL) as a medium for formal education. For this purpose a Human Computer Interaction (HCI) course module, part of the BSc in Computer Science and Internet Computer Science at the School of Computer Science at the University of St Andrews, was selected.

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 [76]. One of the major learning objectives for this HCI course was to give students an opportunity to understand how to incorporate attractive and creative ideas in user interface design. Two cohorts of students from two successive academic years were used in the study, the first group of students worked with SL and the second group used OpenSim. The findings were also used in the evaluation of the prospect of moving the university MULE from SL to OpenSim.
3.7.1 User Study in Second Life

In brief, students were asked to develop intuitive and engaging constructs in SL to demonstrate Dijkstra’s shunting algorithm [180] which is used to convert mathematical equations from infix to postfix notation. For example: “1 + 2” becomes “1 2 +”, “1 + 2 * 3” becomes “1 2 3 * +”, and “(1 + 2) * 3” becomes “1 2 + 3 *” etc. Postfix notation is frequently used for stack-based evaluation. An optional part of the project was to show the stack-based evaluation of the postfix notation. Students were given private parcels of land with minimal restrictions imposed. This allowed them the ability to exclude some or all other residents, and many chose to do so. Students successfully implemented a range of creative and complex constructs. Fig. 3.12 shows working instances of two student projects.

![Completed student projects in SL - Dijkstra’s shunting algorithm](image)

To maintain an accurate association between the student university identity and their SL identity, the registration API (RegAPI) from SL was used via an interface created within the university learning management system, MMS, allowing students to pick a name from the SL list of names and create an avatar. This ensured the accuracy of the project assessment and grades given to the respective student, since their avatar names did not indicate their real identities or university usernames. Submission of student work was found challenging with the initial arrangement of transferring the work to a specifically created avatar in SL. Students requested help for transferring items to another avatar in-world, and how to convert items in-world into a form that could be "rezzed" and used easily elsewhere. Another problem was that even though SL sent e-mails to the lecturer who owned the avatar which received student work, these never actually arrived. It was difficult to confirm successful submission of student work since SL did not provide a notification system to the sender, nor did it accurately verify the successful receipt of the content. During the assessment of student work further challenges were experienced due to the ownership and permission models; with the SL model, when students transfer objects from their inventories certain parts of their projects were received in a non-functional state under the default Next Owner permission settings (in Viewer 1 clients). Further, without suitable copy permissions the scripts that required to “rezz” multiple copies of the objects to represent different notations according to the given user infix notation could not function properly.
Because of these challenges, a special submission tool was created, which checked the correct permissions and could accurately verify successful submissions. It addressed the challenges inherited from the available means of content transfer in SL. This showed a need to provide more sophisticated mechanisms to facilitate student learning, and training the students about the non-intuitive Second Life model of ownerships, permissions and inventory management.

The low turnover of completed System Usability Scale (SUS) questionnaires (30.76%) from the sample of students (N=13) made the quantitative analysis inappropriate. However, the open feedback from students portrayed their experiences and the view about SL environment, in general. Some of the student views captured through their reports are as follows [88]:

“Creating a physical implementation in a virtual world is somewhat trickier than was first expected.”

“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.”

Students seemed mostly happy with the LSL, which can be frustrating for its lack of language features such as complex data types or exception handling. Students acknowledged the fact that SL, as a MUVE, is a good platform for developing educational aids to explain complex theories and concepts.

3.7.2 User Study in OpenSim

It was decided to explore OpenSim with the next student cohort for the same HCI course module student projects as an alternative MUVE platform. For the projects, students were asked to develop an interactive door system for an enclosure with two doors minimum: one from simple touch-open, access key based or password based, and the other from complex arrangements such as keypad entry, combination access and fire escape door with alarms, etc. A dedicated region was constructed in OpenSim environment and each student was given a 50m x 50m square land parcel with full ownership as the parcel owner. Students could modify their given lands to suite with their enclosure designs. Fig. 3.13 shows two successful student projects with creative ideas incorporated.
The latest version available for the study was OpenSim v.0.6.9; it came with certain limitations due to the lack of support and partial implementations of the system components as it was being developed, making its behaviour less reliable. Most of these challenges were documented in the release notes of OpenSim, however some further unexpected behaviours occurred that affected student learning to a significant level. Land parcel access control was required similar to the work in SL. The same method of preventing public access was used, and it produced the virtual fences surrounding the parcels in the viewer applications indicating the word “No-Entry”. However, it failed to effectively stop students entering the parcels that they were not the owner of. Then, a workaround solution with extra effort was tried by adding all students but the parcel owner to the banned list of each parcel. It behaved reliably during the test runs, but in the actual student practical with multiple student accesses it failed to recognise the access control mechanisms reliably. Moreover, students could bypass the virtual fence and banned list protections by flying across the borders of the parcels. While none of the students were bothered by others having access to their lands in this case there will be learning activities in which parcel access restriction is essential.

With this set of HCI student projects \((N=18)\), an effort to have a quantifiable measure about system usability was made. Student feedback about their experience with OpenSim was captured using a modified SUS in the context of OpenSim use. The standard SUS questionnaire has the word system as the subject of measure but for this analysis it was replaced by the word OpenSim to give more context for the student responses. Further, question items were modified to highlight the key areas of interest in examining OpenSim use for engaged learning, while keeping the basic model of SUS question items pairs. The given questionnaire items and the mean \((\mu)\) and standard deviation \((\sigma)\) values for each statement are shown in Table 3.7. 15 students \((83.33\%)\) responded with completed questionnaires.
<table>
<thead>
<tr>
<th>No</th>
<th>Question</th>
<th>μ</th>
<th>σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>I think that I would like to use OpenSim frequently</td>
<td>2.00</td>
<td>0.93</td>
</tr>
<tr>
<td>Q2</td>
<td>I found OpenSim unnecessarily complex</td>
<td>3.27</td>
<td>1.22</td>
</tr>
<tr>
<td>Q3</td>
<td>I would like to use OpenSim for meetings with other students and staff</td>
<td>1.93</td>
<td>0.70</td>
</tr>
<tr>
<td>Q4</td>
<td>I think that I would need the support of a technical person to be able to use OpenSim</td>
<td>2.33</td>
<td>1.11</td>
</tr>
<tr>
<td>Q5</td>
<td>I found the content in OpenSim interesting</td>
<td>2.40</td>
<td>1.12</td>
</tr>
<tr>
<td>Q6</td>
<td>I think that OpenSim would be unsuitable for assessed coursework</td>
<td>3.20</td>
<td>0.94</td>
</tr>
<tr>
<td>Q7</td>
<td>I would imagine that most people would learn to use OpenSim very quickly</td>
<td>2.27</td>
<td>0.80</td>
</tr>
<tr>
<td>Q8</td>
<td>I found OpenSim disorientating</td>
<td>3.33</td>
<td>1.05</td>
</tr>
<tr>
<td>Q9</td>
<td>I felt very confident using OpenSim</td>
<td>2.73</td>
<td>1.03</td>
</tr>
<tr>
<td>Q10</td>
<td>I needed to learn a lot of things before I could get going with OpenSim</td>
<td>3.07</td>
<td>1.10</td>
</tr>
</tbody>
</table>

Table 3.7: Modified SUS Questionnaire items and relevant mean and std. dev.

Statements Q1 and Q3 were included to evaluate students’ willingness to use OpenSim. This helped us to decide whether the learning environment was attractive to students and whether they were willing to explore more usage in the future. The opposite statements were Q8 and Q6 respectively. Statement Q6 was of particular importance as it helped to evaluate the student perception of the specific area of the course assessment. Assessment methods have to be acceptable to students to be effective [181]. 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 of OpenSim with other courses, questions such as those outlined above would indicate how feasible that is. Statements Q5, Q7 and Q9 were included with the objective of assessing student difficulties with respect to using OpenSim. 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 Q5, Q7 and Q9 were paired against statements Q2, Q4 and Q8 respectively.

The resulting usability value was 40.3, which indicated that improvements were needed in the system usability aspects of these OpenSim based learning activities. The research reported in this thesis shows how appropriate user guidance and training for lecturers and students can help achieve this. The responses for the questionnaire statements Q1 and Q3 indicate that students’ willingness to use OpenSim is below the average norm (3) of the scale. The opposite statements, Q8 and Q6 respectively, show above average values. As described previously, Statement Q6 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 Q5, Q7 and Q9 were below the average norm (3) indicating students were more towards to disagree with the statements. Even though the statements Q2, Q4 and Q8 are the counter statements respectively, only Q2 and Q8 showed agreeing response mean values. The statement Q4 had an anomalous mean value; the main reason for this was that the students had 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.

Discussions with the students also helped to see their view on using OpenSim as an educational medium. Apart from the instability of the OpenSim version, the students were quite happy to explore it further as a tool for their learning.

3.8 Concluding remarks

This chapter presented an overview of SL and OpenSim system models and the findings of user studies on student learning activities in these two MUVEs. Identified system models and complex functional behaviours of SL and OpenSim indicated a need for user support educating all types of users involved in using these environments for learning. Moreover, user studies indicated the significance of having a usable MUVE for educational activities. Not only do the complexities in these MUVE system behaviours give users a steep learning curve, but also introduce a challenging task to the academics to fit those into the institutionally established formal educational practices while preserving high usability. The next chapter presents the research findings and user support models for managed learning with SL and OpenSim.
4. User Support Models for Managing a MULE

4.1 Introduction

This chapter presents a set of user support models for managing a MULE in the context of blended learning. The chapter starts with a model of blended learning involving MULEs and proceeds with a set of case studies examining possible blended usage. Section 4.3 presents a study carried out to identify important management considerations for OpenSim-based MULEs. Section 4.4 presents a taxonomy developed to describe the management considerations for different learning situations. Section 4.5 suggests a process model for helping users to follow policy-based management for MULEs; Section 4.6 concludes the chapter.

4.2 A Model for Blended Learning with MULEs

The continuing change and advancement of immersive technologies necessitates an ongoing reshaping of learning strategies [182]. Jung and Latchem [183] endorse this view by highlighting the need for new theories and models for blended education with the advent of new digital technologies. According to Voos [184], blended models will continue to expand the domain of their constituent technologies as these are superseded by new developments.

With respect to blending MUVEs with existing learning systems at a technical level this study recommends a loosely coupled learning environment arranged to provide flexibility and useful management options.

4.2.1 The Model – Strategic View

The opportunities for using technology enhanced learning alone are far fewer than having it integrated into blended learning scenarios [62] which have a mix of technologies that give complementary opportunities to the end user. This mix of technologies and learning tasks, often referred as the appropriate mix, is one of the main rationales for this model’s development. Furthermore, by appropriate mixing of technologies, students can obtain increased connectivity among their peers and engagement with teaching staff [185]. Brenton suggests that rich collaboration and user friendliness are expected norms on multiple platforms in blended learning [186]. Also, the unique learning affordances of MUVEs [66] can add value to the blended experience.

The model shown in Fig. 4.1 is suggested as a high level user support model for academics wishing to explore the potential ways of using MUVEs in their teaching and learning.
There can of course be scenarios that provide completely adequate and successful teaching and learning experiences with only one type of environment (e.g. classroom only or e-Learning only) or a combination of types (e.g. classroom and e-Learning together only). It depends on the required learning experience and the fitness for that purpose. Therefore, this model can help academics to examine possible scenarios of MUVE usage according to their requirements. It is important to reflect the MUVE learning affordances given by Dalgarno and Lee [66] (see Chapter 2) to examine the advantages of different usage scenarios of blended MULEs. The five affordances suggested by them indicate unique advantages one can obtain by using MUVEs for learning, which may be difficult to achieve from other approaches.

If academics wish to incorporate MUVEs in a blended manner it may be effective to have all three environments combined in a complementary manner by selecting the best learning affordances each environment can offer. However, it is also important to avoid learning affordances and practices being made redundant by overlap or replication.

In the case of blending only a single environment with MUVEs, traditional learning can benefit by using MUVE learning affordances. For example, traditional learning practices can take the lead on course delivery and assessment while MUVE based learning activities can play a support role. Since there is no e-Learning component, maintaining data consistency between the two environments can be minimal.

The combination between MUVEs and e-Learning without classroom participation can be useful in cases where MUVEs help to overcome missing traditional learning affordances. For example, distance education can benefit from immersive presence [187] and collaboration opportunities.
As the scope of this research is to provide suitable user support for MULE management, a particular interest is how the selected MUVEs can fit with existing (e) learning infrastructure. Since, at present, e-Learning solutions have proven to be effective when combined with traditional learning methods, an analysis of use cases in the context of existing blended learning environments is presented in the next section.

### 4.2.2 An Analysis of Use Cases for a Blended MULE

Since OpenSim/SL were not specifically designed to cater for educational needs, there are no dedicated user roles, use cases or process flows that can be taken as learning activities, unlike e-Learning systems such as Moodle. The goal of this section is to compare OpenSim/SL use cases with Moodle use cases in order to uncover the challenges that can be encountered when managing a MULE in a blended situation.

Banerjee [188] has presented a high level analysis of learning environment activities using UML use cases. OpenSim/SL use cases remain unchanged from one application domain to another, because those are the only possible ways of interacting with the 3D environment. In that respect, use cases are a reliable and accurate approach to analyse OpenSim/SL with an e-Learning system. The lack of an analytical understanding of appropriate use cases for a given learning environment can create challenges for blending. Furthermore, it can result in academics deviating from the MUVE learning affordances defined in [66].

The default user roles in Moodle follow a close correspondence with generic education institution needs for course management. It defines six user roles: Administrator, Course Creator, Teacher, Non-Editing Teacher, Student and Guest[50]. These Moodle roles are designed and aligned with the education institutional roles for convenient blending with traditional learning processes. In contrast, as we have discussed in Chapter 3, OpenSim/SL user roles and access rights are based on virtual land ownership. When we use OpenSim/SL within a formal educational setup that requires a certain degree of blending with Moodle we have to use these available OpenSim/SL roles appropriately, being careful to avoid conflicts.

The system architecture of OpenSim/SL ensures all user categories have the ability to manipulate their avatar actions and other properties within the 3D environment. Therefore, the common generic user role can be taken as 'resident user'; i.e., any registered user with the system who has access to the simulated environment. This common nature of the user role results in a poor granularity for defining learning use cases when compared with generic e-Learning systems. Considering the role attributes and available use cases, the default role resident user is taken as the most appropriate role for the student role in Moodle. Use cases of the resident user are compared with the Student role (Moodle 2.0) in Fig. 4.2.

With system support for rich text based content management and integration, Moodle can incorporate a diverse set of student activities as shown in Fig. 4.2. Student activities and interactions with Moodle can be categorised into 4 major types (shown in the use case diagram). The specialised learning activities such as Assignments, Forum, Choice, etc. are unique methods to support different learning activities. A particular attribute of these learning activities is that they have a close association with traditional...
learning practices of a similar nature; i.e., most of the tasks are predominantly text based and expect students to display a reasonable level of comprehension in communication as part of their assessments.

Figure 4.2: The Comparison of the key use cases of the student role in Moodle with available OpenSim/SL use cases

On the other hand, OpenSim/SL user activities display a more abstract and generic nature and their emphases are on 3D simulation, collaboration and immersion rather than advanced textual features. These user actions occur in a fairly intuitive manner, which gives the user a higher level of perceived presence compared to web-based e-Learning. The three main user actions available for the generic student role are not necessarily specialised for supporting learning, as shown in the diagram. Furthermore, careful observation of the available roles and use cases shows that the non-editing teacher
role in Moodle also corresponds to the generic resident user role in OpenSim/SL. Therefore, the OpenSim/SL use cases available for students indicate a primitive and abstract nature compared to dedicated systems for teaching and learning, and can be challenging when managing blended learning in OpenSim/SL.

The other main role investigated was the teacher (with editing privileges). Among the OpenSim/SL roles, the most suitable roles to represent teachers are estate manager or land owner. Here, the role estate manager has the functions to manage the estates in OpenSim/SL, as a whole or with different settings for selected regions within the estate. The role land owner indicates the user who owns regions or land parcels in a given region. The important factor is that having the ownership of land gives that user the ability to manipulate the 3D environment with a certain level of content management and avatar activity management. Estate owners were not identified as a direct corresponding role type for the teacher role in Moodle, although the estate owner role has a larger set of functions including all the functions that land owners or estate managers have. Further, even with the institutional arrangements, there is a clear separation of roles between the module coordinator and the teachers for a given module. By taking estate manager and land owner roles as corresponding to the teacher role in Moodle as well as in the traditional setups, it can be helpful to manage Moodle and OpenSim/SL based blended learning.

In Moodle, although teachers and students use the same set of extended use cases to participate in learning activities, such as Forum, Choice, Assignment, etc., for specialised learning activities, due to the system design models, teachers always have a higher level of access rights and privileges within these activities. For example, unless authorised, a student may not be able to observe peer work but the teacher observes all the actions of the entire student population of that course. Moreover, teachers have the ability to decide which learning activities are included in the module and how they are used within the context of Intended Learning Outcomes (ILO). The students do not get the right to modify these already defined learning tasks, but participate in them according to the defined time (submission deadline, task duration) and behavioural (single or group activities, submission format, etc.) constraints. It is a challenge to achieve this nature of role privileges within OpenSim/SL. Figure 4.3 compares the teacher roles in Moodle with OpenSim/SL and depicts the similar nature of structural mismatches one could encounter when integrating OpenSim/SL with Moodle.
The non-editing teacher in Moodle is quite helpful when it comes to map teaching support roles such as tutors and demonstrators. External observers for quality assurance and module evaluation can also be assigned to this role for passive engagement in the module activities. When it comes to OpenSim/SL, any user role without land ownership becomes a generic resident user and we are not able to differentiate role privileges other than by setting explicit management policies for particular students and teachers. In contrast, if the students are given land ownership at parcel level, they are then in the
same category as land owners including certain types of land management functions similar to those that teachers have, on their owned land.

If we compare the use cases shown in Fig. 4.2 and Fig. 4.3, we can see that only the land management ability as the land owner differentiates the tasks of a teacher from students in OpenSim/SL. This usage of access control based on land ownership to maintain the difference between student and teacher privileges may not sound rational since these systems are from two different domains; however, these are the only possible methods to separate student and teacher roles in OpenSim/SL. What if the students have to be given land ownership to explore further learning opportunities? For example, in the LAVA project [189], students follow virtual excavation tasks on their given land; in HCI projects [88, 98] students were assigned individual land parcels of equal size; with these ownerships, it would be similar to having the teacher privileges for a course in Moodle. How this scenario would fit into the existing learning practices and the institutional policies is a matter of uncertainty that depends on user attitudes and behaviour. This work aims to provide user support to overcome this challenge.

Figure 4.4 shows a comparison of the use cases of the course creator role in Moodle with the estate owner role in OpenSim/SL. This role is important to this analysis as both Moodle and traditional learning arrangements have closely corresponding roles, course creator and module coordinator, respectively. In OpenSim/SL, the closest matching user role to these is the estate owner. Other than the usual teaching activities and land management as shown in Fig. 4.4, estate owners can assign teachers as estate managers. This corresponds to the Moodle course creator ability of assigning teachers to the module. However, the most important difference between these two roles is that estate managers cannot create regions or estates on their own; they need administrator support for server commands. In a large scale teaching environment, delegation of course creation to module coordinators helps system administration to be separated from learning management, which also gives finer levels of control.

System administration use cases in the administrator role are not considered here as quite often these tasks are separated from the academic and teaching support staff. Universities usually have a separate unit to provide IT services and system administration of e-Learning systems, among other system support tasks, is considered as an institution wide responsibility of these staff. When integrated for blended MULEs, system administration of OpenSim/SL is supposed to be a task of the system support staff. If SL is used as the MUVE, none of the system administration tasks are available for the educational institute, as Linden Labs keep the exclusive ownership and access rights of their server environment. Because of this reason, this research seeks to provide user support for managing at the application level in OpenSim/SL.
The detailed analysis of the use cases presented above supports the view that integrated OpenSim/SL can provide complementary functions but cannot replace existing types of practice distinctive to e-Learning and traditional learning; in fact, it reiterates the MUVE learning affordances [66] and suggests suitable considerations when design blended learning with OpenSim/SL. Moreover, inappropriate use case integration between e-Learning systems and OpenSim/SL can result in inconsistent data and difficult management challenges; this can be reduced with the user support models and tools this work proposes.
4.2.3 Use Case Mapping Challenges – A Case Study

The previous section elaborated on the importance of accurately mapping use cases and roles between OpenSim/SL and Moodle for successful managed learning in a blended MULE. However, it is understandable that for effective and widespread use of the proposed ideas, there should be a further level of concrete analysis of deployed systems where users can try to practice managed learning in a blended MULE. For that, a case study on Sloodle [190], an open source solution to mediate learning between Moodle and OpenSim/SL was carried out. For this case study, a prototype learning environment that meets the proposed blended MULE expectations was installed and tested. Moodle (version 1.99) as the e-Learning platform, Sloodle (version 1.0) and OpenSim/SL (version 0.7.2) as the MUVE were used. All these systems are free and open source.

With reference to the MUVE learning affordances given by Dalgarno and Lee [66] we can identify five of these that are suitable for use in a blended learning environment: learning activities that require increased spatial domain knowledge, experiential learning opportunities, intrinsic motivation and engagement, improved knowledge transfer and high collaboration [66]. These types of activities can be better supported in the MUVE instead of e-Learning or traditional environments. This analysis is performed with the basis of that view.

Before examining individual technical implementation offers of Sloodle, it is useful to discuss the underlying system integration model it follows; in fact, this model has resulted in the development of Sloodle Tools, which are further discussed later. The architectural goal of Sloodle is to mediate learning activities between Moodle and OpenSim/SL [190]. The way this goal is achieved is interesting; Sloodle takes Moodle tasks (Assignment, Quiz etc.) as the basis for their system design and develops a corresponding 3D content/functionality to represent that behaviour inside OpenSim/SL. The level of Moodle dependence is very high such that Sloodle is executed as a plug-in of Moodle (php scripted web server pages). This tight dependence results in certain Sloodle tools compromising their learning functionality when compared with the learning affordances of OpenSim/SL and Moodle independently. The observed functionalities are briefly described below while examining the challenges for a blended MULE with Moodle and OpenSim/SL.

Sloodle learning functions include 11 activities to map selected Moodle activities such as chat, forum, glossary, choice, content display and limited support of quiz and 3D content submissions for assignments. These functions are available as deployable tools (scripted composite prim objects) inside the 3D environment. A summary of Sloodle tools along with the corresponding Moodle task are listed in Table 4.1.
<table>
<thead>
<tr>
<th>Sloodle Tool</th>
<th>Function</th>
<th>Moodle Correspondence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Webintercom</td>
<td>Synchronise Chat messages between Moodle and 3D MUVE</td>
<td>Chat</td>
</tr>
<tr>
<td>Presenter</td>
<td>Display media files</td>
<td>Content Display</td>
</tr>
<tr>
<td>Toolbar</td>
<td>Blog, Gestures and Avatar list display</td>
<td>Forum</td>
</tr>
<tr>
<td>Quiz chair</td>
<td>MCQ question support</td>
<td>Quiz</td>
</tr>
<tr>
<td>Pile on quiz</td>
<td>MCQ question support</td>
<td>Quiz</td>
</tr>
<tr>
<td>Prim drop</td>
<td>Submit prim for assignment in Moodle</td>
<td>Assignment</td>
</tr>
<tr>
<td>MetaGloss</td>
<td>Access Glossary in Moodle</td>
<td>Glossary</td>
</tr>
<tr>
<td>Sloodle Choice</td>
<td>Synchronise Choice between Moodle and 3D MUVE</td>
<td>Choice</td>
</tr>
<tr>
<td>Vending machine</td>
<td>Distributed content objects for 3D MUVE (OS)</td>
<td>--</td>
</tr>
<tr>
<td>Awardsystem</td>
<td>Connects Sloodle points with Moodle Gradebook</td>
<td>Gradebook</td>
</tr>
<tr>
<td>Picture Gloss</td>
<td>Connects textures from Moodle Glossary to a prim object</td>
<td>Glossary</td>
</tr>
</tbody>
</table>

Table 4.1: Sloodle tools and corresponding Moodle functions

Webintercom and Presenter provide important features for combined learning activities in Moodle and OpenSim/SL; synchronised user communications between the two environments, i.e., chat communication in OpenSim with chat communication in Moodle, can be useful for uninterrupted communication. This is particularly important if the students are expected to participate in a collaborative learning session with high interactivity. The published content for learning in Moodle can also be displayed in OpenSim using Presenter, which is a useful function for maintaining the consistency of learning activities and content; however with the advanced Media on Prim functionality of OpenSim Presenter may not always be required for content display.

However, using OpenSim chat channels to publish student compositions in Moodle tools such as Forum, Glossary and Wiki, is questionable as those entries are supposed to feature rich text and content. Usually, students and teachers tend to replace the medium and the submission methods of the traditional form of essays and comprehension tasks to e-Learning system, by using these generic e-Learning tools that are specifically designed to align with the generic transferrable skills of writing and comprehension. Moreover, these tools are generic to be used in a range of different subjects as they also allow students to express their attained core skills from the module by incorporating rich text or images to enhance their answers. These activities cannot be supported through OpenSim chat or IM channels due to their limitations of message size and text formatting, at present.
Furthermore, asking students to participate in quizzes, assignments and text based assessment activities in OpenSim instead of Moodle can introduce further difficulties for student work. Except for questions that require short and specific answers, quite often, in written answers there can be a grading component that corresponds to the student’s ability to present the answer. This is fairly common when it comes to structured or essay type questions. Sloodle tool support for Moodle Quiz and Assignment tools is limited to Multiple Choice Questions (MCQ) and prim submissions; when it comes to complex text based assessment activities, students are often required to re-login to Moodle after of their initial submission, to enrich the entries that have been done through raw chat interfaces while they were inside OpenSim. Accessing Moodle through an in-world media browser directly would allow a certain level of rich text support as shown in Figure 4.5. Accessing Moodle directly in web format can be important for formal assessment and feedback to prevent teachers introducing unproductive use cases that do not match the learning affordances of MUVEs.

![Figure 4.5](image.png)

**Figure 4.5:** Creating a rich text entry for a Moodle course using the in-world web browser without Sloodle mediation (direct access to Moodle through URL)

Another important fact to consider is the accurate map of user collaboration facilities provided in MUVEs with the e-Learning environment. Specially, web based learning activities by their nature are not designed for rich collaboration in real time, but a user session based individual browsing, asynchronously. The efforts to use collaborative web browsing techniques with Moodle have not been successful in attracting a wide user community to date, due to the complex system properties and the implementation challenges. A collaborative web browsing (‘co-browsing’) tool is being developed as an extension to the Sloodle system [191]. However, it is too early to evaluate such a tool for successful and complete web browsing for serious collaborative learning activities.
Additionally, the recent add-on functionality of Media on Prim lets the students benefit from a range of media data formats such as HTTP documents and video streams, without specific tool support from Sloodle. Although this functionality can support student activities in the e-Learning system while they are inside the virtual world, due to its open nature Media on Prim can compromise the privacy and confidentiality of a student’s work if other students are also present, nearby. This is a new challenge which was not present with in-world web browser or external web browsers as any interaction with those browsers is private with respect to other users. In particular, assignments, quiz activities and forum posts that are graded need to be hidden from the other students. Moodle supports this feature with a strong access control and user privileges. Therefore, a clear understanding of which learning tasks / resources should be displayed in-world as part of the blended MULE, considering the confidentiality needs of such tasks or resources, is necessary to maintain the same level of security of those as in Moodle.

Prim Drop and Vending Machine are important tools for mediating the 3D content ownerships, their associations and avatar identity registration with Moodle course modules. Vending Machine does not correspond to a particular Moodle activity but it ensures distribution of the other Sloodle tools for learning so that teachers can comfortably setup their preferred learning tasks in-world. Because of this approach, even a teacher with a limited experience of integrating OpenSim with Moodle can confidently design and practice blended learning in a MULE. Prim Drop facilitates the assessment tasks in particular, as teachers can publish an assignment in Moodle as part of summative assessment, asking students to develop 3D content to be evaluated. Effectively, there is no seamlessly integrated mechanism to update the Moodle Assignment parameters such as submission time, number of attempts, etc. other than the Prim Drop tool.

Finally, the Sloodle tools Registration Booth, Access Checker and Login Zone that provide the functions required for platform integration with accurate identity and access control mapping between OpenSim and Moodle, were investigated. In fact, these tools are quite important for successful data and user consistency in blended MULE solutions. Although, they are not related to a particular set of learning use cases, they ensure uninterrupted teaching and learning sessions at the system administration level; hence, facilitate system administrators of Moodle and OpenSim to automate most of the role and data mappings.

Therefore, in summary, trying to achieve all learning use cases of an e-Learning system in MUVEs as part of the proposed blended MULEs is not advised for serious learning scenarios. In the case study, it was identified that textual and document based learning content and activities that require complex text comprehension with rich data formats are not yet fully supported in OpenSim/SL compared to Moodle. As a result, academics that wish to use and explore blended MULEs should rationally decide on appropriate use case implementations according to the learning affordances of the domain and the requirements of their teaching and learner support. Moreover, students should be encouraged to explore learning activities according to the learning affordances of the environment. Essentially, we should avoid forceful integration of ineffective learning use cases between the environments for the mere reason of blending.
4.2.4 A Case Study on a Learning Aid within Different Domains

This study examines the different design and delivery options of learning content for blended MULE (in OpenSim/SL). It is not enough just to deliver the old content in a new medium; we must seriously reflect on how to design and deliver education according to the new medium [67]. Also Voos has indicated that "It is likely not the ‘blendedness’ that makes the difference, but rather the fundamental reconsideration of the content in light of new instructional and media choices" [184]. These views from related work suggested conducting a further study, examining the nature of educational content when MUVEs get integrated with the existing e-Learning environments. Therefore, a case study comparing the two versions of the same learning aid, one as a web based aid and the other with OpenSim was carried out.

For this purpose, two distinct implementations of the link-state Internet routing protocol OSPF (Open Shortest Path First) [192] simulations as learning aids were selected. OSPF uses Dijkstra’s Shortest Path algorithm [193] as its mechanism to build and calculate the distances to all known destinations. Importantly, understanding Dijkstra’s Shortest Path algorithm can be a challenging task for an unaided student due to its complex nature. Like many algorithms, it can leave some students bored and disengaged, therefore, demonstrating a user-friendly learning resource to help students who would otherwise lose interest has always been a demand in computer science education.

The first attempt was to implement a web-based learning resource with animation support. This implementation also had a certain level of interactivity with feedback to self-assess user’s understanding of the algorithm behaviour. A working instance of this learning aid demonstrating Dijkstra’s algorithm is shown in Figure 4.6 [194].

![Diagram of OSPF algorithm simulation](image)

**Figure 4.6:** An operational instance of the interactive web based OSPF algorithm simulation learning resource
Although this Web based OSPF learning resource was a useful complement to the static textual explanations of OSPF in textbooks and lecture materials, several obstacles to student learning were identified associated with its domain limitations. The learning affordances a MUVE can provide as defined in [66] were missing in this aid, i.e., the difficulty of depicting 3D simulations, poor support for real time multiuser collaboration, poor support for experiential learning and lack of user immersion were observed. However, with the intrinsic support for textual content of the platform, this version of OSPF animation provides useful text based learning content and dynamic activities such as formatted routing table, node – arc summary are displayed at each step and textual user input and feedback are supported for self-assessments. It also provides links to additional external resources external for an enthusiastic student to refer to for further study. A possible disadvantage on student learning with this learning aid can be students may tend to visualise the network topologies and algorithmic behaviour believing that the possible networks are limited to the 2D arrangements, although such view is not theoretically correct. If students are affected by this limited support for correctly grasping the underlying subject matter, they may only be able to demonstrate the first three levels of the SOLO taxonomy [103] in an assessment.

Accordingly, another learning resource for simulating OSPF behaviour has been developed in a locally installed OpenSim island [94]. The web-based OSPF learning resource demonstration has been replicated with the 3D content and scripting for the required animation; however, it is only a small part as an introduction to the island, when compared to the extended scale and broader subject matter it presents. Therefore, further extensions and engaging learning use cases have been associated with this MUVE learning aid.

An important advantage with this implementation is the extensive facilitation for collaborative group activities. Certainly, it is one of the MUVE learning affordances; as emphasised throughout, learning resources should be carefully designed to exploit it, however. Student collaboration on network topology creation, setting various weights and resources, and observing and commenting on others’ use of the learning aid are new use cases that have been introduced compared to the 2D web aid. In fact, these use cases of learner collaboration are essential for a broader understanding of a complex concept, which would otherwise be limited to individual insights. User collaboration occurs through interaction between learners while interacting with the MUVE, through the learning interface that provides support rather than barriers to learning [78]. Students have been given the opportunity to use OpenSim communication channels (chat, voice, IM) to share their ideas while engaged in their learning. The 2D web version could have been augmented by chat and/or vice, and could have allowed for student defined topologies, and could even allow for multiple concurrent presence, but the end result would have been highly complex and unwieldy.

Students can easily practice the highest level of SOLO, Extended Abstract responses; an example of what can be achieved in Extended Abstract level learning engagement is shown in Fig. 4.7. It shows a hypothetical Internet core as a hypercube of degree 4, which has been collaboratively constructed; a difficult concept to draw or visualise in 2D. These capabilities are an excellent opportunity for students to reflect on what they have learnt and to collaboratively overcome individual concerns on liminal spaces related to a threshold concept of their learning [195].
The main emphasis given in this OSPF simulation implementation was to depict the associated concepts in 3D visuals as much as possible instead of the textual alternatives. Although experiential learning opportunities based on simulated 3D worlds can be implemented in other ways, without using a MULE, users are likely to feel that the environment is too contrived and not rewarding in terms of immersive learning [78]. Therefore, a reasonable amount of textual content and activities have been appropriately redesigned and mapped to 3D content to explore the MUVE learning affordances. By doing so, however, certain learning use cases that are primarily associated with textual content cannot be satisfactorily mapped on to 3D content. For example, the use of MUVE raw text channels to display routing information was one of the options considered. But lengthy textual descriptions of state updates and routing information are neither aesthetic nor easily readable as they scroll out of view. Indeed, this strengthens our argument for using complementary systems for blended learning. An appropriately designed web-based e-Learning resource can provide rich textual content and complements MUVEs if there is sufficient integration. Moreover, as discussed in the previous section, such a resource could be accessed interactively through in-world or external browsers, while participating in MULE based learning activities.

![Figure 4.7: A collaborative activity in the Routing Island – the construction of hypothetical Internet core as a hypercube of degree 4](image)

Another challenge with a MULE learning content is the provision of external resources of the same calibre for further studies. Being a novel and growing technology for learning, MUVEs possess a reasonable amount of 3D learning resources to be referred to for a particular area of study. Even if there are 3D contents available from external sources, accessing and execution of those content can be challenging due to the heterogeneous content ownership management policies and platform constraints. An innovative solution may be to display video and interactive content from an external source using the media display-on-prim functionality in OpenSim, which can help to share learning resources in MUVEs.
This case study provided a further view on using MUVEs with existing learning infrastructure. Importantly, it reiterated the value of considering learning affordances of each environment can offer and select the best fit for the learning need; moreover, learning content development should also follow the same approach in which MUVE content and e-Learning content should complement each other for productive blended learning with 3D support.

4.3 Important Management Aspects for a MULE

A study for identifying important management aspects for a MULE was carried out using an OpenSim based learning session. The research overview, experiment design and data analysis are discussed in the following sections.

4.3.1 Research Model

Student behaviour and system administration have been widely considered as important aspects of effective learning environment management. Based on the research findings from OpenSim/SL use cases and role analysis and the function behaviours [12] that were discussed previously, these two factors were hypothesised to be important for managing MULEs.

Previous studies on student engagement with the learning environments (traditional and e-Learning) has also defined self-regulation (cognitive) and positive conduct (behavioural; here in MUVEs, system management shapes the users’ positive conduct in a constructive and civilised manner) as the main factors [196]. Student engagement includes two dimensions: self-regulated learning and the level of disruptive behaviour in the class (here in MUVEs, system management controls are indeed used to address the disruptive behaviours) [197]. Importantly, engagement with the MULE may not necessarily represent student engagement with learning, although there can be a positive correlation if the learning tasks are in constructive alignment [108]. However, the opposite; i.e., if student engagement with the MULE is low, so it is with the learning tasks that depend on the MULE, obviously. If students do not engage with the MULE, there is a high likelihood that they generally have low engagement with their other learning activities as well. Hence, the management practices used in a MULE should not negatively affect student engagement.

Schunk [152] has suggested that there is a need for more research aimed at improving students' self-regulatory skills as they are engaged in learning and to examine how learning environment contexts affect the amount and type of self-regulation displayed. Importantly, in contrast to games, MUVEs may require higher level of user self-regulation, which can be an important factor for managed learning with MULEs. As discussed in OpenSim/SL system studies environment management was also considered as a factor to examine.
4.3.2 Experiment Design

It was planned to examine these three factors: self-regulation, environment (system) management and student engagement with the environment as the research variables. A two phase study was carried out: first, students from two course modules visited a MUVE supported learning environment (in OpenSim) as part of their studies. It was important for the usefulness of the data that the students participated in an actual credit bearing learning session rather than following an artificial task. Further, it helped students to consciously reflect on the experience they had. The two modules have different learning objectives levels in the Scottish Credits and Qualification Framework (SCQF) [198] (Table 4.2). This study focused on student engagement with the environment; although students had different levels of the same learning task, both samples were similar with respect to the study measures. The differences in the modules did not affect the experiment; hence, measurements were considered as a single sample consisting of 59 students for the data analysis.

The learning environment, Wireless Island [87], is a dedicated region for wireless communication education. It provides interactive simulations with various configuration settings for students to explore and learn. It also includes supplementary learning content such as lecture notes, lecture media streams and a museum of the history of wireless communication.

<table>
<thead>
<tr>
<th>Setup</th>
<th>Module Associated</th>
<th>SCQF level</th>
<th>Number of students</th>
<th>Brief Description of the Learning Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CS3102 Data communication and Networks</td>
<td>10 (Junior Honours)</td>
<td>31</td>
<td>Interact with Wireless Island and observe Wireless traffic simulations with different settings. Be able to explain basic conditions and problems related to Wireless communication.</td>
</tr>
<tr>
<td>2</td>
<td>CS5021 Advanced Networks and Distributed Systems</td>
<td>11 (MSc)</td>
<td>28</td>
<td>Interact with Wireless Island and observe Wireless traffic simulations with different settings. Be able to explain basic conditions and problems related to Wireless communication.</td>
</tr>
</tbody>
</table>

Table 4.2: The summary of experiment groups and module details

![Figure 4.8: Wireless Island overview and a simulation](image)
In Fig. 4.8, the left side image shows the island and the right side shows an interactive learning aid for simulating the Exposed Node problem in wireless communication.

It was decided to have 6 students per region in order to spread the system load. Therefore, five regions were created in OpenSim with Wireless Island on each. This resulted in an identical learning set up for each student, and students were given their assigned region as their home to start the learning task. The corresponding OpenSim region map and an island map are shown in Figure 4.9. The root island (Learn) was an empty island which could be used as a sandbox. A square block represents an island (256 m x 256 m virtual area), and the islands are distributed as shown on the map, to minimise adjacency problems and to simulate the isolated island look and feel. Tiny green dots indicate the student avatar positions when the image was taken.

![Figure 4.9: The map of experiment regions: left - the world map, right - a region](image)

The second part of the study, i.e., the data gathering, was based on a questionnaire with 15 questions divided into two sections: Avatar Engagement – 7 questions, and MULE Management – 8 questions. Additionally, five open-ended questions were included to help students to express their opinions.

Students who were more engaged with the 3D learning environment tended to experience greater learning gains, regardless of prior knowledge [199]. Although there are previously used instruments to measure student engagement, all of those are mainly based on school (traditional academic environment) properties. Finn's measure of school engagement was more comprehensive than other constructs [197]. It is based on available activities in a classroom for students. Students’ situational interests and their sense of presence are contributors to their engagement with the learning environment [199]. Considering the previous studies and the unique nature of MUVEs it was decided to develop a set of questions that characteristically represents MUVEs and examines key features of possible engagements. Importantly, as the related MUVE studies have used, generic activities that any user can perform in-world irrespective of their roles or privileges were focused. Questions Q1- Q6 in the user engagement section of the questionnaire represented 6 prime activities that any avatar could follow to engage with the environment. Content creation, environment interactions and collaborative presence are factors for virtual world engagements [85]. Q1- Q6 questions cover these, and Q7 directly asks the
student for their opinion of their engagement with the environment. A further analysis on the question scores will be discussed later.

The 8 questions in the MULE management section had relevance for the two factors of the thesis: self-regulation and MUVE system management; however, the questions did not directly represent the variables. It was decided to confirm through statistical analysis, therefore, treated as 8 related questions, as shown in Table 4.3. Similar to the first 7 questions, these 8 questions were also defined with respect to the MUVE context considering the system functions and avatar interactions on learning.

<table>
<thead>
<tr>
<th>No</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Section 1 – MUVE User Engagement</strong></td>
</tr>
<tr>
<td>Q1</td>
<td>I changed my appearance as I like to appear</td>
</tr>
<tr>
<td>Q2</td>
<td>I created content objects in the environment</td>
</tr>
<tr>
<td>Q3</td>
<td>I tried to change land or content objects in the learning environment</td>
</tr>
<tr>
<td>Q4</td>
<td>I communicated with others regularly</td>
</tr>
<tr>
<td>Q5</td>
<td>I have followed other avatars collaboratively during my learning</td>
</tr>
<tr>
<td>Q6</td>
<td>I moved to all the places in my island and teleported to other islands</td>
</tr>
<tr>
<td>Q7</td>
<td>My activities in the environment resulted in a high engagement with my learning tasks</td>
</tr>
<tr>
<td></td>
<td><strong>Section 2 – MULE Management</strong></td>
</tr>
<tr>
<td>Q8</td>
<td>I think my behaviour affected others’ learning</td>
</tr>
<tr>
<td>Q9</td>
<td>The open space and others avatars allowed me to interact as in a real-world learning session</td>
</tr>
<tr>
<td>Q10</td>
<td>Use of real identities increases proper behaviour of students</td>
</tr>
<tr>
<td>Q11</td>
<td>Land and Content management controls are important for environment management</td>
</tr>
<tr>
<td>Q12</td>
<td>System control and management practices are important for reliable learning</td>
</tr>
<tr>
<td>Q13</td>
<td>System management settings should not reduce the MUVE usability</td>
</tr>
<tr>
<td>Q14</td>
<td>Appropriate system security and controls ensure a successful learning experience</td>
</tr>
<tr>
<td>Q15</td>
<td>Students should responsibly use the learning environment</td>
</tr>
</tbody>
</table>

Table 4.3: Questionnaire items

### 4.3.3 Observed Student Engagements

This section presents observations on student behaviour during and at the end of the experiment. The main observations were from avatar appearance, content and land related student activities.

Avatar appearance changing can be fun, although creating attractive appearances within a limited time can be challenging. OpenSim used with the Hippo viewer (version 0.6.3) gives an additional step when changing user appearance compared to SL. Users first have to create a body-part, edit it and then wear it to change the shape of the default avatar. However, few students spent long enough to create more sophisticated shapes, clothes and even change the avatar gender (the default avatar shape, known as Ruth, is shown in the left most image of Fig. 4.10). Postgraduate students were less interested in changing their avatar, whereas many undergraduates went a further step by comparing the avatar appearances with their friends’. However, the impact of making avatars look good should not be underestimated as it can take substantial portions of the available learning time.
Figure 4.10: Varying levels of avatar customisations:

In Fig. 4.10: left inset – default avatar, second image – only clothes changed, third image – body shape and clothes changed, right inset – gender, body shapes and clothes changed.

Content object creation is one of the fundamental mechanisms available in MUVEs. As observed, students tried a range of constructions as well as editing of the existing objects. Some of these alterations directly affected the learning experience; activities such as wearing the control buttons of the media display, moving and changing the internal arrangement of the lecture theatre, and creating constructs on the simulation area (shown in Fig. 4.11), should have been discouraged through management policies for effective learning. However, in this exploratory study, it was planned to observe such actions and use them as evidence to inform future work; hence, students were given unrestricted access to their environments.

Figure 4.11: Student content creation and content alteration in the learning environment
Figure 4.12: A region compared at the end of the session: left – original layout and right - the terrain modified and content altered region

Some of the land alterations also affected the learning experience. Terrain changes can significantly affect learning. Learning aids and content objects in a MUVE are put in-world with specific positions; if students change the terrain shape and land height it can completely change the expected learning experience. In extreme cases, the learning content may have been buried in land preventing students even seeing it, let alone accessing and interacting with it. This is a form of Denial of Service (DoS) type attack when put into the MUVE context. These have to be discouraged through usable policies. A student interaction caused the learning environment to be significantly altered compared to its original layout (Fig. 4.12). This observation was a one-off incident, as the majority of students refrained from changing land settings. Compared to the postgraduate (Masters) students, the undergraduate (Honours) students showed high interactivity, resulted in a range of user-created objects, altered content and changed land terrain. The undergraduates were keen on exploring game-like features, and engaging their friends for collaborative activities, although those activities were not related to the learning. Students that were keen on completing their tasks may have had less motivation to explore the MUVE, however. Students were allowed to follow their preferred behaviour and environment interactions as a mean of learning through exploration [65] without any restriction. An assurance was given that their behaviour would not affect their grades, but that it would delay completion of the learning tasks.

In brief, as discussed above, there were number of incidents observed, which highlights the need of student self-regulation and system management. In addition to the above mentioned on land and content related activities following actions were observed: For example, streaming video lecture displays are reset whenever an avatar hits the play button, disturbing the other viewers. Also, in certain instances student’s simulation arrangement was too close to another’s setup, simulation interference was observed affecting the student learning. Some avatars wore learning content and moved randomly affecting others’ learning. These can be easily solved if suitable environment management practices are followed and students show self-regulation.
4.3.4 Data Analysis

32 completed questionnaires (54.28%) were received, which consisted of 20 responses from postgraduate students (71.4%) and 12 responses from undergraduate students (38.7%). A preliminary clustering of the questions into two categories: (Q1-Q7) user engagement and (Q8-Q15) MULE management, was made for further analysis.

The MULE management section was designed to examine the two prime variables associated with the study. Statistical analyses were done for accurate variable identification; Exploratory Factor Analysis (EFA) using PASW [version 18.0] was used to test the question set Q8-Q15. Pre-tests were conducted to examine the fitness of data to be used for EFA. The Bartlett Test of Sphericity gave $\chi^2 = 155.257$, $p<.001$ suggesting that the correlation matrix (R-Matrix) of items shows highly significant relationships and can be clustered based on relationships. Moreover, when all the question items per factor have consistently high loadings (>0.6) the sample size effect is negligible; a good recovery of factors can be achieved with samples of size is well below 100 [200]. Furthermore, the Null Hypothesis of R-Matrix being an Identity Matrix can be rejected with significance, indicating high suitability for EFA. Additionally, the Kaiser-Meyer-Olkin (KMO) test to examine the accuracy of using the data sample for the EFA; KMO value = 0.714 (>0.7; [201]) validated the fitness. The sample size to variable, $N:p$ was high and meets the guidelines given in [202].

Principal Component Analysis (PCA) was used to extract factors; two factors ($Eigenvalues>1.0$) was obtained. These highest two factors contribute nearly 72% of the total variation of the aspect MULE Management. Further, Orthogonal Verimax Rotation with Kaizer Normalization was employed and obtained the rotated factor loadings. PASW output is shown in Fig. 4.13.

To remove weak loadings 0.6 was used as the cut-off as suggested in [203] [201]; anything over 0.50 is generally classified as a “strong” item loading [204]. The Orthogonal Verimax Rotation seems accurate as the two factors relatively equally contribute (37.13% and 34.74%) to MULE Management. The Component Transformation Matrix showed symmetry over the diagonal, indicating the Orthogonal Rotation is accurate, and the rotated factor loadings are correct. As the rotated factor loadings indicated, questions Q9, Q15, Q10 & Q8 were considered as representing one variable. The objectives of the questions represent the student behaviour with Self-regulation. The second factor is represented by Q11, Q14, Q12 & Q13 questions relates to MUVE system environment management; this variable was named as Environment Management.
The identified two variables explain nearly 72% of the behaviour of the MULE Management. Moreover, there were not any other strong and significant variables revealed through the factor analysis; it indicates that the extracted two variables, Self-Regulation and Environment Management, contribute to MULE Management.

**Variable Summary**

**Student Engagement**

The questions associated with this variable were analysed for descriptive statistics. Table 4.4 summarises the analysis results. The One-Sample Kolmogorov-Smirnov test for normality indicated the question means are normally distributed \( [X \sim N (3.598, 0.283)] \) with significance (\( \alpha=0.997 \)); it implies that the descriptive statistics sufficiently represent the sample for analysis.

<table>
<thead>
<tr>
<th>No</th>
<th>Question</th>
<th>Mean</th>
<th>Mode</th>
<th>Std. Dev.</th>
<th>Std. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>I changed my appearance as I like to appear</td>
<td>3.09</td>
<td>3</td>
<td>0.466</td>
<td>.082</td>
</tr>
<tr>
<td>Q2</td>
<td>I created content objects in the environment</td>
<td>3.69</td>
<td>4</td>
<td>0.471</td>
<td>.083</td>
</tr>
<tr>
<td>Q3</td>
<td>I tried to change land or content objects in the learning environment</td>
<td>3.66</td>
<td>4</td>
<td>0.483</td>
<td>.085</td>
</tr>
<tr>
<td>Q4</td>
<td>I communicated with others regularly</td>
<td>3.53</td>
<td>4</td>
<td>0.507</td>
<td>.090</td>
</tr>
<tr>
<td>Q5</td>
<td>I have followed other avatars collaboratively during my learning</td>
<td>3.44</td>
<td>3</td>
<td>0.504</td>
<td>.089</td>
</tr>
</tbody>
</table>
The questions Q1 and Q5 (Q1 in particular) showed averages around 3 (Neither Agree nor Disagree). Q1 relates to the student engagement through the measure on the level of avatar appearance modification. In fact, few students tried to modify their avatars to look either similar with their physical appearance or to simulate a fantasy outlook. However, most of the students were comfortable using the default avatar or with minimum customisation, such as colour change of clothes, and committed their time more on other engagements and learning tasks, which resulted in this lower mean value. Q5 asks about collaboration, specifically on following the behaviour of others or learning steps. In this learning activity, the students were given individual task sheets that they should complete, but not specifically asked to collaborate with other students assigned into the same region. The majority of the students may have felt that they should not misuse the flexibility within the MULE, as their individual performances in the tasks were being assessed, so adopted a self-restrained approach to collaborating with others.

The questions Q2, Q3, Q4 and Q6 were based on the remaining avatar actions associated with the possible areas of avatar engagement. Content creation, land and content alteration, avatar communication and avatar exploration activities showed reasonably similar average responses. Additionally, the majority of the students confirmed that response (Mode = 4). Almost all the students indicated a high, positive average with most of the students agreeing with the statement.

Q7 played the concluding role for the student behaviour question set. It let the students reflect on their activities and then evaluate their learning engagement as a consequence of those. It directly inspects student engagement as a measure of student activity. Student responses showed a high average, indicating that they agreed with the question (Mode =4). This shows a possible relationship between environment interactions and learning engagement.

**Self-Regulation**

The One-Sample Kolmogorov-Smirnov test for normality indicated that the question means are normally distributed \( X \sim N (3.876, 0.378) \) with significance \( (\alpha=0.516) \); it implies that the descriptive statistics sufficiently represent the sample for analysis. Table 4.5 summarises the analysis results.
<table>
<thead>
<tr>
<th>No</th>
<th>Question</th>
<th>Mean</th>
<th>Mode</th>
<th>Std. Dev.</th>
<th>Std. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q8</td>
<td>I think my behaviour affected others’ learning</td>
<td>3.31</td>
<td>3</td>
<td>0.592</td>
<td>.104</td>
</tr>
<tr>
<td>Q9</td>
<td>The open space and others avatars allowed me to interact as in a real-world learning session</td>
<td>4.05</td>
<td>4</td>
<td>0.354</td>
<td>.064</td>
</tr>
<tr>
<td>Q10</td>
<td>Use of real identities increases proper behaviour of students</td>
<td>4.02</td>
<td>4</td>
<td>0.309</td>
<td>.056</td>
</tr>
<tr>
<td>Q15</td>
<td>Students should responsibly use the learning environment</td>
<td>4.10</td>
<td>4</td>
<td>0.296</td>
<td>.051</td>
</tr>
</tbody>
</table>

Table 4.5: Questions on student self-regulation and descriptive statistics

Q8 tends to be a self-assessing question as students had to think about their behaviour reflectively and critically. This was important to meet the objectives of the question set, as students answered the rest of the questions with a reflective mind associating every little detail that they have experienced or felt during the engagement. Students may have been doubtful about the degree they consider their behaviour and impact on others’ learning; therefore, an average of 3.31 (more towards the response “Neither Agree nor Disagree”) while the majority confirming that (mode = 3) was observed.

Questions Q9, Q10 and Q15 have more or less recorded nearly the same averages (~4 = Agree) while the majority confirms that preference as the Mode is 4 for each question. Q9 was designed associating the privacy concerns of being in an open environment that could be seen by others and with a high probability of simultaneous engagement in same learning activity or content. The association of the real-world classroom metaphor reinforces the student comparative observations, resulting in a broader opinion with higher accuracy. Q10 examines the student view on having their real identity (first name and last name) as their avatar username. Avatar anonymity and its impact on student learning has been researched previously in various contexts [177]; the majority of students agreed (Mode = 4 & Mean = 4.02) that there is a positive impact from using their real identities on practicing appropriate behaviour within the environment. Q15 plays the concluding role for the student’s self-regulatory practice with the sense of being a responsible participant in the learning session. The responses indicated the majority of the students agreed that they must use the environment responsibly; a positive indication of self-regulated interaction as an acceptable practice.

**Environment Management**

The One-Sample Komogorov-Smirnov Test for normality indicated the question means are normally distributed \( X \sim N(3.95, 0.175) \) with significance (\( \alpha=0.796 \)) it implies that the descriptive statistics sufficiently represent the sample for analysis. Table 4.6 summarises the analysis results.

These four questions are about student perception of system administration and environment control aspects. Q11 asks about two main OpenSim system administration functions. Land and content management related functions are the most significant activities that an avatar can use to affect the existing learning environment. It is important to examine students’ views on constraining them from using these features if needed for environment management. As the results indicate, students agreed with this statement.
Q11: Land and Content management controls are important for environment management.
Mean: 3.78
Mode: 4
Std. Dev.: 0.420
Std. Error: 0.074

Q12: System control and management practices are important for reliable learning.
Mean: 3.91
Mode: 4
Std. Dev.: 0.296
Std. Error: 0.052

Q13: System management settings should not reduce the MUVE usability.
Mean: 4.19
Mode: 4
Std. Dev.: 0.592
Std. Error: 0.105

Q14: Appropriate system security and controls ensure a successful learning experience.
Mean: 3.89
Mode: 4
Std. Dev.: 0.390
Std. Error: 0.070

Table 4.6: Questions on system environment management and descriptive statistics

Table 4.7: Summary of the analysis variables and their internal consistency measures

Variable Name (Factor) | Mean μ | Standard Deviation σ | Cronbach α | Number of Items
--- | --- | --- | --- | ---
Student Engagement | 3.598 | 0.283 | .802 | 7
Self-Regulation | 3.876 | 0.378 | .829 | 4
Environment Management | 3.950 | 0.175 | .80 | 4

The Spearman Correlation between Self-Regulation and Environment Management was 0.398 ($R^2=0.15$), indicating a weak positive relationship with significance ($p<0.05$). Therefore, it indicates...
that the two variables Self-Regulation and Environment Management are sufficiently independent for measuring two different aspects to represent MULE Management.

Regression analysis was used to examine the variable. A data fitness test was performed as the first step. For the test statistics of anticipated large effect ($F^2=0.35$), Number of predictors ($n=2$), Probability level of Significance ($\alpha = 0.05$) with the desired statistical Power level of ($1-\beta = 0.8$), the required sample size is $> 31$. Therefore, our sample size $N = 32 (>31)$ fulfils the analysis requirements as defined in [205] and [201].

The linear regression analysis model summary and the model fit (ANOVA) $R^2 = 0.759$ ($p<0.001$) indicate that about 75.9% of the variation in the student engagement is determined by the environment management and student self-regulation with the learning activities in the MULE, by means of a combined effect. The regression test coefficients and the summary are mentioned in Table 4.8. As the variable relationship with predictor parameters of the model, the model path coefficients are .240 for Self-Regulation, which is significant ($p<0.05$) and .657 for Environment Management with significance ($p<.001$).

<table>
<thead>
<tr>
<th>Variable</th>
<th>$\beta$</th>
<th>Standard Error</th>
<th>t-value</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self-Regulation</td>
<td>.240</td>
<td>.097</td>
<td>2.482</td>
<td>0.019*</td>
</tr>
<tr>
<td>Environment Management</td>
<td>.657</td>
<td>.092</td>
<td>7.312</td>
<td>0.000**</td>
</tr>
</tbody>
</table>

* $p < 0.05$
** $p < 0.01$

Table 4.8: Regression analysis summary

These analysis outcomes suggest that student self-regulation and environment management tend to be important aspects to manage a MULE.

4.3.5 Open Feedback

The questionnaire had a section with five open-ended questions, to capture the student opinions in general. Not all participants responded to these questions; however, a range of responses was observed with a reasonably good turnout. Some of the selected student expressions are shown in brackets.

**OFQ1** – Are there any privacy concerns you have regarding the use of 3D Multi User Virtual Environments for learning activities?

This question aims at capturing privacy concerns that the students have or felt while engaging with the learning activities. Importantly, this was taken as a measure for both Environment Management and Self-Regulation. The majority of the students felt that their activities are visible to others so the actions were committed consciously knowing that they are in an open environment. It is interesting to see that students felt the immersion at most while knowing that their work could be monitored, which is also possible with other information systems as a forensic measure, if they commit misconduct. However, students having a feeling of being a part of an open community may have resulted in shaping their behaviour. It is an open research possibility for further study.
[I feel psychologically aware about my progression being observed by others]
[No privacy with my activities]
[It is open, so privacy is a big concern]

OFQ2 - Are there any security challenges when using MUVEs for your learning, as an overall opinion?
With respect to the study, this question has the objective of identifying students’ trust of the existing MUVE (OpenSim) security implementations and policies employed. Moreover, students must have confidence about the learning environment reliability to engage comfortably in assessed and serious learning tasks. Overall, students showed their lack of awareness about the security and administrative measures available in the MULE.
[I believe it is safe and secure]
[There can be, but I’m not so sure, I need to learn more]
[I’m not sure]

OFQ3 - What are the areas of MULEs that should be addressed or improved?
Some students expressed their need on having a private space if the learning activities are not designed for collaboration. Secondly, various concerns were raised on familiarisation with the MUVE. Importantly, some students worried about certain challenges they faced due to the MUVE at the early stages of the learning activity. The importance of providing necessary user support was substantiated from student feedback.
[There should be availability of private space for some activities with data which are secure]
[Students need to be trained before they use the virtual world for assessment and laboratory work]

OFQ4 - If the MULE security is enhanced, can it affect the rich features and the usability of the system?
This question helped us observe the student view on implementing various environment management strategies to enhance system security. A concern on MUVE control as it can affect the intrinsic characteristics and usability of the environment. Students came up with vibrant answers, both supporting and against. A contextual approach for deciding the required level of management for learning needs was suggested.
[Yes, because it might hinder the entire idea of collaborative learning in a virtual environment]
[I don’t see an issue of security]
[It can be!]
[May be; but should not implement such measures]

OFQ5 - Any other comment/concern/suggestion about using 3D MUVE for learning?
As a final thought, all students who expressed their answers indicated the benefits of using MULE as a teaching and learner support environment. Importantly, their positive comments are highly encouraging and supportive of future studies on MULEs.
[I believe it should be handled on a wider scale and it is a great tool for interactive and collaborative learning]
[Generally, I like it as an educational methodology]
[It is interesting to use these stuffs]
In general, this open ended feedback showed some of the concerns students had. It was observed that student expressions were scattered across different functional and systems properties of OpenSim when considered in isolation; however, in the general context, those showed some relevance to the identified variables self-regulation and environment management.

4.3.6 Study Limitations

Careful analyses and tests were employed to minimise the impact of the following limitations. Due to the nature of the research, the study sample was limited to a particular set of students. These students have provided their feedback and answers based on their experiences, which were validated through observation. Furthermore, data used for the statistical tests were examined for their fitness through rigorous analysis ensuring the accuracy of the results. Therefore, the received data can be considered accurate. The questions used were appropriately designed, although they have yet to be examined for psychometric measures. It is a challenge to find a widely accepted standard set of psychometric measures in particular for MULE (OpenSim), as the field of study is still growing.

4.4 A Taxonomy for Managing a MULE

Based on the identified important management aspects, self-regulation (SR) and system environment management (EM), taxonomy was developed to map the possible scenarios of user engagement with respect to these two aspects. A two level scale, Low and High, was used for each aspect considering the nature of user activities. Therefore, the taxonomy presents four possible scenarios of user engagement: \{EM-High; SR-High\}, \{EM-High; SR-Low\}, \{EM-Low; SR-Low\}, and \{EM-Low; SR-High\}. Four parameters: control and monitoring, trust and security, perceived usability, and perceived benefits were incorporated into the description of the scenario characteristics. Control and monitoring along with trust and security emphasise the level of administration and management mechanisms and their impact to the learning session. Perceived usability represents how a student tends to experience the environment, especially with respect to his or her view on SR. Similarly, the perceived benefits are mainly associated with the educational gains from the environment scenario. Another four factors relevant to learning: nature of the interactions, engagement for achieving Intended Learning Outcomes (ILO), nature of learning environment and learning activities, and integration with blended learning infrastructure, were associated with respect to each scenario. Different characteristics of each of these factors based on the SR and EM scenarios, and their probable impact on the learning activities are shown in the taxonomy (Fig. 4.14).

The taxonomy can help academics for designing their management practices as suitable with the expected outcome or MUVE learning as well as the learning affordances MUVE provide. This user support can provide an insight on how SR and EM aspects can be used for a given scenario of learning.
As an overview, SR-High state represents students engaging with the MULE showing high self-regulation; they more likely refrain from undue exploitation of system features that are not related with their learning. In contrast SR-Low state represents students willingly exploit system features disengaging them from the learning objectives (e.g. a behaviour that is more suitable for a game progression). Also EM-High indicates system management practices are well implemented whereas EM-Low means a system management practices are not associated as required.

The four possible scenarios demonstrate unique characteristics (Fig. 4.14). Briefly, EM-Low & SR-Low do not provide reliable and successful learning activities, as indicated in the scenario characteristics. It is similar to using an isolated virtual world without any control or management practices while the students are keen to exploit features for entertainment, disengaging them from the ILOs. The perceived usability and benefits for learning depends upon the individual user preference. The other quadrant of EM-Low (SR-High) suggests having small-scale learning activities with high reliance on mutual agreements, although it is learner supportive. As EM is low, there is a difficulty of enforcing the control mechanisms to increase the trust and reliability of the learning environment among users and to meet the institutional regulatory requirements. Further, if integrated, it can compromise the existing blended
learning infrastructure. Learning activities with research and formative assessment might fit into this category.

The two quadrants with EM-High indicate that the MULE can be considered as a part of the institutional blended learning infrastructure, as the required environment management methods are practiced. In fact, lecturers and module coordinators are expected to implement policies to achieve the EM-High state, while incorporating the student SR-High state. Even if the SR-High is not attainable, EM-High scenario would help to have formal educational activities at large-scale in an integrated blended learning environment, although, the students might not explore the rich and flexible features of MUVE and may not be committed for achieving ILOs, as they feel constrained against their preferred behaviour. However, EM-High & SR-High is the desired state, which can provide higher usability, trust and educational value with cohesive student engagements. Learners with self-regulation are sensitive to the learning environment and possess the ability to follow the most suitable arrangements given by the learning environment without conflicts [206]. This suggests the rational of the research aim at EM-High & SR-High state when managed learning is practices with user support.

4.5 User Support for Policy Based MULE Management

4.5.1 Policy Based Management in Academia

Policy based management is widely used in academia. In fact, it has often been chosen as the preferred method to communicate a range of expected practices and agreed strategies in relevance to the academic environment at varying levels. At the strategic level, policies are usually made for a time period steering the institution and involved stakeholders to its goals. Leading universities divide their policies into main strategy areas to support the policy making process by reducing complexities due to the scale and wide spread nature of their operations. For example, the University of Cambridge’s Educational and Student Policy [135] includes a list of different policies at the operational level on academic and student matters, while providing a 3 year Learning & Teaching Strategy [207] as the top level policy document. University of St Andrews has a set of policies, in 15 sub categories, to communicate the university policy on teaching, learning and assessment [208]. The University of Dundee follows its e-Learning strategy with a well-defined set of activities and terms relevant to an e-Learning focused educational environment [136]. Likewise, almost all universities and Higher Education Institutes (HEIs) have highlighted their teaching and learner support activity management through a set of policies specialising in academic domain requirements. It has been the de-facto approach of communication when it comes to academic activity management and institutional monitoring and control.

User guidance for policy implementation and accepted practices adhering to the policies are often important for the success of policy based learning management. Staffordshire University has defined its Learning Policy in a way such that under each policy, implementation mechanisms are defined to achieve the respective policy [137].
This study suggests the policy based management for MULEs as part of the educational infrastructure. Suitable MUVE specific operational policies can be derived from the existing hierarchical learning management policies in a university or HEI. Since MUVEs (OpenSim/SL) introduce a challenging learning curve for use and management this research expects to provide user support for implementing policies at the operational level (i.e., OpenSim system function level).

4.5.2 A Process for Policy Implementation in MUVEs

As discussed earlier, the challenges associated with system models of OpenSim/SL can affect the accuracy and effectiveness of management practices. Because of that, when academics try to implement policies at the MUVE operational level they may find difficulties. In order to support management of learning through policy implementation using OpenSim/SL functions this study provides a set of user guidance tools. To make the MULE management practices more certain and convenient a process model is developed to facilitate policy implementation at the MUVE level, which is shown in Fig. 4.15. Kolb’s experiential learning stages [65] was incorporated as it cohesively facilitates the required tasks at different levels.

![Diagram showing the process model for policy implementation in MUVEs](attachment:process_model.png)

**Figure 4.15:** Process to be followed for developing policy considerations
The proposed taxonomy and the institutional learning management policies incorporate abstract conceptualisation of the management needs. In fact, this study suggests a close association with existing academic policies so that inconsistent management practices can be minimised at the MULE system level. An input from learning management facilitate blended learning needs with 3D support, since a single set of policies should be declared for the academic activities. MUVE (OpenSim/SL) functions and their complex interrelations represent the concrete implementations of the system. As users have to rely on implemented MUVE functions that incorporate concrete implementations this stage involves the available functions and their interrelationships that can affect expected outcomes of the declared policies. Finally, the two steps, effective consideration and reflective observations, associate with the policy implementation mechanisms involved in the management aspects self-regulation (SR) and system environment management (EM) as the outcome of the process. It is important to mention that the identified aspects, SR and EM, are the main focuses on the management practices, hence policy implementation mechanisms. The use of Kolb’s model with MUVEs has been successful in previous studies [33] [76]. This proposed process with the tasks aligned with Kolb’s Experiential Learning model can help users to overcome the steep learning curve of managing MULEs with confidence.

4.6 Concluding remarks

In this chapter, an abstract model to integrate MUVEs with the existing e-Learning and traditional learning environments has been discussed with possible integration opportunities at different levels along with a set of case studies. The chapter also presented a study carried out in OpenSim for identifying important management aspects with respect to MUVE supported learning: student self-regulation and environment management factors were identified to orient management practices within a MULE. Associating these factors taxonomy of user interactions for identifying management needs for different learning scenarios is introduced. Finally a process model is proposed for policy implementation at OpenSim/SL level, which will be further discussed in the next chapter.
5. Tool Support for Learning Management in OpenSim

5.1 Introduction

This chapter presents work done on providing user support for managed learning in OpenSim according to the process model discussed in the previous chapter; the developed tool can be used for SL management as well. Section 5.2 discusses the need for user guidance on OpenSim functions for policy implementations. An interactive user support tool was developed using graph theory and was analysed statistically for accuracy and structural mapping using the complex network parameters, which are explained in Sections 5.3 and 5.4. Section 5.5 presents the use of this tool for facilitating the management of a live educational environment in OpenSim. The findings of this case study show the contributions that the tool can make to support academics for MULE management. Section 5.6 concludes the chapter.

5.2 The Need for User Support in OpenSim Management

OpenSim based MULE management policies have to utilise OpenSim functions for their implementation. The available functions have a complex nature due to the functional interrelationships and dependencies across multiple levels. Not only do the EM based policy mechanisms have to be based on these functions, but also SR related management mechanisms are supported if the students can be informed about how to achieve the expected behaviour through these functions. Students can unintentionally behave contrarily to what is expected in policies, or can feel confused about their correct behaviour despite their commitment to practice self-regulatory learning if they are not aware of the these functions.

If we reconsider a basic consideration on managing avatar flying on a specific land parcel we may come up with an example policy as shown in Table 5.1. As the policy suggests it can be achieved through student voluntary practice (SR) and suitable control within the virtual region (EM). As we have discussed previously region level controls can introduce conflicts; the region level land management on avatar flying should not conflict with the parcel level (e.g. a lecture theatre area) fly settings. Secondly, students that are flying from other land parcels toward this specific parcel can still fly until they land on that parcel, despite its settings to disable fly. Once the incoming students reach the ground thereafter they cannot fly within the parcel. This conflict requires self-regulatory practice from students without exploiting this system feature. Moreover, if the land parcel setting is performed for a group, there can be conflicting settings on managing flying at land and group levels. These possible conflicting behaviours can be summarised briefly in Table 5.1.
As described in the previous chapter, the implementation of policies can be affected by possible conflicts at OpenSim function level. Often, the participating students can also benefit by informing them about these interrelationships. Importantly if the students are given dedicated parcels they should know how to manage their land parcels consistently with higher level management policies.

<table>
<thead>
<tr>
<th>AAM-Mobility_Fly-01</th>
<th>Students must refrain from flying during the lecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implementation</td>
<td>Student voluntary practice and/or prohibit flying - Land settings</td>
</tr>
<tr>
<td>Possible Conflicts</td>
<td>i) Region fly settings override Parcels’ settings</td>
</tr>
<tr>
<td></td>
<td>ii) Students flying toward the lecture theatre can still fly inside the lecture area until they reach the ground</td>
</tr>
<tr>
<td></td>
<td>iii) Group owned land fly settings override Parcel settings</td>
</tr>
</tbody>
</table>

Table 5.1: An example management policy with brief details on possible conflicts

A structured way of presenting the policy may appear more certain compared to explaining possible conflicts in lengthy paragraphs as part of a policy document. However, OpenSim has more than 200 distinct functions that a user can try from the client. Most of these functions are interconnected and the functional interrelationships can create significantly complex instances for an unfamiliar academic user. Therefore, a user study was carried out to examine specific user support needs in this regard.

5.2.1 User Study

Previously discussed user support models (the taxonomy and policy implementation process) can help users coordinate management of a MULE with respect to the formal education infrastructure. However, at the implementation level users may still find difficulties with OpenSim due to its complex system models.

In this study, the unique need for user guidance if the users decide to follow policy based management of MULE was examined. It is important since we need to identify specific user guidance needs due to the nature of MUVEs (OpenSim) that is distinguishable from the generic user guidance needs for any new users of a system. The study can help user support tool design since such support should address the user needs that are not covered in the existing standard user guidance available for OpenSim/SL.

The study was conducted with the participants in the Scottish Informatics and Computer Science Alliance (SICSA) PhD Conference 2011; over 100 PhD students from 13 Scottish Universities took part. These PhD researchers engage in teaching as demonstrators and tutors with technology adaptation at the grass-root level, such as laboratory experiments, simulations, assessments, etc. Moreover, this cohort is positive about adopting new technologies into practice, regardless of the challenge. Therefore, these young academics were particularly suitable for this study.
A short questionnaire was prepared taking into account the time these participants could commit during the conference. The questionnaire had 5 questions, which were designed in accordance with previous findings by associating user self-regulation (SR) and environment management (EM) aspects, orienting towards the thesis; therefore, this also helped to further validate the previous findings. Questions were designed with a 5-point Likert scale as: Strongly Agree (5), Agree (4), Neither Agree Nor Disagree (3), Disagree (2), Strongly Disagree (1). A detailed analysis of the question scores will be discussed in the next section.

5.2.2 Data Analysis

For data accuracy, the skill and knowledge variance within the sample was minimised. The conference provided an excellent opportunity to educate the audience through a poster and publications presentation portraying the objectives, models and findings of MUVE research at St Andrews. During the time span of the conference (2 days) this research project was discussed with 71 PhD students and identified 49 participants (69.01%) with prior experience of MUVEs (SL and/or OpenSim), of which 43 (87.76%) participated in the study (shown in Fig. 5.1). The study had five questions and an open-ended interview. All data items were collected anonymously; the university affiliation was collected only to label the data distribution as shown in Fig. 5.1.

As shown in Table 5.2, the majority of the responses was in favour of the questions resulting in an average ~4 (Agree) for all questions, as the mean values are normally distributed [X ~ N(4.022,0.1752)], satisfying Kolmogorov-Smirnov (K-S) test. Moreover, the questions Q3 and Q5 show Strongly Agree (5) as the most common answer, strengthening our observations.
Q1 was used as an instrument to examine the participants’ overall opinion of MUVE based learning. The question helped understand the participants’ perception of using MUVEs for education. On average, the participants were of the opinion that they agreed ($\mu = 3.91$) with the unique benefits of using MUVEs for learning, while ‘Agree’ was the most common response (mode).

According to the research model on user guidance, Q2, Q3 and Q4 were used for measuring items for the variable *MULE Management*. In fact, the three questions were used to represent the management aspects (SR and EM, Q2 and Q3, respectively) and the impact of management considerations based on SR and EM (Q4). With pre analysis it was found that Q2 and Q3 are strongly correlated (Spearman’s rho = 0.826, $p<0.001$) indicating that both questions measure the same variable; the effect of the parameters of management. The participant responses for Q2, Q3 and Q4 were further analysed and obtained a reliability measure for the variable *MULE Management*; the Cronbach alpha = 0.84 (>0.80), showed a strong internal consistency. Therefore, combined responses can be used.

Q5 was a direct question that asked the participant opinion about the need for user guidance on MUVE (OpenSim) functions. The phrase, *supportive presentations*, was used to distinguish text-based guidance from graphical representations of the information. The question showed the highest mean (4.30) out of the five and the majority strongly agreed with the statement (mode = 5) indicating that user guidance is essential. Q5 was used to represent the dependent variable. The variable was named as *Need for User Support*. Table 5.3 presents the summary of variables. The participants were aware of the user training already available or SL/OpenSim and this response suggests that they believe there is a need for further support for managing SL/OpenSim based MULEs.

### Table 5.2: Questionnaire items and descriptive statistics of the responses

<table>
<thead>
<tr>
<th>Question</th>
<th>Mean ($\mu$)</th>
<th>Mode</th>
<th>Std. Dev. ($\sigma$)</th>
<th>Std. Err. of Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1 3D virtual worlds (MUVE) facilitate learning and teaching with unique advantages.</td>
<td>3.91</td>
<td>4</td>
<td>0.684</td>
<td>0.104</td>
</tr>
<tr>
<td>Q2 Appropriate MULE user behaviour (student and staff) is important for the success of educational activities</td>
<td>3.93</td>
<td>4</td>
<td>0.799</td>
<td>0.122</td>
</tr>
<tr>
<td>Q3 Appropriate MULE management without losing its flexibility and rich features, is important for the success of educational activities.</td>
<td>4.09</td>
<td>5</td>
<td>0.840</td>
<td>0.128</td>
</tr>
<tr>
<td>Q4 The MULE management through policy considerations on user self-regulation and environment management benefit learning.</td>
<td>3.88</td>
<td>4</td>
<td>0.625</td>
<td>0.095</td>
</tr>
<tr>
<td>Q5 There is a significant need for user guidance with supporting presentations on MUVE functions.</td>
<td>4.30</td>
<td>5</td>
<td>0.741</td>
<td>0.113</td>
</tr>
<tr>
<td>Variable</td>
<td>Question Items</td>
<td>Mean</td>
<td>Standard Deviation</td>
<td>Cronbach’s α</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------------</td>
<td>--------</td>
<td>--------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>MULE Management</td>
<td>Q2, Q3, Q4</td>
<td>3.969</td>
<td>0.756</td>
<td>0.84</td>
</tr>
<tr>
<td>Need for User Support</td>
<td>Q5</td>
<td>4.30</td>
<td>0.741</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 5.3: Analysis variable summary with consistency measures

Linear regression analysis was used to examine the variable relationship. A sample size test was done for the fitness for regression analysis. For the test statistics of anticipated medium-large effect ($F^2=0.21$), Number of predictors ($n=1$), Probability level of Significance ($\alpha = 0.05$) with the desired statistical Power level of $(1-\beta = 0.8)$, the minimum required sample size was 39. Therefore, our sample size $N = 43 (>39)$ meets the requirements for the analysis as defined in [201, 205].

PASW (18.0) linear regression analysis and the model fit (ANOVA) $R^2 = 0.686$, indicates that appropriate user support is required for managing OpenSim based MULEs. The variable relationship with predictor parameters of the model, i.e., the model path coefficient ($\beta$), is .828, which is significant ($p < 0.001$). It indicates that sufficient user support strategies can be required for implementing learning management policies through SL/OpenSim functions. This suggests supporting users with suitable training and guidance tools for their MULE management practices at the system (OpenSim/SL) level.

5.2.3 Findings from the Interview

Participants were asked to express their opinion on having user guidance for MUVE functions. Three major areas of interests were identified; in brief:

- The majority recommended that the guidance should be multi-faceted supporting all user groups. In fact, the user guides should indicate the OpenSim/SL function behaviour irrespective of user type.

- Some had the view that the guidance should be a continuous process until the users are thoroughly aware. Their rational was that OpenSim is not yet stable for serious learning activities; a mechanism should be in place for updating the guidance as needed. This was useful feedback indicating that users tend to engage dynamically with the latest system functions. This can easily be accommodated as a feedback step in the training process.

- Some were concerned about the relative position of MULE within the existing learning environment infrastructure. They were informed that this research follows the blended learning approach, and the models are based on a generic blended learning infrastructure with 3D support, as in [12].

A few shared their experiences on using OpenSim and the problems they faced. One participant said: “...I was making some ancient Egyptian sculptures on the sandbox given to me; when I was away someone had raised the land and buried my stuff...” This could have been easily avoided, had the appropriate content management and land settings been used.
5.2.4 Overview of the Study Findings

The study findings indicated that the participants supported the idea of the suitable means of user guidance on OpenSim system functions for managing MULEs. It is important since these participants were aware of available standard user guides for OpenSim, yet still indicated the unique need of user support for managing OpenSim/SL. Moreover the scores for questions Q2, Q3, and Q4 further supported the previously identified management aspects, self-regulation and environment management from another sample of participants.

The study was subjected to the following conditions: participant opinions were captured through the Likert scale as the standard practice in user studies. To increase the accuracy of analysis, suitable non-parametric tests were performed, while pre-testing the fitness of data for the considered analysis. Also the test sample consisted of PhD students, which was essential to minimise the skill variance. However, further studies with broader user communities would be helpful for incorporating other views.

5.3 Rendering OpenSim Function Interrelations for User Guidance

Details about network topology rendering for complex OpenSim functional interrelationships for user support are described in this section. This tool facilitates users understanding the conflicting or cohesive nature of similar functionalities from different MULE management areas.

5.3.1 A Framework of OpenSim Functions for Management

OpenSim based MULE management policies depend on relevant OpenSim functions for their implementation. Academics have to rely on these functional behaviours and their interrelationships to decide appropriate policy implementation mechanisms. Therefore, effectively the OpenSim function categories must be used as a framework for this tool design. The following function category framework was prepared (Fig. 5.2) based on the system study findings to facilitate policy implementations in OpenSim.

As shown in the framework, OpenSim based MULE management policies have to be implemented in coordination with the existing (generic) learning management policy implementations. Higher level management policies often get precedence in defining the ways learning activities are conducted and managed. For example, an institution specific server access policy may not allow certain network communication ports and protocols from outside, which can affect the access of OpenSim environment from public networks. Therefore, due considerations should be given on generic learning management policies when trying to implement MULE management policies.
The function categories, indicated as OpenSim functions in Fig 5.2, determine management practices within OpenSim. Among these categories avatar activities are the fundamental mechanisms that students interact with in a learning environment in OpenSim. This can be the most challenging set of functions to manage as the users interact with the environment through these; hence they are highly vulnerable to misuse by SR-Low users, yet the restrictions on these functions can be costly in terms of perceived usability. The policy implementation mechanisms in this category have a higher level of association with SR aspects than the EM aspects guiding the students on their expected behaviour within the learning environment.

OpenSim System Management core functions include land, content, user and group management. As discussed in Chapter 3, these function groups follow system models that define their behaviour and functional interrelationships. Importantly, most of the OpenSim system management activities are done through a land specific function category; hence the implementation mechanisms tend to incorporate more EM aspects; SR aspects too can be essential for certain user engagements, however. For policies that need to be implemented using content related functions both aspects of SR and EM can be important depending on the learning arrangement according to the taxonomy. User management related functions can be used to promote SR aspects in policy implementation; e.g. appropriate avatar naming strategy can help shape student behaviour. With the group related functions EM based management practices are important in managing group roles and abilities, while SR aspects are essential when defining member behaviours.

The advantage of this framework is that academics can be given a high level, concise view on available function categories in OpenSim. To provide more concrete user support for using OpenSim functions a tool was developed, which is discussed in the next section.
5.3.2 A Network of OpenSim Functions

Over 200 unique OpenSim functions with about 350 relationships among those functions that can be relevant to the learning with OpenSim were identified. Considering this complex information is a challenging task for any user. To overcome this challenge and analyse the inter-function relationships, an approach with graph analysis was followed.

Graphs, for representing complex interrelationships, have been widely used in multiple disciplines with various applications. Graphs can facilitate a rapid uptake of abstract information through visuals [209]. A directed MUVE function graph (di-graph) G can be defined as G = (V, E); where V is the set of OpenSim functions (vertices), and E is the set of relationships, also known as edges, between those functions. Furthermore, the nature of these functions and their structural design seems to form a complex network. With the large number of functions and their complex interactions, more intuitive user support tools are required for OpenSim based MULE management.

In research on an interactive visualisation tool design for e-learning management, Jyothi [210] emphasised that the visualisation of network interactions in a graphical way assists the moderator or the instructor to understand, at a glance, several important concepts without any further investigation or research. Visualisation tools embedded in e-Learning systems allow teachers to reveal useful characteristics appropriate to the management of learning activities [211]. However, the mere visualisation support in a graphical way cannot effectively support educators; a successful guidance tool should provide interactive operations to manipulate the graphics such as zooming and filtering [212]. Therefore, the guidance tool for helping users on OpenSim functions has to be dynamic and intuitive with an intrinsic flexibility given to the users to select the required sub topologies from the entire network. This effect of popping out the selected graph from the rest can help the users to focus on their selected policy and implement those rapidly. Using the 'pop-out' paradigm allows probing of elevated information from graphs, making it a viable approach for efficient data visualisation [213]. The network analysis and visualisation tool Gephi [version 0.8α] [214] was used, as it aids users to rapidly visualise function interdependencies and statistical analysis of the network.

Gephi supports multiple formats of node and edge data inputs to generate the graph view of a complex network. As a result, Gephi has been widely used in complex network analyses such as social networks, biological networks etc. However, most of these complex network data are either readily available for generating the graphs or can be easily extracted from the sources in a format that is supported by visualisation tools such as Gephi. With respect to OpenSim functions neither of these options was possible; therefore, this function topology was developed by creating the data source of functions and interactions identified through system studies.

Gephi allowed two options to develop the network; either to draw the topology on the visualisation template itself using the draw palate or write the graph topology source using the syntax it understands. A graph description language such as DOT [215] can be used to define the textual syntax structure of the graph topology, which is supported by Gephi syntax processor. The DOT language has a simple structure and syntax that enables the expression of nodes links and formatting information for presentation. A DOT language script with basic properties to represent a selected sub network topology of OpenSim function network is shown in Fig. 5.3.
Once the *.dot script file is loaded to Gephi, it generates the network with nodes and edges. Further customisations can be easily done to the selected nodes and edges for enhanced presentation effects using the Gephi GUI tools. In particular, editing the graph using the GUI seems more efficient when only few changes are needed. The corresponding sub topology for the DOT script shown in Fig. 5.3 obtained in Gephi with node size modifications to highlight the major functions areas is shown in Fig. 5.4.

The node colours are used to differentiate function categories. To maintain consistency, the same colour convention that was introduced in the function framework was used. i.e., Avatar Activity functions – green, Land management functions: Region level – red, and Parcel level – brown, Group management functions – yellow, User management functions – purple and Content management function – pink. Middle level functional categories of structural importance are represented in grey. The edge colours were defined to represent the state of the functional interaction, i.e., the usual operational interaction – orange, conflicting or overriding interaction – red and supportive interaction – green. Black edges indicate the structural relationship with the function category. The directed graph edges indicate originating function (source node) and the function that gets affected (target node).

**Figure 5.3:** An example graph topology written in DOT

```plaintext
digraph g{
  "33"[color=green, label=“Avatar Activity”];
  "34"[color=gray, label=“Avatar Mobility”];
  "35"[color=green, label=“Teleport”];
  "36"[color=green, label=“Walk”];
  "37"[color=green, label=“Fly”];
  "33" -> "34"[color=black, label="“”];
  "34" -> "35"[color=black, label="“”];
  "34" -> "36"[color=black, label="“”];
  "33" -> "37"[color=black, label="“”];
}
```

**Figure 5.4:** The graph representation of Gephi for the topology script in Fig. 5.3
Figure 5.5 shows the overall network of functions that has been identified. It shows the complexity that the academics and students have to work with OpenSim based MULEs. This overall network is shown here only to provide the holistic impression; it cannot be considered as a usable guidance. However, fortunately, the Gephi interface allows the users to select a function’s ego network with all related interconnections by filtering out the rest.

Figure 5.5: The complex network of OpenSim functions for educational uses

Fig.5.6 shows the filtered ego network of the function Avatar Flying. The legend presents the colours used to highlight the nature of different function interrelationships in the selected network topology. The example of avatar flying management that was discussed previously can be examined with this network; the network shows how the avatar flying can be managed using the available OpenSim functions from different yet interconnected function areas. If we recall the textual descriptions of possible conflicting situations for flying settings on lecture delivery area, the ego network of the fly function can depict all the information in an intuitive manner for an unfamiliar user. Not only does it provide a clear view of the function of interest, but also helps users to select their policy implementations with the context of conflicts/supportive functions for the particular need of management.
Figure 5.6: The selected Avatar Fly ego-network and the legend for users

Through the Gephi interface, users can easily select the OpenSim functions that they are interested in with the flexibility to define a desired depth of the topology. The developed network topology was designed in a way that by selecting 2 or 3 levels of depth from the selected function node, users can get a complete view of the function behaviour without additional complexities or unrelated information. Section 5.5 presents a study about the tool’s effectiveness and usability for managing an OpenSim based MULE, using an actual educational setup. More details on different ego networks will be presented in the context of relevant Management policies.

5.4 Tool Evaluation for Accuracy

The developed tool can be used to overcome the challenges of textual descriptions of OpenSim function interrelationships. However, no matter how usable or simple the tool is, it has to accurately depict the true nature and the significance of the functions. Therefore, the complex network topology of OpenSim functions was analysed for its accuracy with suitable statistical measures.

5.4.1 Evaluation Using Eigen Vector Centrality

It was decided to use Eigenvector Centrality [216], which is defined as the principal eigenvector of the adjacency matrix defining the network, as the centrality measure to examine the network interrelationships. It has been successfully used for similar purposes in previous studies for evaluating the node interrelationships in a complex network [217].
The Eigenvector (EV) metric has two important properties: first, it captures the fact that a node (a function) that influences many other nodes is influential (has a higher value). Secondly, the property that makes the EV metric unique: the EV centrality measure considers that a node that affects many highly influential nodes is more influential than a node which affects many weakly influential nodes [217]. EV measure follows the approach that all nodes influence their neighbours, without necessarily being confined to the shortest path node connectivity [218]. This is an appropriate measure as our functional network has multi-path interlinks that must be accounted for in an accurate analysis.

The built-in analyser in Gephi was used to obtain the Eigen Vector Centrality (EVC) values for each OpenSim function. The functions were then ranked using these EVC values and assigned the corresponding EV Rank in the ordered list of the functions.

In parallel to this analysis, the list of the OpenSim functions used in the network topology were grouped into 5 categories of ordinal measures (with encoding values) as Very High = 5, High = 4, Moderate = 3, Low = 2 and Very Low = 1 based on the Perceived Level of Significance (PLoS). The PLoS category of each function was obtained as a subjective measure from an expert analyst and further viewed by another analyst in order to eliminate significant anomalies. The grouping of functions on PLoS was performed in mutual exclusion with the EVC values so that the encoded PLoS values for the functions were treated as independent measures based on system studies and expert views.

The two data sets were treated as independent samples of data. A comparison of EVC and PLoS scores for a selected set of 20 functions is shown in Table 5.4, (ordered according to EV Rank).

Spearman’s correlation coefficient between EVC and PLoS was 0.847, (N=211). It indicates a strong positive correlation that explains over 70% of the variance in the values tested, with significance (p<0.01).

<table>
<thead>
<tr>
<th>EV Rank</th>
<th>Function</th>
<th>EVC</th>
<th>PLoS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Content Management</td>
<td>1.000</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>2</td>
<td>Teleport</td>
<td>0.844</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>3</td>
<td>Avatar Activity</td>
<td>0.768</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>4</td>
<td>Avatar Terraform</td>
<td>0.695</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>5</td>
<td>Create Content Objects</td>
<td>0.666</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>6</td>
<td>Permission Settings on an Object</td>
<td>0.619</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>7</td>
<td>Avatar Fly</td>
<td>0.496</td>
<td>High (4)</td>
</tr>
<tr>
<td>8</td>
<td>Edit Content Objects</td>
<td>0.451</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>9</td>
<td>Allow Create Objects</td>
<td>0.443</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>10</td>
<td>Manage parcel access list</td>
<td>0.370</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>11</td>
<td>Near field spatial management</td>
<td>0.331</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>12</td>
<td>Region Estate Management</td>
<td>0.330</td>
<td>Very High (5)</td>
</tr>
<tr>
<td>13</td>
<td>Bypass Permissions</td>
<td>0.289</td>
<td>High (4)</td>
</tr>
<tr>
<td>14</td>
<td>Group Management</td>
<td>0.286</td>
<td>High (4)</td>
</tr>
<tr>
<td>15</td>
<td>Force permission ON/OFF</td>
<td>0.285</td>
<td>High (4)</td>
</tr>
<tr>
<td>16</td>
<td>Allow Script Run</td>
<td>0.209</td>
<td>Moderate (3)</td>
</tr>
<tr>
<td>17</td>
<td>Alert massages to users</td>
<td>0.031</td>
<td>Low (2)</td>
</tr>
<tr>
<td>18</td>
<td>Set Parcel Name</td>
<td>0.027</td>
<td>Low (2)</td>
</tr>
<tr>
<td>19</td>
<td>Set Music URL</td>
<td>0.024</td>
<td>Low (2)</td>
</tr>
<tr>
<td>20</td>
<td>Group visibility change</td>
<td>0.003</td>
<td>Very Low (1)</td>
</tr>
</tbody>
</table>

Table 5.4: A sample of EVC ranks and corresponding PLoS values
Although the PLoS categories were derived in 5 scale ordinal distribution to increase the accuracy in capturing subjective scores from human analysts, in practice Eigen values are more effective in indicating two contrasting samples: highly important and less important items. Therefore, to evaluate the accurate representation of the functional importance in the network topology the PLoS values were further categorised into two sub samples as Importance-High PLoS (≥ 3) and Importance-Low PLoS (< 3). The analysis could have even been done by selecting only the two PLoS samples at the extreme ends (i.e., Very High -5 and Very Low -1); but the entire sample of functions was used to increase the analysis accuracy where the entire topology is accounted for the analysis.

The corresponding EVC measures were grouped into two sets in accordance with the two PLoS samples derived. The descriptive statistics of the EVC values in the two PLoS samples (from PSAW), are shown in Table 5.5. The descriptive statistics show some indication that the high PLoS functions also show a relatively higher EVC values when comparing the means of the two samples. However, this was tested statistically for the accuracy of the topology. For that, the following examinable hypotheses were used (as required by the test statistics).

**Null Hypothesis**: There is no difference between the distribution of corresponding Eigenvector Centrality (EVC) values, across the Importance-High and Importance-Low PLoS categories

**Alternative Hypothesis**: There is a difference of the distribution of Corresponding EVC values across the Importance-High and Importance-Low PLoS categories.

<table>
<thead>
<tr>
<th></th>
<th>Importance_High (PLoS)</th>
<th>Importance_Low (PLoS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N (number of functions)</td>
<td>137</td>
<td>74</td>
</tr>
<tr>
<td>Mean</td>
<td>.221</td>
<td>.092</td>
</tr>
<tr>
<td>Std. Error of Mean</td>
<td>.0138</td>
<td>.0027</td>
</tr>
<tr>
<td>Median</td>
<td>.158</td>
<td>.085</td>
</tr>
<tr>
<td>Std. Deviation</td>
<td>.1612</td>
<td>.0231</td>
</tr>
<tr>
<td>Minimum</td>
<td>.0000</td>
<td>.0000</td>
</tr>
<tr>
<td>Maximum</td>
<td>1.0000</td>
<td>.2013</td>
</tr>
</tbody>
</table>

*Table 5.5: Descriptive Statistics for the two groups of PLoS*

Independent Samples Kruskal-Wallis test was employed using PASW (18.0). The result, (Fig. 5.7) suggests rejection of the Null Hypothesis with significance ($p<0.05$). Having rejected the Null Hypothesis, we can say that high EVC values are more probable to be a member of the high important PLoS population, than the low important population of PLoS. This indicates the accurateness of EVC values to represent the importance of OpenSim functions with their interrelationship behaviours.
The EVC values can also be used as a measure to prioritise different OpenSim management considerations. It can be a significant challenge for a non-expert to decide which functions to focus on or prioritise, let alone figuring out the complex interrelationships between those functions. EVC values can help users to identify critical functions, which have a high influential nature, so that a reasonable consideration can be made for managing those. The study findings help users to rely on a statistically verified ranking mechanism, in case of ambiguity or doubtfulness on which functionalities should be prioritised for their policy implementations.

### 5.4.2 Evaluation Using Betweenness Centrality

EVC values do not take into account the holistic nature of the function network. That is, the analysis using EVC considers the functional significance with interrelationships, but not how the functions are actually interconnected in the structure. For example, a user may consider high EVC functions with high priority to manage, but may not have a clear idea on how the functions are implemented in functional categories and the structural interconnections irrespective of the importance; this can result in difficulties in locating the function’s ego network and the relative position. This is a well-known property in complex networks where EVC values cannot be used. Betweenness Centrality (BWC) [219] was considered as a measure to evaluate the network accuracy in depicting the structural interconnections of the functions. The built-in analyser in Gephi provides BWC measures as well. Table 5.6 shows a sample of 15 functions with their BWC values selected to represent the range of the distribution.

The Pearson correlation coefficient for EVC and BWC $R = 0.179$ ($R^2=0.032$) indicates a negligible relationship on variances as the two measures are effectively representing different properties of the function network. This indicates the use of BWC to complement the previous analysis based on EVC. A ranking mechanism for BWC was not used. BVC values do not provide a meaningful metaphor in the 3D learning environment context since they merely depend upon the number of functional linkages and relative positions among each other. In fact, it would be an imprecise notion to categorise BVC values. Therefore, it was treated as a qualitative study on function architecture; it was performed through observations.
<table>
<thead>
<tr>
<th>Rank</th>
<th>Function</th>
<th>Betweenness Centrality</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Group Management</td>
<td>912.30</td>
</tr>
<tr>
<td>2</td>
<td>Near Field Spatial Management</td>
<td>784.23</td>
</tr>
<tr>
<td>3</td>
<td>Region Estate Management</td>
<td>721.83</td>
</tr>
<tr>
<td>4</td>
<td>Avatar Activity</td>
<td>642.14</td>
</tr>
<tr>
<td>5</td>
<td>Content Management</td>
<td>611.87</td>
</tr>
<tr>
<td>7</td>
<td>Avatar Behaviour</td>
<td>529.38</td>
</tr>
<tr>
<td>10</td>
<td>Region Estate - Estate Management</td>
<td>386.09</td>
</tr>
<tr>
<td>12</td>
<td>Near Field Spatial Management - General</td>
<td>331.07</td>
</tr>
<tr>
<td>15</td>
<td>Region Estate Debug Management</td>
<td>202.68</td>
</tr>
<tr>
<td>21</td>
<td>Group Parcel Access Management</td>
<td>151.31</td>
</tr>
<tr>
<td>50</td>
<td>Group Object Management</td>
<td>63.46</td>
</tr>
<tr>
<td>100</td>
<td>Parcel set passes</td>
<td>11.03</td>
</tr>
<tr>
<td>147</td>
<td>Set Landing Point</td>
<td>3.49</td>
</tr>
<tr>
<td>195</td>
<td>Invite users</td>
<td>0.00</td>
</tr>
<tr>
<td>207</td>
<td>HUD attachment</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Table 5.6: A selected sample of Betweenness Centrality measures of functions

If we consider the top BWC values, they represent the major areas of OpenSim functions: Group Management, Near Field Spatial Management, etc. The mid-ranged BWC values associate with Group Object Management, Group Parcel Access Management, etc. The smaller BWC values represent: HUD attachment, Invite Users, etc. The functions with high BWC values represent the common categories (sub communities) of functions. The mid-range BWC values represent sub categories of functions that cluster several derived solo functions; these are within the major communities, however. Finally, the individual functions such as HUD attachment, Instant Messaging, Invite users, etc. represent the perimeter functions in the functional network; with low values of BWC. Therefore, the distribution of BWC indicates the hierarchical relationship among functions, irrespective of their importance. As a result, users can benefit from the visual representation of this function network for their management needs. Moreover, this validates our approach of depicting the OpenSim functions as a network, since the network properties match with the actual function implementations and their relationships.

5.4.3 Analysis Overview

The statistical and qualitative analysis of the user guidance tool suggests that the required guidance on OpenSim functions can be given by portraying the functions as a network of functional interrelationships. The analyst ranking mechanism used in PLoS was a subjective measure in view of the operational significance of the considered function. Human analysts can have slightly different ranking opinions on functions that give ties on importance as per their observed behaviour, which can affect the analysis. However, these errors are distributed between the two groups: Importance High and Low.
The management policies mainly depend on the academic preference and contextual properties of the learning activity; these contextual policies on managing learning activities in OpenSim can be supported using the tool. The guidance network topology is prepared ‘as is’ with respect to the OpenSim system implementations. A given OpenSim function can be used to implement multiple policies [one to many association: i.e., 1:*] depending on the contextual norms of learning in these MULEs, which depends on user preference. In that respect the function network tool contribution appears to be multidisciplinary in nature; it can probably be used in most of the domains that use OpenSim; however, further studies are recommended before directly using it in a context other than MULEs with OpenSim.

5.5 Case study – Tool Evaluation

To evaluate the tool, one of the learning support projects, the Virtual Humanitarian Disaster training (VHD) project [95], was used as a case study. Since the analysis presented later relates to this project extensively, a brief overview on the VHD project and associated objectives is discussed in Section 5.5.1.

5.5.1 Case Study Overview – The VHD Project

The Virtual Humanitarian Disaster training environment [95] was developed to support training aid workers by providing a more flexible teaching and learning environment in comparison to traditional classroom methods. The VHD project was designed to facilitate training needs through the simulation of various situations encountered in the humanitarian disaster management domain. Most of the environment and learning content have been designed according to standard practices and definitions within the field of humanitarian disaster management. The following is a brief introduction to the field of training for humanitarian disaster management, since the domain requirements play a significant role in shaping the VHD project usage and its management.

Training for humanitarian disaster management

There are many definitions of a humanitarian disaster from various global institutions such as the International Federation of Red Cross and Red Crescent Societies (IFRC) [220] and the United Nations Development Program (UNDP) [221]. A number of studies on disaster management have left the meaning of disaster implicit [222] and intuitive to the context. The VHD project considers a disaster as a crisis event [95], which is either natural or man-made, that affects the inhabitants of a specific location. Humanitarian aid is the assistance provided in response to a humanitarian crisis. Aid may be logistical, financial, material, counseling, etc. aiming at managing the crisis situation by alleviating human suffering and saving lives. Appropriate provision of required training for aid and relief workers can reduce their stress related challenges and preparing them for the demanding situations [223]. The Dept.
for International Development UK, humanitarian emergency report [224] highlights that the quality of aid personnel is a major factor in humanitarian disaster response. A major challenge identified in the report is the lack of straightforward professional training channels into humanitarian work; one of the key objectives of the VHD tool is to promote the professional activities in disaster management among students using an engaging aid.

Field activities on disaster management are challenging and provide instantaneous feedback on the actions, while a sufficient level of theoretical and methodical training facilitate a clearer vision on field activities, explanation, prediction and control abilities to the workers [222]. This indicates the importance of combining theoretical knowledge and field activities for a successful training design. Available technology supported education tools can be helpful for developing effective training programs for classroom teaching and research on disaster management [225]. The benefits and learning affordances of MUVEs [66] prompted the design of a VHD environment within OpenSim.

The VHD project was developed as a learning tool for the honours module (SCQF level-10 [198]) MN4266 – Non Governmental Organisations offered by the School of Management, University of St Andrews. The project relates to the final thematic area of the module, i.e., the strategic and operational challenges faced by NGOs operating in the humanitarian relief industry. The key advantage of the tool is that it allows students to take decisions concerning critical situations within the controlled environment of a virtual world where the wrong decisions taken will not have consequences on lives and property in the real world. The design and implementation of the refugee camp and the training area was based on the specific guidelines and standards given by the Office of the United Nations High Commissioner for Refugees (UNHCR) [226]; Fig. 5.8 shows the VHD training environment layout with information about major locations in a refugee camp.

![Figure 5.8: The map of the designed VHD training environment](image)
Fig. 5.9 shows actual implementations of the VHD islands from two different view angles.

The Moodle e-Learning environment in University of St Andrews was used to associate student identities with the module activities, and Sloodle tools were used to support various learning and assessment tasks in OpenSim and to integrate Moodle and OpenSim user identities and privileges. The Sloodle Registration Booth maps the avatar and student identities across the Moodle and OpenSim platforms. Sloodle Presenter has been used to present learning content and multimedia as scaffoldings for student learning. For the assessment and feedback requirements, Sloodle Quiz Chair and Choice tools have been used.

5.5.2 VHD Project Management – User Support Approaches

Managing the VHD environment for different learning scenarios was a significant challenge faced by the researchers and academics associated with the project. Often, they communicated with the MUVE research staff in the School of Computer Science for troubleshooting and solving the management challenges they encountered. To support their work, the main project researchers were informed about the user support which had been developed. The initial attempt to provide user guidance involved word processed documents with tables of functions since the complex network topology with Gephi had not been completed during the early stages of VHD. The VHD project researchers were asked to refer to the document for policy implementation; their feedback was crucial for the subsequent tool design. To help the VHD researchers, a dual display apparatus that had enough screen size for convenient testing was tried. Fig. 5.10 shows a display screen (dual display setup) snapshot image of a VHD project researcher referring to the guidance document tables while attempting to implement certain policies, in-world (as avatar). Obviously, this practice was a cumbersome experience; the users (VHD project researchers) said that they learnt about OpenSim functions from that document to some extent; however the layout and the presentation with tables made the reference document lacking in usability. A major drawback was the inability to portray the functional interactions and the nature of such complex relationships. The users had to refer to several sections back and forth to see the structure of the functions and then interpret the possible interaction; this resulted in many trial and error practices.
When the function network topology was completed the VHD researchers were asked to use it with a Gephi interface instead of the document they had been using. In a similar setup (dual display) they tried the user guidance network. The main advantage of the functional network representation was the researchers could filter an ego network of a considered function with a desired level of depth. Fig. 5.13, 5.16 and 5.18 show such ego-networks with different levels of function depths that VHD researchers tried. This facility to select a sub-network based on the ego network of a given function allowed them to visualise the functional interrelationships easily and intuitively relate that information with their policy implementation mechanisms during the process.

The following three main scenarios of usage were practiced with the VHD tool to facilitate student learning about disaster management. These scenarios were based on the Intended Learning Outcomes (ILOs), Teaching and Learning Tasks (TLAs) and Assessment Tasks (ATs) of the module. Future potentials of learning and research use cases were also incorporated for policies per each scenario. By doing that, a thorough and holistic evaluation of the OpenSim function network tool was expected in an actual MULE setup.

5.5.3 Scenario I – Training with VHD Environment

This is the primary usage of the VHD environment, in which students are allowed to observe the various constructions in the humanitarian support area and engage in a series of short quizzes as an assessment of the learnt knowledge. The main objective of this arrangement is to provide students with a learning aid that helps them to reflect the learnt theories and related knowledge in an interactive and engaging learning environment. Importantly, this scenario does not allow students to experience unique MUVE tasks such as modifying the environment, content creation or unlimited access. Therefore, the scenario characteristics fit well into the category of High-EM, irrespective of user self-regulation (SR low or high, although High-SR is anticipated). Students are given pre-arranged options of interactivity through guided stepwise learning paths, as the tasks are related to the assessments of the course modules. Fig. 5.11
shows a stepwise participation in the learning activities about dilemma management as part of learning scenario 1.

At the end of the stepwise ordered learning engagement, students are directed to the assessment centre with Sloodle support. Student activities in this scenario are mainly concentrated on the selected Sloodle tools – Sloodle Choice and Sloodle Quiz Chair. Fig. 5.12 shows Sloodle supported student assessment activities. First, the students are guided from one learning location to the next using numbered arrows. In each location, a range of dilemma situations is presented through Notecards, as shown. Then students are asked to participate in the assessment tasks, in which they have to provide their decisions to address the dilemma and support the refugees while meeting the aid agency objectives. Through the Sloodle interface, the grades and answers are presented at the discretion of the teacher, while updating the Moodle records at the same time, for data consistency.

The learning activities in this scenario show the basic and typical learning use cases in an e-Learning online learning environment. Therefore, the policies for managing this MULE followed a strategy on minimizing 3D environment exploration features as such facility can distract the students from their learning objectives. After all, as the VHD project staff indicated in this scenario of use, students are using the VHD Island as only a part of their course with the rest based in Moodle and classroom teaching. A selected set from the policies that the researchers have tried to implement with VHD are shown in Table 5.7. These policies are defined according to the learning management needs of that scenario. Policies are grouped into the OpenSim function categories for clarification.
Figure 5.12: Directed learning paths and Sloodle activities for VHD learning scenario 1

<table>
<thead>
<tr>
<th>Category</th>
<th>Management Policies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Land Mgt.</td>
<td>Students should not terraform the learning environment</td>
</tr>
<tr>
<td></td>
<td>Students should not be given land ownership</td>
</tr>
<tr>
<td></td>
<td>Student must refrain from accessing restricted areas (if defined for a learning activity)</td>
</tr>
<tr>
<td>Avatar Activity Mgt.</td>
<td>Students should engage in the learning tasks for ILOs</td>
</tr>
<tr>
<td></td>
<td>Students should not distract others’ learning</td>
</tr>
<tr>
<td></td>
<td>Students should not misbehave in the environment</td>
</tr>
<tr>
<td>User Mgt.</td>
<td>User names may be mapped with real identities</td>
</tr>
<tr>
<td></td>
<td>Users should be given appropriate privileges for tasks</td>
</tr>
<tr>
<td></td>
<td>Users may not change their given home locations</td>
</tr>
<tr>
<td>Content Mgt.</td>
<td>Students should not alter/move learning content</td>
</tr>
<tr>
<td></td>
<td>Students should not create content objects</td>
</tr>
<tr>
<td></td>
<td>Students should not alter/move the environment content</td>
</tr>
<tr>
<td>Group Mgt.</td>
<td>Students should not change settings of their assigned groups</td>
</tr>
<tr>
<td></td>
<td>Students should not change their assigned groups</td>
</tr>
<tr>
<td></td>
<td>Students should not alter group owned objects or land</td>
</tr>
</tbody>
</table>

Table 5.7: A set of policies for the management of VHD learning Scenario I

For the implementations of these policies, the OpenSim function network tool was used by the VHD staff. Functions related to each policy were examined using the tool; by doing that possible implementation methods were identified. To indicate the benefit received by the VHD staff, an example policy implementation from the above list is selected for further elaboration.
The selected function, Avatar Terraforming (changing 3D virtual land settings) and its functional interrelationships are shown in Fig. 5.13 (depth level 1). The policies indicate that students are not allowed to change the land attributes through terraforming. Here, the VHD staff members have considered a set of land editing functions such as Flattening, Raising, Smoothing, Reverting, etc. As the ego-network indicates, certain functions in parcel and group management intrinsically favour terraforming (green colour edges). Some of the functions at group and parcel management have a toggle behaviour i.e. they can be either true or false (orange colour edges). VHD environment incorporates different land parcels and group settings; therefore, to prevent students from terraforming, the VHD researchers have also used the Region level functions to block-terraform. The VHD staff reported that the use of the tool made their work efficient and reliable since they can check the conflicting interrelations as they set the EM functions through the client application. Moreover, since this scenario is the most frequent type of use, academics indicated that the efficiency and reliability brought in by the network topology in Gephi is quite useful for managing the VHD Island.
5.5.4 Scenario II – Role Play and Simulation

This scenario of learning differs from the first scenario as the students and teachers use the VHD environment as a means of simulating a real conflict and disaster environment. The main objective of this arrangement is to facilitate training for disaster management as a set of field studies. Role-play and task association for each role can be utilized only if the OpenSim functions are understood and properly used.

The scenario setup required special avatar appearances to simulate the real tenants of a refugee camp or conflict zone. Separate textures were developed for designing suitable clothing for the user roles as part of the VHD project. Customised clothes and avatar body-parts were developed and archived so that the role-players can use those through their avatar inventories. Fig. 5.14 shows the designed avatar shapes and clothes for refugee tenants (female and male avatars). Fig. 5.15 displays the avatar shapes and officer uniforms of two key roles in a disaster management activity: an aid worker and a camp official.

Since the system control on avatar customisation is difficult, the students who engage in the role-play should show a strong level of self-regulation to meet the expected learning objectives for themselves and other participants. Students are expected to refrain from changing their given role appearance throughout the session; this was achieved through relevant policies on avatar activity management (SR and EM). VHD project staff used two approaches when allowing students to role-play: i) having a defined set of accounts for specific roles which students use temporarily; and ii) asking student avatars to load related role inventories and objects (clothes, body shapes, etc.) which map their avatars into the associated roles. It is important to note that depending on the selected approach, the management policies on avatar appearance can be different.

![Figure 5.14: Role-play character avatars – a refugee female (left) and a male (right)](image-url)
The scenario characteristics relate to the EM-High and SR-High configuration of the user interactions. Students in this scenario have to follow a highly self-regulated environment interaction as constrained by their assigned role-plays while the VHD training environment enforces a set of strict policies on environment management to simulate the disaster and humanitarian aiding activities as in the real world. A selected set of policies from each major function area are shown in Table 5.8.

Each of the policies was examined using function network topology with Gephi to understand the functional interactions and conflicts for their implementations. For this scenario example a different area of policies from the above list – avatar mobility - is used for this discussion to explain the benefits the VHD staff obtained.

Since the objective was to let students practice as they would in a real environment the avatar mobility activities were subject to control by the VHD staff. The view of the VHD lecturers on avatar mobility (fly and teleport) is that if the students are given the opportunity to teleport or fly, they may not intuitively experience the time taken to resolve a dilemma situation or an emergency that arises in a far corner of the camp and the consequences of their delayed actions due to the physical constraints.
<table>
<thead>
<tr>
<th>Category</th>
<th>Management Policies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Land Mgt.</td>
<td>Students should not terraform the learning environment</td>
</tr>
<tr>
<td></td>
<td>Students should be placed in the required locations</td>
</tr>
<tr>
<td></td>
<td>Student must refrain from accessing restricted areas (if defined for a learning activity)</td>
</tr>
<tr>
<td>Avatar Activity Mgt.</td>
<td>Users should refrain from flying/teleporting to simulate real mobility</td>
</tr>
<tr>
<td></td>
<td>Students should follow the role-play guidelines and refrain from changing their avatar appearance</td>
</tr>
<tr>
<td></td>
<td>Students should not misbehave in the environment</td>
</tr>
<tr>
<td>User Mgt.</td>
<td>User names can be based on Role-plays</td>
</tr>
<tr>
<td></td>
<td>Users should be given appropriate privileges for Role-plays</td>
</tr>
<tr>
<td></td>
<td>Users may not change their given home locations</td>
</tr>
<tr>
<td>Content Mgt.</td>
<td>Students should not alter/move learning content</td>
</tr>
<tr>
<td></td>
<td>Students should not create content objects</td>
</tr>
<tr>
<td></td>
<td>Students should not alter/move the environment content</td>
</tr>
<tr>
<td>Group Mgt.</td>
<td>Students should not change settings of their assigned groups</td>
</tr>
<tr>
<td></td>
<td>Students should not change their assigned groups</td>
</tr>
<tr>
<td></td>
<td>Students should not alter group owned objects or land</td>
</tr>
</tbody>
</table>

**Table 5.8:** A set of policies for VHD use scenario 2 management

In order to experience and intuitively learn about the practical constraints that one could experience during aid working, students must follow the realistic mobility options such as walking, running or crawling. Similarly, they have to follow dedicated routes such as camp gates, aisles between the camp tents, and avoid barbed wire fences and other barriers. They can also learn about refugee mobility and the queuing strategies that they should use in case of an emergency need.
Fig. 5.16 shows the ego-network representing avatar teleport with a depth of 2 levels. Fig. 5.5 presents the same ego-network of avatar fly that VHD researchers used for their work. The ego-networks of each function indicate the related functions or settings that can be used to manipulate the student teleports between places and flying in the VHD Island. Depending on the various OpenSim user roles and VHD land ownerships such as estate owner, region owner and parcel owner, avatar mobility can be controlled within the owned land. Furthermore, the groups in OpenSim can affect the policy implementations on their lands. More challengingly, depending on the roles practiced, students may be given parcel ownership for roles such as Refugee Camp Officials or Lead Aid Workers on their camp installation parcels; moreover, they can be associated with a group defined for similar role categories.

In brief, for controlling avatar flying, the lecturer has to consider 4 functions that control user flying and 6 interactions between those. To control avatar teleport and associated settings, they have to consider 7 functions with 13 interactions. The behaviours of these interactions, as shown in by the colour coding, can be supportive or unsupportive for the required control. Moreover, the contextual implementation of the policy in OpenSim should be noted. For example, if the VHD learning session encourages flying then the VHD staff have to toggle the appropriate set of functional interaction (green links to be true, red links to be false, and orange links as needed) whereas to restrict flying, they have to toggle their selections for each function the other way around. The guidance tool was used to understand these options by the academics involved in running the VHD environment.
5.5.5 Scenario III – Training through Development

In this scenario of usage, the VHD project is arranged to facilitate research projects of honours and postgraduate students (SCQF levels 10 – 12 [198]). The main objective is to let students use the VHD environment as a platform for exploring their ideas about research into disaster management models and methodologies.

This scenario has SR-High characteristics (EM-High and EM-Low), since the students in the VHD environment must show a higher level of self-regulated learning behaviour. Depending on the learning task requirement, the land parcels can be common to every student and act as basic sandboxes. Students access parcels without ownership but are allowed to create content and build learning constructs, which corresponds to EM-High type management needs. On the other hand, if the learning activities that come into this scenario include environment change, land editing and land ownerships at the parcel level, students are given more flexibility in VHD. This arrangement follows the EM-Low type management policies. However, in both approaches, students have a narrow margin for misusing the opportunities, as they should complete their learning tasks with overall responsibility and positive engagement, which requires High-SR interactions. A selected set of example policies for this scenario are shown in Table 5.9.

<table>
<thead>
<tr>
<th>Category</th>
<th>Management Policies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Land Mgt.</td>
<td>Students may terraform the learning environment</td>
</tr>
<tr>
<td></td>
<td>Students may change their working locations</td>
</tr>
<tr>
<td></td>
<td>Student may change land settings on assigned lands</td>
</tr>
<tr>
<td>Avatar Activity Mgt.</td>
<td>Student mobility with no restrictions</td>
</tr>
<tr>
<td></td>
<td>Students should not distract others’ learning</td>
</tr>
<tr>
<td></td>
<td>Students should not misbehave in the environment</td>
</tr>
<tr>
<td>User Mgt.</td>
<td>User names should map real identities</td>
</tr>
<tr>
<td></td>
<td>Users should be given required role privileges</td>
</tr>
<tr>
<td></td>
<td>Users can change their given home locations</td>
</tr>
<tr>
<td>Content Mgt.</td>
<td>Students may alter/move learning content</td>
</tr>
<tr>
<td></td>
<td>Students may create content objects</td>
</tr>
<tr>
<td></td>
<td>Students may alter/move the environment content</td>
</tr>
<tr>
<td>Group Mgt.</td>
<td>Students may change settings of their assigned groups</td>
</tr>
<tr>
<td></td>
<td>Students should not change their assigned groups</td>
</tr>
<tr>
<td></td>
<td>Students may alter group owned objects or land</td>
</tr>
</tbody>
</table>

Table 5.9: A set of policies for VHD use Scenario 3 management
Comparatively, the scenario 1 of VHD use differs from this scenario. The first scenario allows the students to complete the quizzes in Sloodle, which then updates their Moodle records, and the rest of the student activities in-world are not accounted for their grades. In contrast, this scenario allows the students to explore almost all the benefits of using a MULE as a dynamic, engaging and user created learning platform. It also gives the students the opportunity to create their own scenarios in accordance with environmental, human and financial resource requirements. Thus, by engaging with the VHD tool, students are encouraged to reflect on the knowledge and understanding that they have gained throughout the module, which corresponds to higher levels of learning, while also making the student experiences about their learning more realistic.

![Image](image.png)

**Figure 5.17:** An example usage of an allocated land parcel with content and land alteration

Fig. 5.17 shows an example test case conducted by one of the VHD project researchers, to evaluate the guidance tool for help in configuring parcel ownerships, various land settings and content creation. The test case was based on a dilemma situation with a water supply through a natural source (river or lake) and how its geographical location can impact the choices of designing the refugee camp, supply routes and the external influences through the un-fenced pathways. Being close to a natural water source may be beneficial to the refugees, and assist the work of the aid workers but at the same time can introduce more threats for the security of the tenants and create health risks through waterborne disease epidemics. This is a challenging dilemma that a research student can examine with various solution models through exploratory implementations as test cases with the support of the given virtual space in OpenSim. The VHD researchers and lecturers tried various parcel and region level land management settings to test the impact of those with the assistance of the function network in Gephi that was provided.
For a detailed study about this scenario, land management (a more complex and broader management area) is selected as an example to evaluate the significant benefits that VHD staff obtained from using the OpenSim function network guidance. As per the requirements of this learning scenario, academics involved in the module had to provide land ownership, ideally at the parcel levels to the students with access rights for the land management.

Fig. 5.18 shows a section of the ego-network of land parcel management as part of the function category Near Field Spatial Management, obtained through the tool. The image shows the ego-network filtration depth at 3 levels indicating the overall complexity of the selected function area; in practice, sub-networks and depths of 1 and 2 were mainly used for VHD policy implementations. The ego-network indicates the related functions and settings that define a range of activities available for students inside their given parcels. With appropriate filter settings the network topology presented the ego-networks of 6 major functional areas associated with land parcels that facilitated user engagement in the environment. By becoming the owner of a land parcel, a student can explore these functions; depending on the policy, VHD researchers and lecturers decided the level of control over these functions. However, if the students are given land parcels for their research with tight control in place at the estate and regional levels, such provision may not facilitate the expected ILOs. In a complex situation like this which has ambiguity of EM control mechanisms the function network and tool support can significantly reduce the time taken to implement management policies and increase the accuracy of aligning those with the ILOs and TLAs. This benefit was highlighted and agreed by the VHD staff involved in the study.

Figure 5.18: A selection of a complex ego-network for land management
Depending on the policy implementation conditions (with parcel ownership or group owned land) VHD staff practiced controlling mechanisms at the estate and regional levels to prevent any unsuitable changes by students to the implemented policies through override or block of those proactively. Moreover, in this scenario, the VHD researchers suggested that even if the proactive policy mechanisms are not feasible due to the required level of flexibility (EM-Low arrangements), reactive strategies, such as restarting the region, disabling scripts, etc. can be quite helpful in an unforeseen event. Even for such usage, VHD staff felt that they had benefitted from the guidance tool since they can evaluate the situation and consider appropriate actions at run time, as the ego-networks are self-explanatory.

5.5.6 Overview of Tool Support for VHD Project Management

The VHD project provided multi-faceted benefits for the case study objectives since the intrinsic characteristics of the project incorporated multiple learning scenarios spanning across different policy implementation areas. Because of that, any inappropriate management policies were not used for evaluating the tool. The scenario-based policy implementations made the case evaluations as natural as they could be.

The challenge of managing learning activities in an OpenSim MULE by memorising all these settings is overwhelming. In the worst case scenario, with $m$ number of policies, each associating $n$ number of OpenSim functions, the lecturer has to consider $mn(n-1)$ number of functional interactions (where, $\{m, n \in \mathbb{N}\}$). The implementation of the policies that have been examined with VHD project would have been very challenging and unproductive without the guidance support provided as the complexity of the arrangement shows polynomial time characteristics.

The observations and the feedback received from academic and research staff involved with the VHD project indicated the positive and supportive nature of the OpenSim function network tool as a guidance mechanism for implementing their policies. They indicated that with the tool, they could put their policy implementations into practice with a higher efficiency and certainty, making the task more manageable. Moreover, the developer feedback validated the high usability of the user guidance support, which made it much easier to use the complex OpenSim function interaction topologies for policy implementations.

One of the important views was to extend the guidance support covering the introductory and advanced OpenSim activities. Some VHD staff indicated that it would have been complete if they had had some experience in-world prior to engage in the management activities. This is in fact one of the main objectives of this research; since academics from non-ICT domain may find it difficult to master OpenSim based MULE management it can be quite useful to provide support for that training need as well. The next chapter discusses further details about this.
5.6 Concluding remarks

This chapter presented work on user guidance for the management of an OpenSim based MULE. The result of a survey that examined the need for providing specific user support tools for managing OpenSim was discussed. An innovative and usable approach to depicting OpenSim function interrelationships was developed to support academics in implementing their management policies; the statistical verification of the tool accuracy and structural mappings were also discussed. Finally, the findings from a case study of a real educational project have been presented to demonstrate the contributions the user support tool made in managing an OpenSim based MULE. The next chapter presents the training regions developed for supporting academics and students in OpenSim based MULEs across a variety of levels.
6. An OpenSim Training Environment

6.1 Introduction

This chapter presents work on designing and developing an OpenSim training environment. The chapter initially discusses the findings from an experiment conducted to identify a suitable approach. Section 6.2 presents the experiment, data analysis and user feedback on approaches that were tried. Section 6.3 separates the specific training needs for MULE management from generic user training for basic OpenSim interactions. Two separate but similar OpenSim islands were designed for each purpose. Section 6.4 presents the development and implementation these two islands in OpenSim. Technical challenges faced during development are also discussed. Section 6.5 concludes the chapter.

6.2 An Evaluation of OpenSim User Training Methods

The user studies and student learning sessions that have been discussed throughout this thesis indicated the importance of user support for managed learning in OpenSim. For example, students often spend a significant amount of their available time for the learning activity getting familiar with the environment and the functions and mechanisms associated with the learning tasks in-world. For the learning activities with all of the university OpenSim MULEs, we have provided students and academics with a brief guidance document that covers most of the basic tasks they should be familiar with in the environment. This document includes their respective avatar username and password, and instructions for accessing the server. However, beyond the basic user settings it was found that if all information about OpenSim management is included in the user guidance document it would be too lengthy and cumbersome to use, as was observed in VHD project case study. Therefore, an investigation of alternative methods for user training was carried out. In particular, could the MUVE itself be used as a successful training method?

6.2.1 Document Based and Island Based User Support

User manuals and other guide documents are popular instruments for training new users of a software system. Quite often these documents have many screenshots of the application user interface which are used to steer a new user through sequential orders of actions. However, for complex scenarios of user interactions, these types of document can become unhelpfully lengthy and less intuitive. One of the possible approaches to this challenge is to use a video tutorial that captures a demonstration of the software interface. Likewise, if the system is large and complex, addressing the training needs of the entire system using video material can take a relatively larger user training time compared to documents with captured screen images of the key stages.

---

2 In fact, this approach is quite popular for specific functions on YouTube
For OpenSim basic user training needs we refer to the two approaches used by Second Life. Second Life has a user guide document [227] which covers basic avatar actions with the point-and-click UI and special key combinations which perform the same actions. The other method SL uses is to direct new avatars into a uniquely designed region known as Help Island. Help Island in SL is intended to be the first place an avatar interacts with the SL environment if they register through the standard process; an exception is, an institute can have its own SL registration interface through RegAPI in which new users can be directly located in desired SL regions thereby bypassing Help Island. Help Island provides training for basic functions and free content such as inventory items, body shapes and clothes for avatar customisation and opportunities to receive rewards in Linden$; these are used to motivate new users to actively engage in the Linden Labs economic model through the SL Marketplace.

There are no specific research data on evaluating Help Island compared with the user guide document for training new users. One of the major drawbacks when using Help Island for education is that its design explicitly prioritises commercial activity and entertainment use cases in SL. This can tempt some students to engage in game-like behaviour, which they may not have even imagined to be possible had they not been exposed to Help Island. Furthermore, there is no guarantee of a suitable and academically motivated group of users with High-SR behaviour to be met in Help Island when a new student enters SL; it can negatively affect the student shaping their behaviour towards Low-SR practices in learning activities. Because of these reasons, universities often bypass Help Island and locate students directly on their educational islands and provide them with the user guide document for basic tasks. For locally managed OpenSim based MULEs there is no Help Island for new users. Because of the similar functionality between SL and OpenSim and their shared client applications, OpenSim users can use the SL user guide document. However, further information such as OpenSim grid location URL and avatar user credentials have to be added to the document. The Open Virtual Worlds group at the University of St Andrews have been using a modified user guide, derived from the SL user guide for learning support activities across a range of OpenSim-based projects. Although that approach helped for login and access problems, students still showed an initial difficulty when interacting with the environment. It was evident that the document, although it includes the necessary information for completely new users in a detailed manner, is not effective for learning essential OpenSim interactions. Often, the first half of a laboratory session is used by students getting familiar with these OpenSim functions; this can take up the entire session for some students who may find it more challenging to map the training information from the guide onto the skills they practice in-world. Moreover, a greater number of support academics (lecturers, tutors, demonstrators) are usually required to help the students in building their confidence in using the learning environment during the lab session.

In SL with Help Island content, individual users train themselves without additional support mastering the various basic functions available at the client side. Also, the SL user guide and wiki have been popular places that SL users look at to clarify their doubts about the environment and how to achieve their preferred behaviours. Moreover, Second Life discontinued its in-world mentor program claiming that the Help Island is sufficient for the training needs of new users. Due to these observations and challenges in training students and academics in OpenSim, it was decided to develop in-world user support for OpenSim environments (compatible with SL). In order to set a baseline for evaluating the usefulness of the development it was decided to conduct a pre-test to compare these two approaches: a region similar to Help Island and user guide documents.
6.2.2 Pre Test to Examine a Suitable Approach

A prototype OpenSim region with basic information about avatar interactions in OpenSim was developed. The objective was to examine user performance on completing a given set of tasks in-world after visiting the training island. To compare the performance, a controlled study was carried out using the SL user guide. For the accuracy of the results the same training content from the user guide was selected and put on display boards in-world. This way the participants of the two samples, i.e. experiment sample and control sample, will see the same training content but in two different mediums, one in OpenSim and the other on a printed document. Because of this a performance variance between the two groups can be more likely due to the different approach of the training and less due to the content or tasks. Fig. 6.1 shows an aerial view of the prototype island with scattered training content displays and a close up view of the first display the avatars see when they enter the island; it helps avatars to explore the environment. Lines of trees were used to separate different categories adding a more immersive flavour into the training experience instead of forcing them to follow a sequential path as happens in the printed document.

![Figure 6.1: Pre-test Prototype Island and a training guide at the arrival point](image)

10 voluntary participants with no prior experience were identified for this pre study and were divided into two groups, 5 participants for each were selected randomly: a group to access the training island and the other group to refer to the user guide. The two groups were named Group-Doc (the group that used the user guide document) and Group-Island (the one that used the island for training). Participants were encoded according to their groups: i.e., Group-Doc {User1-GDOC, User2-GDOC, User3-GDOC, User4-GDOC, User5-GDOC} and Group-Island {User1-GIsland, User2-GIsland, User3-GIsland, User4-GIsland, User5-GIsland}. The experimental set-up allocated each participant a 30 minute session – 15 minutes to get familiar with the environment using the training material provided (either user guide or training island) and the rest of the time (15 minutes) to follow a set of tasks on a separate island. Individual user sessions were essential since we examined user performance; this arrangement provided a uniform test environment with equal load on the server minimising errors due to variations of server and client performance.
To compare the user performance between the two groups five of the most basic tasks that any avatar should be able to perform confidently for successful engagement with the environment were selected. Walking and flying were selected as the two important tasks that enable avatars to explore the environment. Object creation and being able to perform basic editing on the created object were also selected as essential skills. Finally, a task with basic communication inside OpenSim - IM and chat was selected. The task scenario in brief was as follows:

When you arrive at Test Island you will be located at a starting place. Please complete the following tasks as soon as possible.

1. Your first task is to walk along the road until the end of the road. Please make sure you walk on the middle of the path marked by white dashes on the black tarmac.

2. At the end of this road you will see a signpost asking you to fly over the sea to the island. Please do so and land on the exact location marked with a signpost on the island.

3. Now create an object (cube) on the ground and follow the instructions given on the relevant signpost to edit your cube.

4. Start walking on the second road until you reach its end.

5. Perform the tasks on IM and chat as instructed by the signpost at the end of the road.

It is important that you try to complete these tasks as accurately as possible and as soon as possible for the evaluation requirements.

For task 3, object edit was required to reshape the cube with given dimensions (x, y, z with 2 decimal place accuracy), reposition it on a given location (x, y, z coordinates, with 2 decimal place accuracy) and re-colour only a selected face (the top face) of the object in red. Task 5 included sending an IM with the given text to a previously added friend in the avatar friend list – Test User, and publish the text. Task completed using the chat channel in-world. At the beginning of each task these specifications were displayed to the participants using signposts at the relevant locations.

Another advantage of this arrangement is that the way users perform these tasks can be an indirect indicator of user self-regulation level, i.e., by trying to follow a specifically given set of instructions and tasks with constraints.

The task environment (Test Island) was designed in accordance with the task list considering the required performance monitors. Fig. 6.2 shows the map of the planned island. The red colour circles represent the sensor locations to capture the avatar movement times during the experiment.
To make sure the participants do not fly when they are meant to be walking, the region was divided into two parcels: the one shown on the map has been set up to allow avatar flying, and the rest of the land (the other parcel) has restricted settings for flying. A set of sensor objects were deployed as shown in Fig. 6.2. The LSL sensor function `llSensorRepeat()` was used to implement the sensor functionality with parameters of 1.0m range from the sensor and time interval of 0.1 seconds to repeat the scanning. When an avatar is detected by the sensor it triggers the required functionality as defined in the event call. The LSL `llGetTimestamp()` function was used to obtain the timestamp of the avatar detection, which is within 1ms accuracy. Angular bends on the path are used to evaluate avatar movement performance. Fig 6.3 shows the actual implementation of the island with the designed layout. Sensor scripts were embedded in cubic prims of 0.5m of size. To conceal the sensor locations, these cubic prims were made transparent and were half-buried in the middle of the path at planned locations making all the sensors deployed at the same height (z-axis) along the path. The sensors were not distributed in equal distances from each other but according to the path segment and the bends. Timestamps of object creation, object edit and communication (IM and chat) were traced through the object profile and island chat history.

![Design map of Test Island for the pre-test evaluation](image)

*Figure 6.2: Design map of Test Island for the pre-test evaluation*
6.2.3 Results and Analysis

The first analysis was performed on the times that each user spent in reaching the first set of sensors on the walking path. The distributions of the user times at each sensor are shown in box plot graphs in Fig. 6.4 (for Group-Island) and Fig. 6.5 (for Group-Doc). The distributions generally indicate a higher amount of time for Group-Doc than Group-Island. Also the variance of times per sensor is considerably higher for users from Group-Doc compared to Group-Island.

The distribution of average times for each sensor for the two groups is shown in Fig. 6.6. A clear difference in the mean times can be seen during the early stage of walking. Until the 5th sensor the participants from Group-Doc showed much longer times to reach sensors while walking on the middle of the path. From the 5th sensor to the 14th sensor the time distributions are somewhat similar in their trends but with a reasonable time difference. This may have been due to the fact that the Group-Doc users gained an opportunity to train further during their walk within the first few sensors; yet, the time taken is higher compared to the other group, although the difference is in the range of a few seconds.
For the 15th sensor, a further difference is noted in between the two sample trends with a deviation. One of the important observations during the experiment helped to explain the reason for this. The users from Group-Doc often got slowed down at the end of the walk trying to figure out of how to fly for the next phase of the task. In contrast the users from Group-Island confidently finished walking and started their flying without hesitation. Both groups showed increased times for passing the sensors 6, 10 and 13. These sensors are located at the bends of the path that require extra avatar control effort to maintain their trajectory.
A time comparison of avatar flying between the two groups is shown in Fig. 6.7. The Group-Island users showed lower times compared to the users from the other group; we use pairwise comparison based on the user codes assigned (i.e., User1-G\text{DOC} with User1-G\text{Island}; User2-G\text{DOC} with User2-G\text{Island}; and so on) only for presentation purposes (the figures 6.7 - 6.10); we do not infer on pairwise comparison. Moreover, the users of Group-Island show a lower variance in time against the other group; in fact, it was the expected result as the flying path was a straight line, these users showed more confidence in doing their task quickly and on a straight path while the users who trained from the guide document showed some tremble in all directions during their flying.
Fig. 6.8 and 6.9 show the times spent by each user for object creation and the required editing tasks, respectively. For both tasks, all users from Group-Island showed much less time than the other group of users. For object creation the time difference was slightly lower since it comprised a single task. However, for object editing, which included a few tasks, the time difference between the two groups is high. Also the variance of times is lower with users from Group-Island. For example, most Group-Island users carried out the object resize and re-position task in a single try through typing the exact figures, while the other group (Group-Doc) tried to drag and resize using the mouse showing a poorer understanding of the system which resulted in a trial and error approach.

**Figure 6.7:** Average time taken by the two groups – Task Flying
At the end of the object-editing task users were asked to move along the exit path and carry out IM and chat to finish the task session. Fig.6.10 shows the times used by each participant in the two groups. User3-G_{Doc} and User4-G_{Doc} show a lesser time than User5-G_{Island}, which is an interesting observation. However, when closely examined it was identified that these two users from the Doc group did not
complete the IM messaging and abandoned it. Further details about task completion will be discussed later.

Table 6.1 summarises the average times spent by each group for the four tasks recently discussed. Object edit shows the highest difference between the times while IM and chat shows the lowest difference. In fact, IM and chat are relatively generic tasks that are implemented according to similar functionalities that are used in popular IM & chat applications. On the other hand, object edit is an OpenSim system specific functionality that requires a certain level of know-how to practice well. These may have been the reasons for this result; nevertheless, it implies that when the tasks are complex we can expect a further difference in times for the two groups. Table 6.2 shows the descriptive statistics of times taken by the two groups to complete the entire task. A clear difference in the total mean time can be observed between the groups; however the times were compared with a reference scale for an accurate interpretation. An expert user with extensive OpenSim/SL experience was asked to complete the experiment tasks in a similar manner (as soon as possible completion) for 3 times. The mean total time was 259.41s, which is considered as a reference value to compare the test results of the two groups. In addition to the considerable difference in the total times taken by the two groups to complete the task, a clear difference between the total times can be observed with reference to the expert user time. Group-Island showed a +31.597s of overhead time (12.18%) while Group-Doc had +124.035s of overhead time (47.81%). It is observed that the participants trained using the training island have been able to rapidly apply their skills gained through the training session for their activities compared to the group that trained with the document. In summary the participants that used the island showed

**Figure 6.10**: Average time taken by the two groups – Task IM and finishing
competencies closer to the expert user in completing the given tasks than the participants from the document group.

<table>
<thead>
<tr>
<th>Group</th>
<th>Flying (s)</th>
<th>Object Creation (s)</th>
<th>Object Edit (s)</th>
<th>IM and Finish (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group - DOC</td>
<td>36.78</td>
<td>25.90</td>
<td>140.04</td>
<td>70.75</td>
</tr>
<tr>
<td>Group - Island</td>
<td>26.16</td>
<td>18.91</td>
<td>111.49</td>
<td>64.70</td>
</tr>
</tbody>
</table>

**Table 6.1:** Average time comparison for the four task phases

<table>
<thead>
<tr>
<th>Group</th>
<th>Mean Total Time (s)</th>
<th>Std. Deviation</th>
<th>Std. Error of Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>G_{Island}</td>
<td>291.07</td>
<td>5.76</td>
<td>2.58</td>
</tr>
<tr>
<td>G_{Doc}</td>
<td>383.44</td>
<td>11.07</td>
<td>4.95</td>
</tr>
</tbody>
</table>

**Table 6.2:** Descriptive statistics of the entire task for the two groups

There was a noticeable difference in the completion of tasks between the groups. Table 6.3 summarises the key observations about the task completion. Only one participant from the Group-Doc could complete all the tasks as expected. In particular, the complex tasks such as object manipulation seem to be the most challenging. In contrast, only one participant from the Group-Island use failed to complete all the tasks. However, all the participants successfully engaged in the activities of walking, flying and chat messaging.

<table>
<thead>
<tr>
<th>Participant</th>
<th>Key observations about the completion of the tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>User1-G_{Doc}</td>
<td>No object surface colour change, object positioning is not accurate</td>
</tr>
<tr>
<td>User2-G_{Doc}</td>
<td>No object surface colour change</td>
</tr>
<tr>
<td>User3-G_{Doc}</td>
<td>No object surface colour change, object size &amp; object position are not accurate, No IM message</td>
</tr>
<tr>
<td>User4-G_{Doc}</td>
<td>No IM message, the whole object is red, object position is not accurate</td>
</tr>
<tr>
<td>User5-G_{Doc}</td>
<td>All tasks were done</td>
</tr>
<tr>
<td>User1-G_{Island}</td>
<td>All tasks were done</td>
</tr>
<tr>
<td>User2-G_{Island}</td>
<td>All tasks were done</td>
</tr>
<tr>
<td>User3-G_{Island}</td>
<td>No object surface colour change</td>
</tr>
<tr>
<td>User4-G_{Island}</td>
<td>All tasks were done</td>
</tr>
<tr>
<td>User5-G_{Island}</td>
<td>All tasks were done</td>
</tr>
</tbody>
</table>

**Table 6.3:** Key observations of each user on the task engagement
6.2.4 Participant Feedback

A brief questionnaire was used at the end to examine the participant opinions. The first question asks about the support they received from the user guidance approach and the second question asks their opinion about the alternative approach (either document or island) instead of what they had. This alternative arrangement of questions helps the participants to compare their experience and respond highlighting their preferred method of receiving OpenSim training. The questions were designed with Likert scale answers at 5 levels: Strongly Disagree (1), Disagree (2), Neither Agree nor Disagree (3), Agree (4) and Strongly Agree (5).

The following two questions were asked from the Group–Doc participants; their responses are shown in Fig. 6.11

Q1 – The guidance document helped me to complete the tasks inside the MUVE comfortably

Q2 – A training island would have been a better and more usable method for me to train myself for the tasks

![Graph showing responses to Q1 and Q2]

**Figure 6.11:** Group-Doc user feedback for the questionnaire

All of the participants from Group-Doc are of the view that it would have been a better approach to use a MUVE for their training needs. Interestingly, they have indicated that there is a certain level of challenge for them to map the information they have learnt from the document into the OpenSim context. Acknowledging the fact that they have learnt some information from the user guidance document, these participants showed their doubts about a document for training complex OpenSim tasks.
The following two questions were asked from the Group–Island participants and their responses are shown in the graph in Fig.6.12.

Q1 – The training island helped me to complete the tasks inside the MUVE comfortably
Q2 – A user guidance document would have been a better and more usable method for me to train myself for the tasks

![Bar chart showing responses to Q1 and Q2](image)

**Figure 6.12:** Group-Island user feedback for the questionnaire

All of the participants from Group-Island indicated that they had benefited by the training provided through the island. The majority strongly agreed in this view and also disagreed with using a user guide document instead.

The training island used in this pre-test was developed as a test environment mainly following similar Help Island layouts in SL. An improvement suggested by these participants was to design a more formal educational environment layout. This was an interesting view; it was thought that having a campus like training environment would encourage students to self-regulate during the early stages of their training.

### 6.2.5 Overview of the Findings

Briefly, as discussed above, the participants that tried the user guide document showed a certain level of difficulty in completing some of the relatively complex tasks. It was further observed that participants who had experienced the training island showed more confidence in completing the given task in comparison to the other sample. From the user feedback all participants indicated that using a training island can be a better and usable option.
The percentage time difference between the groups with respect to the expert user suggests that document based training may require further overhead time for complex tasks. In fact, this was evident with students during the laboratory sessions on learning in a MULE where they took substantial portions of their time to learn the basic interactions. The policy implementation mechanisms with the available OpenSim functions are more complex when compared to the tasks included in this experiment. Therefore, it may be realistic to assume that a user guide document may not be a good option for training complex OpenSim management functions.

Also, the study findings suggest that a higher level of user self-regulation was present in the group that used the island compared to the document users with respect to their task completion; the island users showed better ability to perform the tasks as they were required, comparatively.

Considering all these findings it was decided to develop a suitable training environment that can be used as a common platform for user training needs for OpenSim supported MULEs. The environment can act as the first place for new users allowing them to train before starting their educational activities. The next section describes the design of this environment.

### 6.3 Training Environment Design

A training environment designed for OpenSim users can exploit the better points of Linden Labs’ Help Island concept. However, the guide that we provide should also support management aspects of OpenSim functions that help users to achieve High-SR and High-EM states. Another desirable feature identified during the pre-test was to design and introduce a more university-like virtual space as the training environment. That would also help overcome the drawback of Help Island which orients its users towards entertainment and gaming rather than educational practice.

One of the recurring findings during user evaluations and laboratory sessions with OpenSim is that students often get distracted by new skills. Usually they are quite keen on creating content objects, land editing (terraforming) and changing avatar appearance. One of the easiest ways to control students doing this would be to prohibit some of these actions through system functions at region and estate levels. Although such practice would give the expected result, students can get confused by that, because when they learn about something new and try to do it, it is not possible in-world. Moreover, the design of SL and OpenSim does not include proper error messages or notifications to inform users why some attempted actions do not work. Without feedback new users can easily doubt the reliability of the training materials and content being presented. Therefore, a sandbox area with full permission was provided to allow any activity while the main part of the training area had restrictions. In addition to the sandbox a communal area for welcome arriving users was designed with basic and essential information and directional guides to the various places on the island.

A few more places were introduced to the training environment to enhance the training experience. An information centre was designed as the first place of user interaction, simulating real world information provision in public locations. To familiarise the users with the educational activities that they would practice in a MULE a special unit for user discussions was incorporated into the training environment. A cinematic enclosure was also included for video streaming of additional training support. Although
the training centres for specific function categories can host video material, a specially designated location for displaying video content of a longer duration may help the users have a more intuitive experience. An open forum area was also included to present smart tips and efficient practices such as hot keys for using OpenSim. Fig. 6.13 shows the overview plan of the training environment.

The scale of these virtual buildings has to be designed according to the region scale (256mx256m) and avatar size. Usually, the default OpenSim avatar is 1.88m in height and can range from 1.61m to 2.12m. However, due to the physics engine simulation criterion if we implement the correct scale ratio of doors and building sizes to the avatar sizes as they are in the real world, difficulties in avatar movement can be experienced. Accordingly, all the virtual reconstruction projects in OpenSim at the University of St Andrews have been developed with the building scales ranging from x1.5 to x2.0 actual size of those in the real world.

![Overview plan of the training environment](image)

**Figure 6.13:** The design architecture of Training Islands

Training areas and content for this environment include both basic user training and advanced management training needs involving a large number of unique functions and their complex interrelationships. Furthermore, the ways in which a user can try these functions through the client UI also needs to be provided as part of the training. In fact, it can be considered as an effective and convenient way to tell a user about how to practice these functions, since the UI widgets are the only available mechanism to manage an OpenSim MULE from the client side.
Moreover, the need for separating basic users (usually students) from advanced users (usually academics and module coordinators) was also identified for certain educational scenarios. In particular, if the OpenSim component of a module has a low weight then asking students to learn advanced functions may be unnecessary. Furthermore, a substantial amount of the available time for a session may have to be spent on training if a common region with all the information is used. Considering the variations in these possible needs it was decided to develop two similar regions as the training environment. The first region, named Introduction Island, is dedicated for all users looking for basic training needs. The other region, named Management Island, contains training material for advanced functions.

Introduction Island and Management Island were designed and planned with a similar layout using the same set of training buildings and educational units. Further details on their specific designs are discussed in the next section.

6.3.1 Introduction Island Design

The main objective of this island is to act as the first place for new user interaction. In that respect, Introduction Island is similar to SL Help Island except for the explicit focus on educational activities using the University’s MULE infrastructure. A new user is directed to this island for basic training as soon as they register with the University’s OpenSim grid. The arrival area of the island features avatar mobility training. At the end of a training session they can either teleport to their target educational island or to the hub area, which interconnects all the major educational projects in the grid through teleport links.

Taking into account that the users of this island are often experiencing their first OpenSim interaction, special design considerations were incorporated. SL Help Island provides avatar mobility challenges and exploration tasks to progress through its game like layout, which was thought to be an overhead burden to some students demotivating them to engage in the educational tasks. Another consideration was the uniformity of avatar interactions, i.e. a particular effort was made to minimise the time variance of different users (students) to complete their training in this island. Therefore, the avatar exploration challenges due to terrain conditions, mobility paths and interactive activities were minimised so that an average user would be able to complete the knowledge exploration in 20 – 30 minutes. The objective was that if a user takes more time to complete the exploration it would be mainly due to the user’s wish to spend more time on the island but not due to difficulties in exploration or accessing the content provided.

To achieve these requirements an island with a flat terrain (22m height, 1m above the sea level) was designed and continued at the same level except for a river for aesthetic reasons. Moreover, the training centres were designed as educational buildings with easy access including properly angled stairs for avatar walking without the need to fly to access the upper floors. The pathways that access these training centres and other buildings on the island were also designed in an easily accessible manner with straight paths and directional guides including teleportation facilities. The following specific training centres were planned to accommodate the training needs of new users.
**Information Centre**

Information Centre provides basic knowledge about what MUVEs (OpenSim) are and how they help provide an engaging and attractive learning opportunity. The MULE infrastructure is briefly introduced with reference to possible MUVE usages. It also includes basic information about the client UI widgets, their usefulness and how they help avatar activities.

**Avatar Mobility Tasks**

This centre provides the training required for avatar mobility needs. Basic movements such as flying, walking, running, crawling, sidewalks and teleportation are explained. Camera controls are also included. Sometimes, avatars find it quite easy to move around if they can have a flexible view with different angles over the island. Because of the importance of giving this training as early as possible, it was decided to locate it next to Information Centre.

**Avatar Appearance Change**

This centre provides information about possible methods to locate various avatar body parts and wearable content in the avatar inventory and how to use and save these contents.

**Content Object Construction**

This centre provides the information about basic skills in content creation and associated tasks. Details about basic prim shapes, prim specific editing activities and sculptured prim construction and modification are included. Information about how to place a content object at a specific location, modifying it with textures, setting physical properties and the use of inventory are explained in this centre.

**Basic Content Management**

Content object management can be practiced at two levels: i.e., content owner level and content management at virtual environment level. At this training centre in Introduction Island the necessary information to manage content owned by users is discussed. Basic information about setting permissions for individual components in an object, how to script, basic scripting information and accessing advanced permission types on different roles is discussed.
Avatar Communication

This centre includes training information about basic avatar communication methods such as IM, chat and voice streaming. How to use the client UI widgets to chat in text openly and how to use IM on the friend list are included. The common option to select avatars from a given proximity range is also covered in this centre. Settings for voice communication, how to perform and customise gestures, communications with friends, and communication options available as a group member are included.

Land Editing

If the land modifications are allowed users can perform these tasks even without the ownership of land. Therefore, appropriate training helps them to improve their self-regulation. Strength, type and area to apply on terrain edit are important information that users should be aware before they commit any terrain change as the reversal of a terrain change is rather difficult. Also, the available methods to obtain information about the land parcels that users are located in are also included in this centre.

Other places

These generic places include miscellaneous materials and activities for the skill development of users. The discussion rooms are designed with suitable seating and a layout to provide a tutorial or small group teaching session. The Cinema is designed to introduce mainly the large lecture theatre arrangement with the possibility of video display. The Open Forum hosts quick and easy user interactive options as a summative overview of the possible user actions. Also its location near the sandbox can help users to access this information conveniently if they wanted to refresh their knowledge while using the sandbox. The Sandbox is a land area designed with literally no restrictions (except avatar pushing) on avatar actions. It is important to mention that these places are designed to be similar in both islands.

6.3.2 Management Island Design

The design for this island is quite similar to what we have discussed in the previous section on Introduction Island. The Sandbox area and Open Forum are identical but Discussion Rooms and The Cinema provided a different perspective to the users since the main user community of this island is academics. Discussion Rooms and The Cinema include content aimed at the management of OpenSim; importantly the designs, system limitations on audio and video display and layout options are expected to make the academics aware of the challenges they would encounter and possible approaches to manage their teaching and learner support in-world. Apart from these places the following major categories of training centres are included to provide necessary information for managing OpenSim MULEs.
Training for Land Management

To include the three levels of 3D environment (land) management training content of the near-field (parcels), estate (estate & region) and administrator levels, three training centres were designed. Functions on land ownership management, land access management, control and administration of script execution, managing user terraforming and land editing, and administrator tools (God tools) are included, among other related topics. The complex interrelationships between various levels and functions are also discussed in these centres.

Training for Content Management

Considering the required content to be included it was decided to use two training centres explaining the complex content management scenarios such as cyclic permission loss, fair ownership, and composite permission management. Based on the training needs for managing 3D content objects, these two places additionally provide content manipulation tasks highlighting best practice and complex functional interrelationships.

Training for Group Management

Information about various group functions, their corresponding categories and default settings with roles are discussed in this centre. Teachers and course administrators should be able to transform available OpenSim group abilities and group roles to support educational needs, as applicable; this training centre is designed to support that need.

Training for Avatar Activity Management

This training centre provides information about how to manage avatar activities. The academics should be aware of the limitations they have on controlling these functions to manage the learning activities. Moreover, the training content is also designed to show the possible misuses and available solutions. Functional complexities and interrelationships are also discussed in this centre.

Training for User Management

This training centre explains the functions and best practices for user registration, roles and ownerships, privilege & access management and training for suitable user management, e.g., user (avatar) naming. Moreover, some of the important administration functions and commands are also included in this centre.
With these designs and training content for the two islands the Training Environment was developed in OpenSim. It was also decided to brand the islands, training places and content with the University of St Andrews logo to make this the common training place of the university OpenSim grid and the MULE infrastructure. The details about the development of these islands are discussed in the next section.

6.4 Training Environment Development

The developments and implementations were followed in accordance with the design considerations of the two islands. Basic prim objects, scripts and textures are used. To minimise the prim object load on a region special features such as multiple textures and texture repeats were used for the required appearance. Certain special implementation practices were followed as mentioned in the following section. Section 6.4.2 explains the teleport model and implementation details.

6.4.1 Special Implementation Practices

One of the challenging situations with new users is that they usually reach the boundary edge of the region without notice and experience certain unpleasant avatar movements or get into an infinitely submerged location freezing the avatar. Client programmes often fail to rectify this problem and users have to close the application and re-login to the server in extreme situations. This was noticed as a serious problem for some students, which reduced their trust in the usability of the learning environment. To overcome this challenge, a unique method of putting virtual fences between the region boundary and the adjacent sea was used. Fig. 6.14 shows this innovative approach. Not only does it give a meaningful view to the users, but also serves as a unique solution to prevent users getting caught up in movement problems at the edge of the region. By making the image an embedded fence non-phantom users are prevented from going through the fence.

Another interesting thought incorporated in developing The Cinema was to simulate real world large scale cinemas and lecture theatres. A ramped floor with suitable seating giving the students a realistic view was used. Although this is not essential if the user views with suitable camera controls to focus in, it might help new students to experience the immersive nature. Fig. 6.14 right inset shows the implementation of seating with ramped floor.
**Figure 6.14:** Special development practices: region border tree fence (left) and the ramped seating for the cinema (right)

Fig. 6.15 shows the campus like look and feel. All these objects and the environment were developed as part of this research except for a small number of image textures and decorative content used under the Creative Commons licence. Ease of access and user familiarity has been incorporated using a common structure for all the training centres.

**Figure 6.15:** Campus view and developed training centres

The open access centre forum shown in Fig. 6.15 provides an overview of the training places and their locations on the island, and relevant teleport links. It is implemented as a hub connecting all other places on the island.

To display training content and screen images of a client UI, a common thematic design of poster holders was used throughout the training centres. These poster boards are loaded with appropriate training content, text, and UI details as high-resolution images on the prims as textures. For convenient visuals, both sides were loaded with the texture image relevant to the topic discussed on that holder. To add Web-based text resource or video the Media on Prim (MoP) feature of OpenSim (version 7.2.1 and above) was used.
A potential problem with MoP is that users can change the source URI. For example, if a webpage is set as the resource, an interactive option can let the users access a web link or a web widget loading a new resource as MoP. Alternatively they can type a new URI in the default address bar, which appears on top of the MoP. To overcome this challenge a workaround solution using a combination of settings available in MoP was followed. First the control bar was disabled to the role Everyone {resident, visitor} and then the content URI was added to the allowed list making that URI the only authorised source for MoP display. By setting this, even if a user manages to click or select a different resource location, the MoP function will roll-back to its default URI, which is the intended media resource. For displaying multiple videos a video channel can be used. Fig. 6.16 shows a video content display using MoP.

6.4.2 Teleporting between Training Places

It was designed to provide teleport links throughout common areas and training centres for easy travelling. One of the challenges faced during the development of these regions was that having too many teleport links can confuse the user at early stages; importantly, a certain sequential flow of visiting centres in an orderly manner was designed according to their dependencies. These are presented at the Information Centre; the training centres too are located in a way which encourages a flow of training. However, if teleport access is provided to each place then users can practice a different order of visiting places, missing the planned paths of learning. To overcome this and for clear mobility a teleport model for both regions was developed (shown in Fig. 6.17). According to the model only the Forum and the Information Centre allow users to freely teleport wherever they wish; the Information Centre however encourages the flow of training to be followed. On the other hand, all the training centres and other places of the island (Discussion Rooms, The Cinema, Open Forum, and Sandbox) provide teleport links only to Centre Forum and Information Centre. This way the users always get redirected to the training flow. Teleport links in Centre Forum act as the links from a teleportation hub giving flexibility to visit places in user preference.
Development of these teleport links followed a usable design. Deploying a vast number of separate objects as signposts each embedding a teleport link can be the easiest way to implement a teleportation network, yet such an approach would obviously fill-up the space and make the use of those links a difficult task. On the other hand, if we embed all the teleport links in a single object, there should be a user action to select the place they wish to visit. At the same time, places should be given meaningful names to identify them easily. Moreover, these text expressions (place names) should be visible for a reasonable distance within avatar control. Taking all these requirements into account an innovative approach was followed; i.e., signposts and hoardings with suitable physical dimensions are first used as the base of the teleport links. Properly designed textures simulating real world directional guides are then prepared with the relevant expressions and loaded onto the selected faces of those boards.

To add the teleport functionality transparent cuboids were used. These cuboids are created with the exact size required to display the relevant teleport phrase; the transparent body does not affect the visibility of those texts. For teleport functionality through scripts in OpenSim, the OSSl version of the LSL `llTeleportAgent(args)` function has to be used. The OSSl equivalent is `osTeleportAgent(args)` and provides the same functionality if required server settings are met. That is, in OpenSim `osTeleportAgent(args)` is considered as a function that requires system administrator level intervention through setting the appropriate level of server side threat-levels. Arguments of the function are used to provide location information as a 3D vector (`vector target`)
and also the avatar orientation (the direction avatar sees) upon the arrival at the location (vector lookat) at the end of the teleport session. This lookat parameter helps the avatars to be faced at the entrance of the training place making their visit convenient. Fig. 6.18 shows some of the completed teleport boards on the islands.

![Implemented teleport links and locations](image)

Figure 6.18: Implemented teleport links and locations

### 6.4.3 Islands Review

The islands developed include spacious paths and open areas to facilitate unimpeded walking and flying if the users wish. Fig. 6.19 shows the final appearance of a training centre (left image) and Information Centre (right image) including some of the exterior designs and open spaces. Some of the textures are obtained from free and open use OpenSim content sources under Creative Commons licences and are used according to their terms. The other textures are either already available in the OpenSim default texture set or were developed as part of this work.
Fig. 6.19: Developed buildings: a training centre (left) and Information Centre (right)

Fig. 6.20 shows the Cinema and Discussion Rooms (left image), and the interior of a Discussion Room (right image). Discussion Rooms provide the opportunity for having a small group training session in-world as part of a guided training session. The land area of these discussion rooms was separated as a parcel and allows voice and audio streaming for communication of content delivery. Apart from having a suitable interior for group-based training and discussion it was also provided with empty content display boards for displaying video or other media content as suitable for the session.

Developed Open Forum (Fig.6.21 – left inset) with the supporting content for using client side hot-keys and the sandbox area with nearby land modifications (Fig.6.21 – right inset) are shown in Fig.6.21. The Sandbox area was separated from the mainland using a river to help users identify the boundary of the land; the environment management controls have been set differently to the main land by making the sandbox area a separate parcel.

Fig. 6.20: Developed buildings: Cinema and discussion rooms (left) and the interior of a discussion room (right)

Fig. 6.22 shows various training contents and arrangements used in the completed training centres. Some of the training materials are based on the UI functions of OpenSim clients to make users capable of rapid utilisation of their knowledge about using and/or managing the environment as a need arises.
Fig. 6.23 shows examples of tips and guidance notes to overcome content management related problems. It also shows the scale of these content holders for a clear look. Also, the auto scaling with zoom is enabled in media content for users to have a closer look without getting near the display if they wish to.

One of the challenges faced during the content preparation was integrating the Gephi analyser with OpenSim. Although the topology data in DOT scripts can be used, depicting the visual interface inside OpenSim would require a significant level of modification to the core code modules of OpenSim as the architectural designs of the two applications are substantially different. Such modifications would require using a non-standard OpenSim version in isolation for the sole reason of depicting this content. This would mean keeping two different OpenSim grids, one for this training and the other for the rest of the MULEs in the university. This was not considered effective practice. Also, a modified version would have to be maintained throughout new releases of OpenSim for backward compatibility and interoperability with the rest of the grids.
To overcome this challenge it was decided to show the captured outputs of the most common and frequently encountered scenarios of OpenSim management. Another advantage of this approach was that the guidance notes and tips with each ego-network scenario can be included to help the users, which was not possible with Gephi. Fig. 6.24 shows a set of such networks in a training centre with additional supporting notes. However, the topology library and link to the tool is only provided for users with advanced OpenSim management needs, and only after they have completed their training in Management Island.

Fig. 6.25 shows the aerial view of completed Introduction Island and Fig. 6.26 shows Management Island.
Figure 6.25: Aerial view of completed Introduction Island

Figure 6.26: Aerial view of completed Management Island
6.5 Concluding Remarks

This chapter has presented completed development work which implemented a training environment for OpenSim users. First, the chapter discussed the findings of a pre-test conducted to evaluate two widely used approaches to user training in SL and OpenSim – user guide documents and in-world help. The study findings strongly suggested designing two in-world training region for new users, and introductory and an advanced one. The chapter presented the design, development and certain special practices that were incorporated in preparing these training environments. The next chapter presents the evaluation of the advanced training environment as a means of supporting users in OpenSim management for educational purposes.
7. Evaluation of User Support for Managing OpenSim

7.1 Introduction

This chapter presents the evaluation of the training environment for managing OpenSim MULEs. Section 7.2 describes the experimental framework which includes OpenSim based MULE management tasks, MCQ questions and a points scheme. Section 7.3 presents the experimental island prepared to conduct the evaluations. Section 7.4 compares user performance between two groups for the given tasks; this analysis includes evaluation of usability and the (perceived) educational value of the developed environments. Section 7.5 presents user feedback about the training and experiment session. Section 7.6 discusses the case study evaluating further improvements made to the training islands. Section 7.7 discusses findings and limitations; Section 7.8 concludes the chapter.

7.2 Evaluation Scenario

The objective of this evaluation is to examine the unique user support provided by the training environment for managing OpenSim MULEs.

7.2.1 Experiment Setup and Sample Selection

This experiment was carried out with two samples, control and experiment. The experiment population consisted of academic staff, teaching and research fellows, and PG Tutors. The selection of these institutional roles was mainly based on the management tasks they typically practice in academic environments. At the same time, in the institutional context there can be instances where people play multiple roles and also may interchange when a need arises. Therefore, considering these possibilities these types of roles were taken as a single participant population without any role-based discretion for the objectives of the experiment.

The second important criterion to define test samples was to select only the participants that have not had previous experience of using OpenSim or SL. Therefore, following the suggestions of [201, 205] it was planned to set the samples to cater for: an Anticipated effect size (Cohen’s d) = 0.75 [205], desired statistical power level (1-β) = 0.8, and probability level α = 0.05, to perform statistical analysis. For these conditions the minimum sample size required is 30 participants per group (60 for the total). 70 participants were aimed for (35 per group) as the target total sample size, leaving enough space to eliminate any outliers or inaccurate data, if present.
Experiment setup was planned as a 1 hour session per participant allowing for a 40 minute training period and a 20 minute task session with feedback. For the experiment sample the starting place was Introduction Island for 20 minutes and then Management Island for another 20 minutes. The control group used Introduction Island for 40 minutes as the training environment, making the training time equal for the two groups. After 40 minutes of training both groups were given the task scenario and the participants were teleported to Experiment Island to perform the required tasks for the remaining 20 minutes. A strict procedure was followed to stop users at the end of 20 minutes (i.e., at the end of the one hour session) to ensure that there was an equal time duration to perform the tasks.

For each participant, the corresponding .oar files of the regions were reloaded and settings were reset. For the experiment group, named Mgt-Island group, two adjacent regions were loaded with Introduction and Management Island .oar files. For the controlled group, named Intro-Island group, a single region was loaded with the .oar file of Introduction Island.

The experiment was conducted over a two month period with a balanced mix of participants from a number of Scottish universities. Once again the 2012 SICSA conference was used as a forum to attract a good number of participants. The fact that the session takes an hour and users had to provide hardware facilities to participate in the session made it somewhat challenging to conduct the experiment; for the three days of the conference data were successfully recorded from 31 completed sessions. The remaining 39 sessions were completed by visiting the participant’s universities for their convenience and the rest were from the University of St Andrews. The experimental sessions were performed in an anonymous mode from the start to the end including feedback, therefore no personal or institutional information was recorded (thereby complying with the anonymous user study guidelines); moreover, since we are interested only in the comparison of performance between the two samples as a direct measure of the training experience provided by the two islands, user information is not relevant for the analysis.

7.2.2 Task Scenario

The complete task scenario, presented in Table 7.1, shows the intuitive and practical association of the abstract tasks with real world academic module requirements in an OpenSim MULE setup. The names used in the scenario are hypothetical and randomly generated using the publicly available tool, Behind the Name [228].

<table>
<thead>
<tr>
<th>Task Scenario to Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>You coordinate the course module AB30XX while sharing the lecturing with another faculty member. 12 students have enrolled and you are given 2 PGTutors for the module. Considering the Learning Objectives (LO) you have decided to incorporate 3D virtual worlds supported learning (3D Multi User Learning Environments – 3D MULE) activities to a significant portion of your teaching, learning and assessment. User accounts have already been created by the system administrator. You are given a region named AB30XX-Learning with the ownership of the Estate/Region. Some of the content and environment settings have been already done to facilitate your work. The region AB30XX-Learning contains a Sandbox, land areas for individual and group activities, and required teaching and learning constructs as the default layout to suit the module objectives/tasks.</td>
</tr>
<tr>
<td>These are the user details for the module.</td>
</tr>
</tbody>
</table>
When you are inside the region AB30XX-Learning, please complete the following tasks to prepare the environment for the module’s teaching and learning requirements.

1. Decide the suitable MUVE Roles for each user category (tick the suitable cages)

<table>
<thead>
<tr>
<th>Role</th>
<th>Estate Owner</th>
<th>Estate Manager</th>
<th>Parcel Owner</th>
<th>Group Owner</th>
<th>Group Officer</th>
<th>Group Member</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PGTutors</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Students</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Each student has to complete an individual project for the module. Assign the students to the marked land areas (you must create suitable land parcels and provide access restriction). How do you enforce the confidentiality of individual students work within their land parcel? Use suitable method for that. Furthermore, the land parcels should not be considered as sandboxes (Hint - actions that affect the parcel and learning should be discouraged).

3. Make sure that all incoming users arrive at the Information Forum but not elsewhere in the island.

4. The sandbox area is an unrestricted land for user activities; the other areas except for the given parcels must be restricted from alterations. Set appropriate settings in these areas.

5. As the default setting, voice is disabled in the region. Enable voice in suitable places (Hint- lecturing would be quite easy with voice, while library areas/document repositories/study rooms etc. might not be suitable with noise).

6. Lecture displays (presentations) and video content can be displayed through suitable media streaming URLs. Set the given displays with following URLs ensuring that students cannot change or visit other URLs.

7. Students are supposed to be in 4 groups (3 students per each) and assigned to the Tutors (2 groups per tutor). Create the required groups, suitable roles with abilities and invite the users, as shown below.

<table>
<thead>
<tr>
<th>Group Name</th>
<th>Tutor</th>
<th>Students</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alpha</td>
<td>John Carter</td>
<td>Jacques Labelle, Zac Bradford, Malone Macbay</td>
</tr>
<tr>
<td>Bravo</td>
<td>John Carter</td>
<td>Edan O’hanegan, Mario Sexton, Langdon Behrend</td>
</tr>
<tr>
<td>Charley</td>
<td>Andrew Hull</td>
<td>Chauncey Acquafredda, Daly Head, Chet Eldred</td>
</tr>
<tr>
<td>Delta</td>
<td>Andrew Hull</td>
<td>Hildred Bloxam, Gerhold O’hara, Donnchadh Revie</td>
</tr>
</tbody>
</table>

8. Student groups must be given the land access for their group projects. Set the lands suitable for the group activities with identified group roles.

9. A basic set of content objects for student projects (individual and group) has to be given with suitable permissions. Set the permissions for these content objects and place those on the parcels for student access. (Hint - group projects may need collaborative content editing among members. You may also want to prevent accidental deletes of this content set by ensuring the original objects given to the
students remain intact throughout the course duration.)

10. Some of the land areas and content objects (marked) have not been set to your ownership with suitable permission settings. Make those objects and land flexible, so that to facilitate your module coordination tasks in the future (Hint - You may need to use administrator functions).

To complete please visit the blended learning area of the island and participate in the online quiz (there are 10 MCQ type questions).

End of Task.

Table 7.1: The Task Scenario given to the experiment participants

It is important to mention that, in order to include all the task categories which have a meaningful management tasks this scenario was developed with the focus on module coordination. Although the tasks assumed the participant was acting as the module coordinator, these tasks are generic for any user that performs OpenSim MULE management irrespective of their institutional roles.

7.2.3 Sloodle Mediated MCQ Session

A set of 10 MCQs representing various OpenSim MULE management challenges was prepared as part of this experiment. These MCQs were designed to capture participant awareness of complex OpenSim management challenges and associated system models that they were supposed to have learnt from the training islands. Table 7.2 shows the set of questions that was used with the given choices as the answer set for each; the correct response is shown in italic.

<table>
<thead>
<tr>
<th>No</th>
<th>Question and Provided Answers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Which of the following methods allows you to communicate with a selected student avatar privately?</td>
</tr>
<tr>
<td></td>
<td>a) Text Chat</td>
</tr>
<tr>
<td></td>
<td>b) Instant Messaging</td>
</tr>
<tr>
<td></td>
<td>c) Voice Chat</td>
</tr>
<tr>
<td></td>
<td>d) Presenting the message In-world</td>
</tr>
<tr>
<td>2</td>
<td>Which of the following help you to keep your land (except for the parcels given to students for their work) tidy from unwanted content, while letting students practice content creation?</td>
</tr>
<tr>
<td></td>
<td>a) Abandon land</td>
</tr>
<tr>
<td></td>
<td>b) Block Terraforming</td>
</tr>
<tr>
<td></td>
<td>c) Object Return functions</td>
</tr>
<tr>
<td></td>
<td>d) Disable content creation</td>
</tr>
<tr>
<td>3</td>
<td>Which of the following you cannot control through system functions during the educational session?</td>
</tr>
<tr>
<td></td>
<td>a) Avatar appearance change</td>
</tr>
<tr>
<td></td>
<td>b) Flying over the land</td>
</tr>
<tr>
<td></td>
<td>c) Voice communication</td>
</tr>
<tr>
<td></td>
<td>d) Student accessing various land areas</td>
</tr>
<tr>
<td>4</td>
<td>In which scenario can the Cyclic Permission Loss affect your teaching</td>
</tr>
<tr>
<td></td>
<td>a) You give an educational content to a student for reference</td>
</tr>
<tr>
<td></td>
<td>b) You give a student an object, asked to improve and give it back for assessment</td>
</tr>
<tr>
<td></td>
<td>c) Students take a copy of your object and then delete that copy</td>
</tr>
<tr>
<td></td>
<td>d) Students try to modify a locked object</td>
</tr>
<tr>
<td>5</td>
<td>Which of the following permission models is often not challenging for educational content management</td>
</tr>
</tbody>
</table>

162
activities?

a) Composite Permission Model
b) Cyclic Permission Model
c) Fair Ownership Model
d) Base and Effective Permission Model

6 Which of the following Group Role Abilities that can be delegated to the Group Officers safely?

a) Change music and media settings
b) Assign members to any role
c) Remove members from roles
d) Assign and remove abilities in Roles

7 Which of the following actions that cannot help to manage performance issues with the simulation regions at runtime?

a) Disable Scripts
b) Disable Collisions
c) Return objects
d) Deed land

8 Which of the following practices may promote student self-regulatory behaviour while providing a realistic and intuitive learning with natural user collaboration?

a) By having random avatar names for students
b) By having student index numbers as avatar names
c) By having student real names as avatar names
d) None of the above

9 Which of the following actions that Estate Managers cannot perform?

a) Manage Access and Banned users
b) Manage public access
c) Eject a selected user from the estate
d) Manage Estate Manager list for an estate

10 Which of the following MUVE user role type of the university’s MULE that is not suitable for an undergraduate student?

a) Estate Manager
b) Parcel Owner
c) Resident
d) Visitor

Table 7.2: Sloodle MCQ set with the answers

The Sloodle arrangement was designed to let the participants experience a complete MULE training activity; they could have been left to participate in the Moodle assessment directed onto a prim as a webpage, but that would not give them exposure to other possible arrangements of blended learning in OpenSim MULEs. However, due to the available time and skill constraints these users just played the participant role, where the setup of Moodle and Sloodle was already done for them.
7.2.4 Marking Scheme for Evaluation

To evaluate the participant performance for the given subtasks a suitable marking scheme was prepared considering the major activities and functions of each subtask. Depending on the complexity of each subtask and the activities it consists of, varying levels of marks have been allocated. Table 7.3 summarises the marking scheme.

<table>
<thead>
<tr>
<th>Subtask</th>
<th>Sub Task Activities</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Role Assignment</td>
<td>Lecturer role assignment</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>PGTutor role assignment</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Student role assignment</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>6</strong></td>
</tr>
<tr>
<td>2 Parcel Management</td>
<td>Creating parcels</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Sell parcels to students</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Set parcel visibility</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Access and ban lists</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Terraform block</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>5</strong></td>
</tr>
<tr>
<td>3 Teleport Management</td>
<td>Set Teleport routing</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Block routing elsewhere</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>2</strong></td>
</tr>
<tr>
<td>4 Land Management</td>
<td>Sandbox settings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Settings for the rest</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>5</strong></td>
</tr>
<tr>
<td>5 Communication Settings for Teaching</td>
<td>Enable/Disable voice</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Identify Locations</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>2</strong></td>
</tr>
<tr>
<td>6 Content and Media Management</td>
<td>URL set</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Settings for URL lock</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>2</strong></td>
</tr>
<tr>
<td>7 Group Management I</td>
<td>Group creation</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Roles with abilities</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>User invitation to groups</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>4</strong></td>
</tr>
<tr>
<td>8 Group Management II</td>
<td>Land deeds to groups</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Assign roles with (land abilities)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>3</strong></td>
</tr>
<tr>
<td>9 Advanced Content Management</td>
<td>Content look</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Individual settings (VMCT)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Group settings (VMCT)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Deed objects to groups</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>6</strong></td>
</tr>
<tr>
<td>10 Administrative Powers and God Tools</td>
<td>Request Admin status</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Use of God tools</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Content unlock and permission set</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>5</strong></td>
</tr>
</tbody>
</table>

Table 7.3: Allocated score points for each subtask and its component activities

The total score for the set of subtasks was set to 40. A further 10 marks were allocated for the MCQ section, 1 mark for each correct answer. This made the entire experiment task out of 50 for each participant.
7.3 Experiment Island Design and Development

For the experiment a special virtual region was also designed and developed. The objective of this island was to allow participants to conduct the given OpenSim MULE management tasks.

7.3.1 Experiment Island

Fig. 7.27 shows an aerial view of the completed island ready to be used for the experiment. To accommodate individual student parcels and student group parcels the island was implemented with separate land markings as shown in 7.2. Within an experiment session it can be unnecessarily time consuming to mark land areas of equal size adjacent to each other, without support. To minimise this, boundaries of the land parcels were marked, but without dividing the land into parcels.

**Figure 7.1:** Experiment Island

**Figure 7.2:** Land border walls for parcel creation for individual and group projects
To simulate an educational environment that needs to be managed, discussion/lecture halls, content display halls and a sandbox area were used. The island design aimed at providing the infrastructure to be managed where participants were expected to perform the management tasks.

Fig.7.3 shows the content packages (left image) designed for management tasks on group and individual student projects. The left image shows empty content display boards to set appropriate teaching content. These correspond to the content management tasks for the evaluation.

**Figure 7.3:** Content objects and empty content holders for evaluation tasks

A special Sloodle setup linking a backend Moodle course environment was used (Fig. 7.4). The Sloodle tool Quiz Chair was used to generate a MCQ question answering session for participants to engage in as the last activity of the evaluation.

**Figure 7.4:** Sloodle setup with user registration and MCQ quizzes on Quiz Chair
7.4 Data Analysis

This section presents the statistical analysis performed on the experiment data. With the preliminary observations during the experiments and pre-test of the recorded data on fitness for the analysis, two cases, one from each sample, that do not provide any meaningful information for the study were removed from further analysis making the sample sizes 34 for each group.

7.4.1 Data analysis on scenario tasks

This section presents a detailed comparison of the results between the two groups of participants on each of the subtasks. The same naming convention is followed for the two groups throughout: Intro-Island for the group that used only Introduction Island (the control group) and Mgt-Island for the group that used both Introduction and Management Islands (the experiment group).

For comparison purposes the scores for each subtask have been transformed into a percentage score with respect to the maximum possible score for a given activity. This enables a clear comparison not only in a particular activity of a subtask but also among different subtasks and their activities, individually or as a grouped measure.

Subtask 1 – Role Management

The objective of this subtask was to explore how the two islands had improved user knowledge about managing OpenSim in the context of existing institutional learning infrastructure. The percentage scores for this subtask and each activity component are shown in the graph in Fig. 7.5. The Intro-Island sample showed relatively low scores in particular on deciding the corresponding roles for Lecturers and PGTutors when compared to the Mgt-Island participants. For selecting suitable roles for students the Intro-Island sample showed better performance, yet the Mgt-Island sample got the higher score. This is a subtask that directly examined participant knowledge and did not involve any in-world activity. It was evident that the participants from Mgt-Island group had gained a better understanding on user management in OpenSim, as shown in the subtask total score.
Subtask 2 – Parcel Management

This is the first task the participants tried in-world. It consists of 5 straightforward activities. Percentage scores are shown in Fig. 7.6. The first activity, creating parcels, seems to be the most successful for both groups. The Mgt-Island group reported full scores for that, i.e., everyone of that sample completed it correctly. Intro-Island group also showed a high success rate; in fact, creating parcels is a basic task available in the About Land widget and that explains why it was successful for both groups. However, the remaining four tasks, particularly parcel selling and setting visibility showed large differences between the two samples indicating that the Mgt-Island group has outperformed the Intro-Island group. These tasks require advanced skills for managing the land and associated access controls in-world. The results indicate that the experiment group performed better showing nearly 30% score difference.

Figure 7.5: Average performance scores (%) for Subtask 1
Subtask 3 – Teleport Management

This subtask included two activities: one is a very basic task of setting teleport links and the other is a management setting to control teleport behaviour. Fig. 7.7 shows the comparison of percentage scores between the groups for these two activities and the total scores for the subtask. Intro-Island group scored well in the first task nearing their performance with the Mgt-Island. However, when it comes to the advanced second task, a clear score drop was observed while the Mgt-Island group scored relatively well.

Figure 7.6: Average performance scores (%) for Subtask 2
Subtask 4 – Land Management

This subtask examines the user’s ability to perform complex tasks that require configuring a given piece of land. The participants were asked to set two different types of land areas: One with settings suitable for sandbox type functionality, and the other for formal educational requirements. Performing sandbox type settings is comparatively easy since such land area shows non-managed land parcel characteristics, more or less. The settings for the other land area required suitable controls to facilitate the expected learning with high environment management. In fact, the decisions to set these settings required a clear understanding of how different settings affect each other and how to avoid conflict situations. Results shown in Fig. 7.8 indicate that both groups found it somewhat challenging to set suitable land settings particularly for the learning areas. However, a clear success was shown by Mgt-Island group relative to the Intro-Island participants and the overall total score was higher.

Figure 7.7: Average performance scores (%) for Subtask 3
Subtask 5 – Communication Settings for Teaching

Communication settings are easier to perform in comparison to other tasks. In fact, the training contents on UI widgets are common for both groups therefore the Intro-Island group also performed the task relatively well. Fig. 7.9 shows the scores for the two groups. As the results indicate Intro-Island group was able to score reasonably well for the first activity although Mgt-Island group showed a strong lead. The second activity, identifying suitable locations for these settings, required knowledge of different learning activities and MULE learning expectation. Nearly consistent scores were reported by the Mgt-Island group for both tasks while the Intro-Island group showed a lower skill level for the second part. This further suggests that, without adequate training for OpenSim management requirements, a mere training on basic UI widgets - the usual Help-Island like user tools – may not suffice for successful management of OpenSim MULEs.
Figure 7.9: Average performance scores (%) for Subtask 5

**Subtask 6 – Content and Media Management**

This subtask includes a basic activity of setting given URLs as the media sources for educational materials and also advanced control settings. Fig. 7.10 shows the average percentage scores of each group for the two activities and the total score for the subtask.

Figure 7.10: Average performance scores (%) for Subtask 6
We can see setting the given URLs, a basic function available, was performed well by both groups. However, management of media use through URL settings involves several complex tasks, resulting in a much lower score for Intro-Island group compared to the Mgt-Island participants. Furthermore, the Intro-Island group subtasks total score would have been even less if we only considered the management related functions of the subtask. The URL lock activity is an important content management task that ensures the reliability of learning content presented. If it is not properly managed, students can easily navigate into arbitrary URIs in the Internet, not only distracting other learners but also bringing harmful or inappropriate content into the learning session. This shows the risk of failing to provide a formal educational activity if the Intro-Island group participants were allowed to manage due to a lack of knowledge. The consequences of such failures in credit bearing course activities can unarguably impact on a university’s credibility to offer a reliable learning environment.

Subtask 7 – Group Management I

In this subtask participants were asked to perform group management for the module they manage. Creating a group is a fairly straightforward process compared to managing one. The first activity, obviously, asked the participants to create the required groups to represent the module objectives and existing learning setup in e-Learning and traditional environments. Fig. 7.11 presents the scores for each activity and totals for the subtask between the groups. Both groups did well in creating groups. The other two activities are forms of group management; setting roles and deciding suitable abilities require advanced skills. Inviting users to join a group requires further settings to be performed as the group owner, which comes under the group ownership and management. A clear difference on scores can be seen between the two groups for these two activities and for the total subtask score, suggesting that to perform group management tasks accurately users have to be trained in the specific settings associated with groups in OpenSim.
Subtask 8 – Group Management II

This subtask included a further set of activities required to manage group-based learning for the module. The first activity was to set the land areas marked for the group tasks to the respective groups. It is another basic task, which required creating parcels and assigning those to the groups through clicking the UI widget Deed Land. Since parcel creation has already been evaluated as another task, only the deed function was focused on this activity score. Both groups did equally well (Fig. 7.12). The second activity was assigning created roles with suitable land abilities. Once again this is a complex task that requires knowledge on how different group land abilities not only interact with each other but also with the parcel and region ownership models. The performance in this task was low for the Intro-Island group compared to the Mgt-Island group.

Figure 7.11: Average performance scores (%) for Subtask 7
Overall, this subtask was relatively difficult and time consuming, yet the Mgt-Island group did better compared to the controlled sample. As a summary of group management tasks, we can see that the training typically given through Help Island type region is insufficient for management purposes. This reinforces the view that specialised training is required for using OpenSim for formal educational needs.

**Subtask 9 – Advanced Content Management**

The subtask starts with a very basic activity of content lock, which was performed quite well by almost all the participants from both groups (Fig. 7.13). The remaining three activities required advanced skills in settings. Out of the three, the easiest was the VMCT settings per individual use, then group use, and finally deed objects to groups with correct settings. The Intro-Island group scored much less when compared with the Mgt-Island group. In particular, the more complex tasks that required a thorough understanding of interrelated content settings were relatively well performed by the participants from the Mgt-Island sample.
Subtask 10 – Use of Administrative Powers and God Tools

The subtask, excluding the first activity, represents some of the ultimate administration activities that an authorised user can perform through the client application. Requesting admin status is a pretty straightforward task that can be done with a single menu click. However, it is available in the Develop menu and can be difficult to locate without training; in fact that can be the reason for the comparatively low score for the Intro-Island group when requesting Admin status. The most important finding about the next two activities is the very low performance by the Intro-Island group indicating their inability to perform complex administration tasks in-world. Academics usually have to set proactive management settings in practice. However, if there is a need for intervention to rectify a problem, academics have to enforce the required state of the learning environment through management. For example, in OpenSim MULEs, this becomes a prominent need to fix certain types of content, land and user related problems. Unfortunately, it is evident that the standard training given in Help Island like introduction environments and user guides do not provide this information putting the users at risk of failing to meet the needed level of OpenSim management.
Table 7.4 summarises the results for each subtask in a comparative manner highlighting the % score difference (Δ) as well. To examine the suitability for statistical analysis in a combined manner, we compared the means of each subtask between the groups. The results, *F* statistics and significance values were obtained from SPSS (version 19).

<table>
<thead>
<tr>
<th>Subtask</th>
<th>Intro-Island Score (%)</th>
<th>Mgt-Island Score (%)</th>
<th>Δ (%)</th>
<th><em>F</em></th>
<th>Sig.*,**</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>48.04</td>
<td>64.22</td>
<td>16.18</td>
<td>17.677</td>
<td>.000</td>
</tr>
<tr>
<td>2</td>
<td>50.29</td>
<td>79.41</td>
<td>29.12</td>
<td>36.033</td>
<td>.000</td>
</tr>
<tr>
<td>3</td>
<td>57.35</td>
<td>73.53</td>
<td>16.17</td>
<td>5.166</td>
<td>.026</td>
</tr>
<tr>
<td>4</td>
<td>34.41</td>
<td>54.41</td>
<td>20.00</td>
<td>21.094</td>
<td>.000</td>
</tr>
<tr>
<td>5</td>
<td>66.18</td>
<td>83.82</td>
<td>17.64</td>
<td>4.526</td>
<td>.037</td>
</tr>
<tr>
<td>6</td>
<td>47.06</td>
<td>73.53</td>
<td>26.47</td>
<td>11.091</td>
<td>.001</td>
</tr>
<tr>
<td>7</td>
<td>46.57</td>
<td>69.61</td>
<td>23.04</td>
<td>9.812</td>
<td>.003</td>
</tr>
<tr>
<td>8</td>
<td>32.35</td>
<td>55.88</td>
<td>23.53</td>
<td>21.647</td>
<td>.000</td>
</tr>
<tr>
<td>9</td>
<td>51.29</td>
<td>76.65</td>
<td>25.36</td>
<td>43.735</td>
<td>.000</td>
</tr>
<tr>
<td>10</td>
<td>19.12</td>
<td>74.51</td>
<td>55.39</td>
<td>153.600</td>
<td>.000</td>
</tr>
</tbody>
</table>

*p<0.001, **p<0.05

Table 7.4: The comparison of subtask score means between the groups

It is important to test each of the subtask scores for their suitability for a combined measure to use in the mean comparison. As the result set shows in Table 7.4 we can see all of the subtask scores show a significant mean difference (*p<0.05*). Therefore, we can proceed with the analysis using the cumulative values for the groups with the confidence of not having the Type I error in the final analysis (Type II error was eliminated by using the power level of *1-β* at 0.8).
Fig. 7.15 shows the cumulative percentage score (average) per each group for the entire task scenario. A clearly visible difference between the two samples was noted; however, it was decided to evaluate the difference statistically for accurate conclusions.

![Average % Marks for All Subtasks](image)

**Figure 7.15:** Average percentage score for the task scenario

### 7.4.2 Analysis of MCQ responses

All the participants from both groups successfully tried this activity; each correct answer was given a point. The number of correct responses for the questions is compared in Fig. 7.16.

Questions 1 and 10 show a high turnover of correct answers for both groups and questions 6 and 7 show the lowest total correct answers per question, suggesting that these are the easiest and most complex questions, respectively. To compare the difference in performance between the two groups the percentage difference per question was calculated as a ratio to the maximum possible; these are shown in Fig. 7.17. The minimum difference of scores reported in question 8 indicates that both groups are more or less equally capable of giving the correct response. In fact, question 8 is about student naming practice that can be useful for the management of OpenSim; it does not involve any complex management setting that users have be aware of. In contrast, question 4 resulted in the highest score difference. It asks about one of the complex permission models in OpenSim putting it into the educational context of MULE management. A close examination indicates that the questions that directly examined the participant knowledge on complex system and management models of OpenSim showed a higher score difference. This suggests that Management Island trained the participants better for the needs of OpenSim based MULE management compared to Introduction Island.
To avoid type I error propagated into the final analysis with the combined measures the two group scores of the MCQs were tested comparing the means. The obtained results indicate significant mean difference between the two samples ($F = 89.00, p < 0.001$).
7.4.3 Comparison of Group Scores

From the descriptive statistics it was found Intro-Island group reported mean score ($\mu_1$) of 19.912 and Mgt-Island group reported mean score ($\mu_2$) of 33.206. The respective standard deviations are $\sigma_1 = 2.790$ (Intro Island) and $\sigma_2 = 3.389$ (Mgt-Island). The box plots for the total scores of the two samples are shown in Fig. 7.18. Although there is a clear difference between the mean scores and the distribution parameters, it was decided to confirm the nature of results though a statistical analysis. The following examinable hypotheses were used.

**Null Hypothesis** – There is no difference in mean performance scores between the group trained with Management Island and the group trained with Introduction Island

**Alternative Hypothesis** – There is a difference in mean performance scores between the two groups

SPSS (Version 19) is used to test these hypotheses with ANOVA and the output of the test is shown in Fig. 8.15. The One Sample Kolmogorov-Smirnov test indicated that the final score data samples are normally distributed ($\alpha = .06$) while the Levene Test for homogeneity indicated that the variances of the two samples are not significantly different ($p > .05$) fulfilling the assumptions needed to use an ANOVA test on the data set. As Fig. 7.19 shows, ANOVA resulted in significant $\alpha (p < .05)$ indicating that the Null Hypothesis should be rejected.

![Final Scores Distribution of the Tasks - Between Groups](image)

*Figure 7.18*: Total score distribution box plots for both groups
This indicates that there is statistically significant evidence to say that the mean performance scores of
the two groups are different; the score values indicate, therefore, a better performance by the
experiment group with Management Island against the controlled group with Introduction Island.
Moreover, the detailed analysis of each subtask performance and MCQ results indicate a better
performance by Mgt-Island group, particularly with complex MULE management tasks. Therefore it
indicates that the provision of management training seems to be important for making the users
competent for conducting managed learning in OpenSim. Furthermore, the prepared tool support
sufficiently enhances the user capabilities in performing these management tasks in OpenSim compared
to the existing standard practices for training users.

Accuracy of Data

The data used for the above ANOVA analysis were collected from a pre-determined minimum sample
size required for the test. The total sample size 68 provided sufficient scores for the analysis; however,
the results were further analysed for sufficient anticipated effect with the obtained descriptive statistics
to validate the accuracy of data for ANOVA. For that Cohen’s $d$ [205], the standardised mean
difference, i.e., sample mean difference divided by the standard deviation ($S$), given by (1), was
calculated.

Where,

$$d = \frac{\bar{x}_1 - \bar{x}_2}{S}$$  \hspace{1cm} (1)$$

$$S = \sqrt{\frac{(n_1 - 1)\sigma_1^2 + (n_2 - 1)\sigma_2^2}{n_1 + n_2}}$$

Figure 7.19: SPSS (v19) output of ANOVA for testing hypothesis
; \( n_1, n_2 \) are sample sizes and \( \sigma_1, \sigma_2 \) are standard deviations for the groups, respectively.

With the obtained data the \( d \) was found to be 3.054, which suggests that the data from the experiment samples are strong enough to provide accurate analysis for requirements of \((1-\beta) = 9.999E-1 \) and \( \alpha=0.001 \) (where min. total sample size required as 32). Therefore, we can confirm that the results are not due to statistical errors and sufficiently indicate a clear mean difference.

### 7.4.4 Usability Analysis

One of the main objectives of this experiment was to compare the usability of the training provided on the two islands. A SUS questionnaire was used at the end of each experiment session. The standard questions and the received average response values for both groups are shown in Table 7.5.

<table>
<thead>
<tr>
<th>Question</th>
<th>Average Response Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>I think I would like to use this system frequently</td>
<td>3.94 Inte-Island</td>
</tr>
<tr>
<td>I found the system unnecessarily complex</td>
<td>1.85 Inte-Island</td>
</tr>
<tr>
<td>I thought the system was easy to use</td>
<td>3.86 Inte-Island</td>
</tr>
<tr>
<td>I think that I would need the support of a technical person to be able to use this system</td>
<td>2.09 Inte-Island</td>
</tr>
<tr>
<td>I found the various functions in this system were well integrated</td>
<td>3.82 Inte-Island</td>
</tr>
<tr>
<td>I thought there was too much inconsistency in this system</td>
<td>2.15 Inte-Island</td>
</tr>
<tr>
<td>I would imagine that most people would learn to use this system very quickly</td>
<td>3.79 Inte-Island</td>
</tr>
<tr>
<td>I found the system very cumbersome to use</td>
<td>1.82 Inte-Island</td>
</tr>
<tr>
<td>I felt very confident using the system</td>
<td>3.88 Inte-Island</td>
</tr>
<tr>
<td>I needed to learn a lot of things before I could get going with this system</td>
<td>1.94 Inte-Island</td>
</tr>
</tbody>
</table>

**Table 7.5:** Average response values for the SUS questionnaire

The distribution of the SUS scores is shown in Fig. 7.20. Other than the frequencies, we can see that the ranges of the SUS score distributions for the two groups are the same suggesting similar results. Moreover the mean SUS scores are: for Intro-Island group = 73.06 and Mgt-Island group = 76.18, which indicate a similar usability experience from the two regions. The result was examined statistically.

For the statistical evaluation following two examinable hypotheses are used:

*Null Hypothesis* – There is no significant difference of usability between the two islands

*Alternative Hypothesis* – The usability scores of the two islands are significantly different
SPSS (Version 19) was used to test the hypothesis with ANOVA and the output of the test is shown in Fig. 7.21. The One Sample Kolmogorov-Smirnov test indicated that the SUS data are normally distributed ($\alpha=.347$) while the Levene Test for homogeneity indicated that the variances of the two samples are not significantly different ($p>.05$) fulfilling the assumptions that should hold for an ANOVA test. As Fig. 7.21 shows, ANOVA resulted in non-significant $\alpha$ ($p>.05$) indicating to retain Null Hypotheses.

**Figure 7.20:** SUS value distributions of the two groups

<table>
<thead>
<tr>
<th>Test of Homogeneity of Variances</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUS_Score</td>
</tr>
<tr>
<td>Levene Statistic</td>
</tr>
<tr>
<td>------------------</td>
</tr>
<tr>
<td>1.794</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ANOVA</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUS_Score</td>
</tr>
<tr>
<td>Sum of Squares</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>Between Groups</td>
</tr>
<tr>
<td>Within Groups</td>
</tr>
<tr>
<td>Total</td>
</tr>
</tbody>
</table>

**Figure 7.21:** SPSS (v19) output for ANOVA to test hypothesis
The statistical analysis says that there is no significant evidence to reject the Null Hypothesis. This is in fact an interesting finding, which suggests that both islands are more or less equally usable for the purpose of training. We can see that both environments followed the same design architecture, the same ways of presenting content and were situated in the same server-client environment of OpenSim; only the training content was different. Therefore, from a usability perspective the islands are more likely to be equally usable, which was found to be true statistically as per the analysis. Both islands reported an average score of 74.89 suggesting a very good usability.

8.4.5 Educational Value

Educational value was measured through a questionnaire that has been used with other projects [92], with required modifications to put it in the Training Environment context. The questions ask the participant opinion about the educational benefits they have felt by using the system through a 5-level Likert scale and convert the total score (out of 50) into a percentage score. The questions and the respective average scores for both groups are shown in Table 7.6.

<table>
<thead>
<tr>
<th>Question</th>
<th>Average Response Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>I felt I have learnt something by using this system</td>
<td>4.62 4.41</td>
</tr>
<tr>
<td>The content on the Training Islands provides believable information</td>
<td>4.35 4.32</td>
</tr>
<tr>
<td>I found it easy to find out information on various complex MUVE functions and settings</td>
<td>3.35 4.59</td>
</tr>
<tr>
<td>The quality of the material presented was consistent</td>
<td>4.03 4.20</td>
</tr>
<tr>
<td>Interactive settings and practicing the training tasks in-world allowed me to improve my skills on using virtual worlds</td>
<td>4.06 4.55</td>
</tr>
<tr>
<td>I feel that using this system helps to develop my understanding of how to manage 3D Multi User Learning Environments</td>
<td>2.56 3.91</td>
</tr>
<tr>
<td>I found the system educationally stimulating</td>
<td>3.74 4.29</td>
</tr>
<tr>
<td>I was able to practice different management settings easily after using the system</td>
<td>2.29 4.02</td>
</tr>
<tr>
<td>The knowledge provided by the system allowed me to practice various tasks on managing the virtual environment</td>
<td>1.82 4.47</td>
</tr>
<tr>
<td>The educational content was intuitive and easy to understand</td>
<td>3.61 4.44</td>
</tr>
</tbody>
</table>

| Table 7.6: Average response values for educational value questionnaire |

Both groups indicated that they learnt something new in general. However, the scores for the specific questions that were associated with the outcome of the training and the tasks showed a clear difference between the groups. In particular, the 6th, 8th and 9th questions asked about the educational value for completing the given tasks; the Intro-Group on average disagreed with these statements reflecting the lack of support received. In contrast, the Mgt-Island group strongly agreed with those statements indicating that the training they had from Management Island was sufficient to perform OpenSim management tasks. Overall, Mgt-Island group participants showed an average educational value of 86.47 and Intro-Island participants showed average educational value as 68.89. This high difference is also visible through the score distribution shown in Fig. 7.22. We can see the range of Mgt-Island
educational value distribution is shifted two intervals towards the full-score compared to the control sample.

![Educational Value Score Distribution](image)

**Figure 7.22**: The distribution of educational value scores of the two groups

For a statistical comparison of the group scores two examinable hypotheses are used:

*Null Hypothesis* – There is no significant difference between the means of the educational value from the two islands

*Alternative Hypothesis* – The educational value from the two islands are significantly different

SPSS (Version 19) is used to test the hypothesis with ANOVA and the output of the test is shown in Fig. 7.23. The One Sample Kolmogorov-Smirnov test indicated that the educational value data are normally distributed ($\alpha=.091$) while Levene Test for homogeneity indicated that the variances of the two samples are not significantly different ($p>.05$) fulfilling the assumptions that should hold for ANOVA test. As Fig. 7.23 shows, ANOVA resulted in significant $\alpha$ ($p<0.05$) indicating to reject Null Hypothesis.
The analysis indicates that there is a statistically significant mean difference of educational value between the two groups. A higher educational value was reported for Management Island indicating its better suitability for managing OpenSim.

### 7.5 Open Feedback

Four open ended essay type questions were asked from the participants. The objective was to focus on the feedback that could not be covered within the standard SUS and the educational value questionnaires.

Overall, the response rate was satisfactory given the fact that the participants can leave the questions unanswered. Some of the participants provided ‘No’ as the answer for one or more questions. The received answers consisted of a range of opinions and concerns, which can be summarised as follows.

**Q1 - Is there anything you liked about this training environment in particular?**

A variety of answers was reported by the participants; the most common answer was about the look and feel of the designed environment. 11 participants more or less indicated that they thought the environment layout and constructs used, the textures and the pleasant outlook helped them engage willingly and tempted them to explore further. Another common view was that the simplicity of the training content and ideas presented gave users more room for their desired speed and ways of learning. Some of the selected statements are as follows.

From Mgt-Island sample:

[I liked the view of the islands most]

[The garden, buildings and images are beautiful... made me use the island willingly]

[Presentation is excellent. I learnt a lot.]
From Intro-Island sample:

[...A nice place to start my activities in virtual worlds]
[I found the provided information easy to understand, but it is not complete]

Q2 - Is there any difficulty you faced during the training session?

A different set of participant responses were received while the majority left the answer space blank or indicated No difficulties. Out of the meaningful statements, a common response was the difficulty in following a sequential path of learning. The training centre access was open and flexible after visiting information centre giving the avatar a freedom to wander around. However, some participants raised the concern of needing additional guidance. Another view was the concern about practicing the learnt OpenSim functions on that island, even on the sandbox. This was an interesting observation, which suggests high self-regulation among certain users. There were a few statements indicating the difficulty of flying and camera control during the early stages of the training session.

From Mgt-Island sample:

[The two islands were well designed. I didn’t find any challenge using those]
[Starting was bit challenging but the support content in these buildings made it easy. No difficulty.]

From Intro-Island sample:

[Training experience was not bad. I found some difficulties with your island at the beginning. I would prefer to have some hints to use this island]
[The first session was good. Second session with tasks was quite difficult.]
[I was lost in finding my way from training place to place. I suggest some form of arrows and numbers to guide users]

Q3 - What do you think about the tasks and questions you tried at the end of the training session?

Overall, a clear difference between the responses of the two samples was observed. The majority of the Intro-Island group clearly indicated that the tasks were difficult and the training they received was not sufficient. Furthermore, some of the participants indicated that they had felt a bit worried since certain questions and tasks were too complex even to understand. In contrast, from the Mgt-Island group, a lower number of responses were observed. The majority of the answers showed a confident view; a supporting observation for this study. There were few responses indicating that the training time and/or the task time was not sufficient to complete. This was specific to the experiment setup due to the limited time; in practice, students and staff can be given enough time to practice on these islands.

From Mgt-Island sample:

[I found the tasks are somewhat complex, but I finished most of those well]
[Some of the tasks are explained in the training places. It helped me]
[...I think the time I had to train and do these tasks was not enough ...]
[...the questions and tasks are more biased to the second island I used. They mostly include some advanced stuff, but I could manage most of the tasks. I’m not sure about my answers though...]

From Intro-Island sample:

[...For the tasks and questions the training I had was not helpful at all...]
[...The end questions and scenario were very complex. I couldn’t do much… this island must teach us about those advanced settings as well...]
[I felt something missing when I try to complete the tasks. Have I missed anything in the training places?]

Q4 - Is there any other comment about the training environment or the session you have experienced?

The responses were quite varied reflecting different views. The opinions were equally balanced between critiques with constructive options and positive views commending the value of the environment. On closer observation, it was revealed that the most of the critiques came from the Intro-Island sample directly claiming that the island is not complete and it should be improved. Again this was a recurring comment from this group of participants, which was due to the fact that they were not given the opportunity to experience Management Island. However, that view supports the research in an informal way indicating that there should be a properly designed training facility for the management of OpenSim.

From Mgt-Island sample:

[... It improved my skills for using this new technology. I could finish most of the tasks comfortably ...]
[This tool is highly recommended for training new users. I think it is useful for everyone]
[Everything looks fine. I enjoyed it!]

From Intro-Island sample:

[The virtual island I used was good. But I think it doesn’t help. It just helps a user to learn basic stuff. I recommend completing it with all details ...]
[It was fun. Things can be improved, but I don’t know]
[It didn’t help me for the tasks. I learnt about something, but not much help for the given tasks]

7.5.1 Unstructured Interview Findings – Informal and Unguided

At the end of each session a concluding discussion was opened up. A few of the participants said that they had no questions and finished the study without having this discussion.

In general, the participants found the entire session was interesting and enjoyable. They had learned new information and trained themselves in using OpenSim as a MUVE. The participants from the Mgt-Island group said they could perform scenario tasks well at the end, despite having known nothing about the management functions beforehand. This was positive and encouraging feedback that stressed the value of the training environment.
However, the Intro-Island group often noted that the training they had was not adequate, particularly in reference to the task scenario. They said that Introduction Island should be extended by including training content for advanced tasks. This was in fact, a further support for the research case that specific training on OpenSim functions is important for managing OpenSim MULEs. When they were informed that they were the control sample, and that there was another island for management training, they showed a keen interest in using Management Island. They were informed that once the study was completed it would be part of the university virtual world grid and available for download from the open virtual world repository [229] at the University of St Andrews.

Some participants expressed their willingness to have a multi-user session with others. Once the clarification was made that these specially designed individual sessions were mainly aimed at obtaining data for research evaluation they agreed.

### 7.6 Case study

Most of the constructive feedback was incorporated into later versions of the training islands. Directional guides and flows of learning were added; a brief guide document with initial access details including client setup and login was prepared.

One of the limiting factors with the experiment setup was that users had a relatively small time period to experience the environment, although it was quite sufficient for the expected tasks. In order to try the training environment in an MULE management activity it was decided to follow a set of mini case studies that used the two islands as the main form of training for OpenSim management.

For this evaluation a group of postgraduate students who have selected MUVE related research for their master’s dissertation projects was used. There were 6 taught MSc students that engaged in OpenSim projects; however, only 4 students participated in the study. The student project duration was 12 weeks; the first 2 weeks were used for project familiarisation and literature survey tasks. These 4 students had not used Second Life or OpenSim before, therefore they fitted our requirements. They were asked to first access the OpenSim grid with the environment and then allowed them to keep a locally installed environment on their own computers for practice. At the end of the first two weeks they were asked to provide their feedback.

#### 7.6.1 Structured Feedback

For the feedback on usability and educational value the same questionnaires as before were used. The responses from each postgraduate student are recorded in Table 7.7 for SUS and Table 7.8 for educational value. Likert scale values represent the standard responses as discussed previously, ranging from Strongly Agree (5) to Strongly Disagree (1).
<table>
<thead>
<tr>
<th>Question</th>
<th>Student Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>I think I would like to use this system frequently</td>
<td>5 5 5 4</td>
</tr>
<tr>
<td>I found the system unnecessarily complex</td>
<td>1 1 2 1</td>
</tr>
<tr>
<td>I thought the system was easy to use</td>
<td>4 5 5 5</td>
</tr>
<tr>
<td>I think that I would need the support of a technical person to be able to use this system</td>
<td>1 1 1 1</td>
</tr>
<tr>
<td>I found the various functions in this system were well integrated</td>
<td>4 5 5 4</td>
</tr>
<tr>
<td>I thought there was too much inconsistency in this system</td>
<td>1 2 1 1</td>
</tr>
<tr>
<td>I would imagine that most people would learn to use this system very quickly</td>
<td>4 5 5 5</td>
</tr>
<tr>
<td>I found the system very cumbersome to use</td>
<td>1 1 2 2</td>
</tr>
<tr>
<td>I felt very confident using the system</td>
<td>5 5 5 4</td>
</tr>
<tr>
<td>I needed to learn a lot of things before I could get going with this system</td>
<td>2 1 1 2</td>
</tr>
</tbody>
</table>

Table 7.7: SUS questionnaire response per each participant

The student responses for SUS indicate a clear bipolar orientation towards the two groups of questions. They have shown Strongly Agree and Agree scores to positive questions (odd numbered questions) and Strongly Disagree and Disagree scores to the negative questions (even numbered questions). This shows the high usability of the training environment. The question scores of educational value showed positive values indicating higher scores for each student.

<table>
<thead>
<tr>
<th>Question</th>
<th>Student Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>I felt I have learnt something by using this system</td>
<td>5 5 5 4</td>
</tr>
<tr>
<td>The content on the Training Islands provides believable information</td>
<td>4 5 5 5</td>
</tr>
<tr>
<td>I found it easy to find out information on various complex MUVE functions and settings</td>
<td>5 5 5 4</td>
</tr>
<tr>
<td>The quality of the material presented was consistent</td>
<td>5 4 5 4</td>
</tr>
<tr>
<td>Interactive settings and practicing the training tasks in-world allowed me to improve my skills on using virtual worlds</td>
<td>4 5 4 4</td>
</tr>
<tr>
<td>I feel that using this system helps to develop my understanding of how to manage 3D Multi User Learning Environments</td>
<td>5 5 4 4</td>
</tr>
<tr>
<td>I found the system educationally stimulating</td>
<td>3 5 4 4</td>
</tr>
<tr>
<td>I was able to practice different management settings easily after using the system</td>
<td>5 5 5 4</td>
</tr>
<tr>
<td>The knowledge provided by the system allowed me to practice various tasks on managing the virtual environment</td>
<td>4 5 5 5</td>
</tr>
<tr>
<td>The educational content was intuitive and easy to understand</td>
<td>5 5 4 4</td>
</tr>
</tbody>
</table>

Table 7.8: educational value responses per participant

SUS scores and educational value scores are shown in Table 7.9 for each student. The average SUS score is 92.5 and the average educational value score is 91.0. These responses suggest that the training
environment facilitates user training needs for managing OpenSim. The improvements made on the two islands after the experiment and user feedback made it more usable and effective for training as a combined environment. Moreover, these users had the opportunity to explore the two islands thoroughly, making their feedback accurate.

<table>
<thead>
<tr>
<th>Student</th>
<th>SUS Score</th>
<th>Educational Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student 1</td>
<td>95.0</td>
<td>90</td>
</tr>
<tr>
<td>Student 2</td>
<td>97.5</td>
<td>98</td>
</tr>
<tr>
<td>Student 3</td>
<td>90.0</td>
<td>92</td>
</tr>
<tr>
<td>Student 4</td>
<td>87.5</td>
<td>84</td>
</tr>
</tbody>
</table>

Table 7.9: SUS and educational value scores for each participant

7.6.2 Student Feedback and Other Observations

These users were supposed to complete their MSc project familiarisation phase within 2 weeks (16.67% of their project time) according to the course guidelines. They reported that they could complete their OpenSim training between 4-5 days (gained 64.28% - 71.43% of time saving, which they used for their project design work and further literature survey). Although it is a rough measure it may indicate the value of training they received.

After the initial stage of the training, two students asked to access training structures and content objects in the islands for further development. These two students were engaged in dissertation projects to developing MULEs for teaching computer networks and supporting student learning in Routing Island on the University OpenSim grid. The objective of their work was to make a middle-layer of educational islands for avatar interactions between the training environment and the corresponding educational islands, i.e., Routing Islands in this case.

The feedback through unstructured and unguided discussions indicated that they could clearly state their design and development plans focusing on the expected learning outcomes of the regions by making the assumption that the students who visit their regions have already trained for environment interaction and management. They also indicated what they had to include in these prerequisites, i.e., without the training content provided in our training islands for their island designs they would not have been able to complete their projects. These two projects were given MSc merit grades on completion; the students commended the support they received from the training environment; further evidence for the value of this training support not only to use and manage but also to develop and extend OpenSim MULEs.

7.7 Discussion

It was found that the training island for advanced settings of OpenSim had improved user capabilities for managing learning in Opensim MULEs, in contrast to the widely used Help Island type introductory islands. With the support of statistical analysis, we can therefore say that the provision of appropriate user support for OpenSim based MULE management is important for successful managed learning.
The findings and analysis results showed that the environments developed are highly usable irrespective of their training focus. Similarly, the findings indicated that for managing MULEs the provision of user support for advanced OpenSim system functions gives better educational value than using the available introduction environments and user guides. In overview, the findings suggest that the study contributions can help users in managing their OpenSim based teaching and learning.

7.7.1 Study Limitations

The experiment, its evaluation and the case study were affected by the following limitations. Careful measures with rigorous statistical methods and pre-tests were used to minimise the impact of these limitations.

The participants showed differences in the ways they preferred to interact and perform the given tasks, indicating different intrinsic personal skill traits. Certain approaches resulted in efficient completion of tasks while some other methods took longer times. However, this effect was distributed randomly in both groups and had equal effect in both samples; thus the comparison of performance was not affected. Moreover, any user session that was significantly affected by the time limit was not observed; often, it was the case that the participants were not aware of the functions and settings to use. Secondly, the data collection used did not take account of failed attempts at the given tasks. The halfway attempts were not considered for the task scores. However, this was common for both groups therefore did not affect the comparative analysis. The feedback captured through Lickert scale responses had quantified ordinal values, which may not provide an exact value for their response, but rather a range. However, this is a well-known feature in user studies; appropriate statistical tests were used and the data found suitable for the analysis.

Considering the statistical analysis results and the experimental methods followed, we can say that the evaluation results sufficiently meet the study’s objectives. Although further studies are recommended for generalised conclusions, the results of our evaluation indicated that the user training environment developed as a thesis contribution is supportive of the mainstream use of managing OpenSim MULEs.

7.8 Concluding Remarks

This chapter discussed how the evaluation of the training islands was designed and carried out. Detailed analyses of collected data were performed along with usability and educational value data. The findings showed that the entire training environment is usable while the special provisions made for training advanced settings of OpenSim show higher educational value and are essential for managing learning in these MULEs. The suggested improvements from participant feedback were made and the modified training environment was evaluated as a further case study; higher usability and educational value was observed. This chapter indicates the contribution of this research work and the tool support through evaluation. The next chapter summarises the research findings and thesis contributions while providing an outlook to possible further research extensions.
8. Conclusion

8.1 Introduction

This chapter summarises the research carried out and reviews the extent to which the objectives of have been achieved. It presents an overview of the selected MUVEs (Second Life and OpenSim) in the context of higher education, with particular respect to managing multi-user immersive 3D learning environments (MULE). The research contributions made and the objectives achieved are critically assessed while linking the present body of knowledge with possible future research opportunities. The thesis then finishes with concluding remarks.

8.2 Research Review and Contributions

MUVEs provide unique learning affordances and opportunities for an engaging educational experience of a type that cannot be achieved in existing 2D educational platforms. The immersive nature of the simulated environment in 3D space presents an intuitive look and feel representing user activities in the real world. These opportunities for enhanced user presence introduce novel challenges for the effective management of 3D learning environments. This thesis has focused on these challenges and investigated a possible solution through user support with respect to the two widely used, closely compatible MUVEs, Second Life and OpenSim. A unique set of user support models, tools and training solutions have been designed, developed and evaluated with respect to these environments. The research contributions in the form of user support models and tools are specifically aimed at providing the guidance needed for effective management at the system and implementation level.

8.2.1 Thesis Review

The thesis of this research has been defended for its significance and originality; it can be summarised as follows.

*The provision of user support for enhancing user self-regulation and contextual environment management can improve the management of learning in a Multi User Learning Environment.*

Evidence in support for this thesis has been shown as follows.

*Theoretical Underpinning* – Relevant educational theories and models have been discussed along with their applications in MUVE based education. Theoretical models and user studies of the management of learning environments have been used as the basis for the thesis and its research approach.

*Domain Feasibility* – System studies have been carried out on the selected MUVEs. The findings have been incorporated into the design and implementation of user support models and tools for managing
learning in these MUVEs. The close association with the OpenSim and SL system architecture ensures the compatibility and accuracy of these solutions.

Validity and Acceptance – This research has presented the evaluation of the thesis through implementations of the proposed user support tools and their evaluation for usability and educational value. The research contributions were quantitatively and qualitatively evaluated with appropriate case studies of working projects and artefacts in OpenSim and SL.

The work that has been done on this thesis has resulted in a number of peer reviewed research publications associated with research objectives. The contributions and study outcomes presented in these research publications have been accepted by the research community, which we take as further support in defending the thesis for its validity, acceptance and its original contribution, as a whole.

8.2.2 Objective Achievements and Overview of Contributions

The research sought to achieve the following objectives:

Objective O1 - Identify appropriate system models of the selected MUVEs for learning
Objective O2 - Identify productive use cases for teaching and learner support in these MULEs
Objective O3 - Identify the most significant management aspects for MULEs
Objective O4 - Develop user guidance models for MULE management
Objective O5 - Design and implement prototype user guidance and training tools for the selected MUVEs
Objective O6 - Evaluate the guidance and tools and make recommendations for learning management in the selected MUVEs

MUVEs in general and OpenSim in particular leave the management of learning environments on these platforms as the sole responsibility of the academics that use them. This thesis has discussed and evaluated how system and usage models of OpenSim/SL can produce significant challenges for the academic community when it engages in formal educational activities in MULEs based on OpenSim/SL.

The overview of the research achievements and the summary of research methodology along with the thesis and research objectives were presented in Chapter 1. Chapter 2 described relevant literature encompassing a broader knowledge base to date and reviewing related work.

Chapter 3 described the initial work which paved the pathway to the successful completion of the thesis. The system studies contributed to a synopsis on Second Life and OpenSim with respect to educational use while highlighting the possible challenges for learning management due to their system architecture. This chapter initiated the discussion on the need for managed learning in these MUVEs. Chapters 2 and 3 presented the work carried out with respect to the first objective, O1, of this research.
Chapter 4 discussed the use of MUVEs in blended educational scenarios. The chapter presented the analysis of possible integrations between OpenSim/SL and Moodle, which was supported by the related case study findings. The chapter also presented a user study and analysis carried out for identifying important MULE management aspects. The contributions of this chapter included the user support models relating MULE management aspects and a management process model linking MULE management with the existing learning management practices in academic institutions. The work discussed in this chapter represented achievements with respect to research objectives O2, O3 and O4.

Chapters 5, 6 and 7 presented the work carried out on the last two objectives, O5 and O6, in a collective manner.

Chapter 5 presented the research tasks conducted for achieving objectives O4 and O5. It explains the design and development of the OpenSim function network topology as a user guidance tool. The statistical analysis of the tool evaluation showed the accuracy of the information depicted by the tool while the case study demonstrated the user support given by the tool in a real educational MUVE project.

Chapter 6 discussed a virtual environment implemented in OpenSim for user training. Firstly, the chapter presented the findings of a user experiment to identify a suitable method of user support provision. Conclusions derived from the findings resulted in designing and developing a dedicated user training environment catering for introductory and advanced uses of OpenSim for managed learning.

Chapter 7 discussed the evaluation of the user support tools and guidance for managing an OpenSim based MULE. The findings of an experiment and a set of case studies with analysis demonstrated the unique support given by this training environment.

In overview, the work presented in each chapter contributed to the achievement of the stipulated research objectives thereby defending the thesis of this research. Therefore, we believe that the study has substantiated the research thesis through the various contributions and outcomes discussed throughout this dissertation. The next section discusses how these contributions could support further research.

8.3 Future Work

No research is complete if it cannot guide interested researchers in possible directions for further studies. This research focused on user support for MUVE based learning management. The research scope was confined to SL and OpenSim as required by the contextual goals of the study. The findings and contributions have successfully achieved the research objectives; however, the following further studies are suggested by the implications of this study.

This research examined the state of the art system models of Second Life and OpenSim. The core system and functional models have been the basis for the user support tools provided for learning management in these systems. However, these systems grow and evolve with certain modifications catering for wider user communities. One such important change, although at the experimental stage yet, is hypergrid support in OpenSim for interconnecting different OpenSim grids. If the experimental
The hypergrid concept is widely adopted for educational use, this clearly has the possibility of raising serious management challenges in the areas of access control, ownership and copyright management, and security. We believe that there should be further research on how the management practices of MULEs from different institutions can cohesively work together while overcoming the new challenges presented by the hypergrid infrastructure; to that end the user support models and tools developed in this research can be extended as required. Such research will be very important for the success of interconnected multi-institutional MULEs in future.

The work carried out in this research contributed a network topology and a reusable network dataset on OpenSim/SL functions written in a standard graph description language. The network topology can be used in support for OpenSim management at the function level, as discussed. An interesting future research would be to incorporate another layer of network information linking to the OpenSim function topology, which can provide support for generic learning management policies associated with formal education. As we have discussed an OpenSim (SL) function can be used to implement multiple policies, an extension to the given tool with that support could be useful; however, that development require further research on academic policies and their implementation strategies in general.

The model of blended learning with MUVE incorporation at system integration and user activity levels can introduce many further research avenues. This study focused on providing user support for managed MULE learning within blended learning setups; however, there can be a number of future research studies investigating different aspects of learning and teaching in blended MULE arrangements. Moreover, according to the research scope the study investigated only a selected set of applications in blended MULEs; further studies are recommended for investigating generic and bespoke blended learning applications with MUVEs in order to identify the requirements needed to create productive blended MULEs.

This study provided a set of user support models and tools for managing OpenSim and SL supported learning. These user support contributions have been sufficiently associated with suitable theories and pedagogical models for MUVE learning. However, the research scope does not include the evaluation of the effectiveness of student learning, which is a broader area of study. A valuable future research could examine student learning improvements in managed MULEs. Such research can take the advantage of the tools provided for managing OpenSim/SL based MULEs and examine student learning performances in their course modules. These evaluations can also incorporate the existing body of literature on evaluating student learning experiences in these managed MULEs.

This research is situated in university academic use. The research outcomes can provide a basis for similar studies in other application domains of OpenSim and SL. Although the user support models and tools developed are based on OpenSim and SL system architecture, a certain level of educational needs were associated with those tool designs. Therefore, it would be necessary to carry out application domain specific studies before directly applying the user training models and tools, for users in general.

This study did not consider contributions that required customising the OpenSim system architecture as well as proposing an additional layer of abstraction to mediate the MUVE and e-Learning domain for learning management. However, there could be situations which would favour that approach. It could be useful to consider the findings and user support tools contributed by this research for the design and development of mediating system layers for OpenSim/SL as a further study. The findings of this
research could help the development of a mediating layer as the basis for shaping its architecture and catering for educational needs and user support.

Another viable future research direction could be to develop a collection of tools that support specific needs of learning activity management in OpenSim and SL. In this study we have discussed challenges with using Sloodle as a consequence of its design model to map Moodle activities without considering MUVE affordances. A successful alternative may be to consider provisioning user support tools for teaching and assessment related basic tasks in-world facilitating to use the MUVE learning affordances. These tools can be deployed as in-world artefacts and avatar inventory items. The contributions and findings of this work can provide a good foundation for such research.

Another possible future study related to this work is the development of automated tool support for management policy development and implementation for OpenSim. It would be interesting to combine the findings and contributions of this work with a single process of policy development and implementation of those at system level through a set of integrated tools forming a management workbench. Although it would be a complex research project with a broader scope incorporating multiple application domains, such provision could be quite useful for managing complex and dynamic MULE learning scenarios rapidly with minimal user intervention. The user support models and tools developed by the research described in this thesis could be used as components of this management solution.

Finally, the contributions of this research can provide a good foundation for the future research on management and usage strategies of the 3D Immersive Web. The 3D I-Web is a major research interest for generic use as well as for education. The research methodology and the suggested user support for managing OpenSim/SL based MULEs demonstrate effective methods for integrating 3D virtual environments with existing 2D platforms. The outcomes and recommendations of this research could be extended for future research investigating possible approaches for harnessing the 3D I-Web into the existing 2D web infrastructure and the management of these.

8.4 Concluding Remarks

This thesis has presented a set of models and tools for supporting the management of a multi user immersive 3D learning environment (MULE). The work presented in this thesis is unique and new to the best of the author’s knowledge. The research findings show originality and a contribution to the field of MULEs. This thesis provides a review of the transposition of management and user support strategies from traditional and 2-dimensional e-Learning environments to novel MULEs. The contributions for the management of MULEs based on OpenSim and SL have been discussed and demonstrated with user guidance tools. These user support tools have been evaluated with user experiments and suitable analyses. The findings have shown that the work presented in this thesis and the provision of user support are important for managing learning and teaching support in MULEs as part of mainstream education.


Appendices

Appendix 1: List of Publications


Appendix 2: Terminology

**Avatar** – A virtual persona where the users are represented in-world. It can be of any form, such as, human like, animals and abstract shapes. Appearance of an avatar can be changed according to user’s desire.

**Estate** – The highest level of land entity that groups several regions together to support delegation of land management privileges. An estate has an estate owner and one or more assigned estate managers.

**Inventory** – A virtual folder that keeps a reference to the different types of content objects of an avatar. Users can take objects or content types from their inventory or save the developed or copied content in the inventory for future use.

**God** – The in-world administrator role to perform admin tasks through the client viewers.

**Island** – A virtual region surrounded by sea.

**Parcel** – The smallest unit of land that a virtual region can be divided into. At the maximum size a parcel is the same size as a region (65,536 m$^2$) and at the smallest size it is 16 m$^2$.

**Physics Engine** – is the software component that accounts for simulating the physics. It mainly simulates the gravitational forces and relevant dynamic properties. Physics can either be enabled or disabled at various levels depending on the desired simulation.

**Prim** – the smallest unit available for content object creation. Different virtual worlds provide different shapes; however, every system ensures a set of basic 3D shapes required for construction.

**Region** – a virtual land with fixed size of boundaries. Often, surrounded by sea (Island), if not adjacent to other land.

**Rezzing** – the action that creates content in-world or taking a content object from an avatar inventory to populate it in-world.

**Sandbox** – an openly accessible land area within a virtual region with no (or minimum) restrictions to practice content creation and to get familiar with the virtual environment. Students can be allowed to access freely and let them experience the virtual environment before engage in educational tasks.
Teleporting – an action that instantaneously moves an avatar between two regions or between two places within a region.

Appendix 3: How to Access the Training Environment*

1. Go to the web front-end of the University of St Andrews 3D MUVE grid on OpenSim by accessing http://www.openvirtualworlds.org/start/start.php

2. Follow the given instruction accordingly to complete the following:
   a. Register for an avatar
   b. Download a preconfigured viewer or set the grid URI of your viewer
   c. Login to the grid using your avatar information (Avatar Name and Password)

3. Once you are inside the grid you will be located at the HUB area

4. Search for region “Introduction” – Introduction Island or region “Management” Management Island using your viewer and teleport to the desired region

*Please note that URLs and virtual island locations can be changed.