Hybrid Remote/Virtual Laboratories with Virtual Learning Environment Integration

MJ.Callaghan, K.McCusker, J.Lopez Losada, JG.Harkin and S.Wilson

Intelligent Systems Research Center
University of Ulster, Derry
Northern Ireland, UK
mj.callaghan@ulster.ac.uk

Abstract— **In recent years there has been significant growth in the use of 3D virtual worlds for e-learning and distance education. These immersive environments offer the ability to create complex, highly interactive simulations using in-world modeling and scripting tools. Virtual learning environments' support teaching and learning in an educational context, offering the functionality to manage the presentation, administration and assessment of coursework. This paper will discuss the practicalities of using immersive virtual worlds for teaching and training purposes in the context of a hybrid virtual/remote laboratory. It will describe the linking and integration of a virtual world with an existing virtual learning environment and show how this approach can be extended to add external hardware which can be controlled and accessed remotely. In addition it will show how user interactions with the virtual and real elements of the laboratory can be recorded back into the virtual learning environment for subsequent assessment by the tutor. A number of demonstrative examples will be introduced and discussed.**

Keywords; Virtual worlds, engineering education, virtual learning environments, hybrid virtual/remote laboratories, hardware integration

I. INTRODUCTION

The constant emergence of disruptive technologies continues to offer new and exciting opportunities for educators [1]. Developments in recent years have seen web-based Virtual Learning Environments/Course Management Systems rapidly become an integral part of teaching and learning provision in further and higher education [2]. This evolution progresses unabated as educators strive to adopt and adapt web 2.0 technologies in the provision of more interactive teaching materials and learning environments which allow students the ability not only to view content but to interact and organize it to suit their personal needs [3]. Video games and virtual worlds are moving into the mainstream as traditional media industries struggle to keep with up digital natives and their desire for information, technology and connectivity [4-7].

This paper discusses the suitability of one popular virtual world for educational and teaching purposes. It explores how Second Life could be used to create experiential based learning experiences in 3D immersive worlds [8-10]. In addition it demonstrates how a range of external hardware devices can be integrated into virtual worlds, facilitating two way communications/control/manipulation of real and virtual artifacts between the real and virtual world. It then shows how data from these devices can be visualized in virtual worlds. A number of exemplars projects are introduced including the Engineering Education Island project in Second Life. Approaches to integrating Virtual Learning Environments and Virtual Worlds are discussed and the practicalities of introducing a top down narrative with a formal teaching structure and explicit user objectives/learning outcomes in a virtual world scenario detailed.

Section 2 of the paper discusses the Second Life virtual world and its functionality. Section 3 describes a range of University of Ulster research in virtual worlds. Section 4 provides an overview of how virtual learning environments can be integrated with virtual worlds. Section 5 demonstrates how complex learning experiences can be created in virtual worlds. Section 6 examines how external hardware devices can be integrated into virtual worlds and how the data from these devices can be visualized, manipulated and controlled. Section 7 concludes the paper.

II. VIRTUAL WORLDS AS TEACHING TOOLS

Second Life (SL) is an Internet-based 3D virtual world launched in 2003 and developed by Linden Labs. Second Life facilitates an advanced level of social networking where residents can explore and socialize by participating in individual and group activities [11]. Second Life contains basic tools that allow users to create and script interactive 3d content. The ownership and rights to this content remains with the creator who can subsequently chose to retain it, give it away freely or sell it. This flexibility and intellectual property ownership and the fact that basic membership of Second Life is free has led to the establishment of a very active educator community in Second Life and many universities, colleges and schools using the environment for teaching [12]. Figure 1 shows a class being taught in real time Second Life where each avatar represents a logged on student.

Figure 1 Teaching in Second Life

III. UNIVERSITY OF ULSTER IN SECOND LIFE

The University of Ulster, Magee campus Second Life project began in September 2006 where initial research focused on the potential of virtual worlds for remote collaborative working and undergraduate/postgraduate teaching [13]. Engineering Education Island was created in Second Life to investigate if virtual worlds could be used effectively for teaching and learning in this context. When residents first arrive on Engineering Education Island they land at the welcome center, a large futuristic building which includes general information on the project and teleport links to the demonstrations and exhibits. Each floor of the virtual laboratory contains a range of interactive engineering demonstrations and simulations (Figure 2). Other facilities on the Island include a virtual lecture theatre where students can attend classes and collaborative working facilities where students can work together remotely.
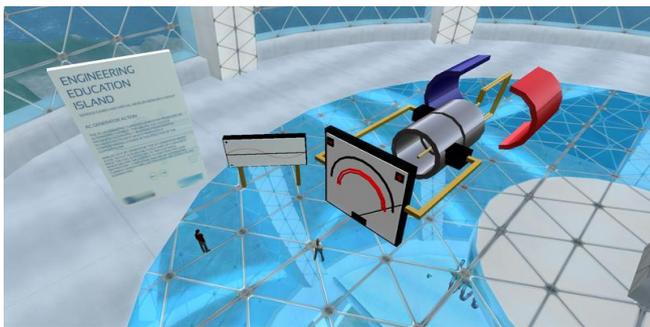

Figure 2 AC generator demo inside the virtual engineering laboratory

Figure 3 shows a typical interactive simulation on the Island which demonstrates the main functions/components of a direct current electric motor. It allows the user to start and stop the motor and to view the magnetic fields. Scripting was used to turn on and the off magnetic fields and to create the particle system showing current flow.
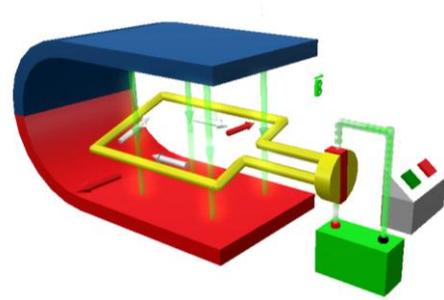

Figure. 3 Direct Current Electric Motor (Interactive)

IV.   INTEGRATING VLE'S AND VIRTUAL WORLDS

Moodle (Modular Object Oriented Dynamic Learning System) is a free and open source e-learning software platform which allows users to create and manage teaching materials [14]. The extensive course management tools available in Moodle do not currently exist in most virtual worlds whose main strength in this context is the provision of immersive spaces for social interaction and experiential based learning [15+16]. The SLOODLE open source e-learning software project (Simulation Linked Object Oriented Dynamic Learning Environment) offers the functionality to link Second Life and Moodle and to exchange and synchronize data flow between the two environments [17]. Interaction with external applications is facilitated by the Linden Scripting Language which can communicate with the PHP driven Moodle database, using XML-RPC, and outbound HTTP requests. Initially the interactive demonstrations and simulations created on Engineering Education Island did not have any explicit user objectives, formal structure or learning outcomes and had no means of recording user actions/interactions. To remedy this a number of projects were designed where the interactive web based resource aspect was augmented with a virtual world. A Moodle website with Sloodle functionality was setup which was subsequently modified and extended for this project. The approach taken uses the following structure, a registered student logs on to Moodle to access and review the learning/course material. The practical aspect of the taught material requires the student to enter the virtual world and carry out a range of task based activities, either individually or as part of a group related to a single learning objective (Fig 4).

PRACTICAL 1	Become familiar with the theory and operation of DC generators	
STUDENT NAME	Kerri Macchi	
TASKS	LEVEL OF COMPLETION	
Enter virtual world and go to interactive simulation	Completed	Not Completed
View video in world about learning outcomes	Video viewed	Video not viewed
View slide show showing tasks	Slides viewed	Slides not viewed
Press button 1 to turn on power	Interaction 1 Completed	Interaction 1 not completed
Press button 2 to observe magnetic fields	Interaction 1 Completed	Interaction 1 not completed
Overall percentage of tasks completed	XX% Completed	
Total time student spend on assignment	XX Hours XX Minutes	

Figure 4 In-world learner centered activities

Completion of practical outcomes in this context is based on the student's presence and interaction with in-world demonstrations in the virtual world. These actions are automatically recorded in Moodle for later retrieval and review by the student and the lecturer. The functionality required to track and record user actions/interactions is this way was not available in Sloodle at the time. Two simple extensions were required, a proximity sensor was needed which would record a registered avatar's visit to the demonstrations in Second Life and a tracker component which would write to a database each time individual avatars interacted with the demonstrations e.g. press buttons to start animations. When the functionality was completed three initial exemplar projects were created based on augmenting and enhancing existing in-world demonstrations to validate and test the approach shown. In the practical now discussed the student is required to understand the operation of a DC electric motor (Figure 3), its individual components, their interaction and the effects of the magnetic fields.

When the student logs into Moodle they are presented with the list of practical assignments for this course. The student selects a practical assignment and is then given the details of tasks they need to complete inside the virtual world (Fig 5).

Object Name	Task De...
Task 1	Enter virtual world and go to interactive simulation.
Task 2	View video in world about learning outcomes
Task 3	View slide show showing tasks
Task 4	Press button 1 to turn on power
Task 5	Press button 2 to observe magnetic fields

Figure 5 List of tasks for practical assignment 1

They logon to Second Life and teleport to the required location to carry out the individual tasks. Each stage in this process is recorded inside Moodle. As the avatar approaches the interactive demonstration shown in figure 6 the sensor shown detects their presence, transmits this data back to the database and displays this information back to the student.

Figure. 6 Proximity sensor and feedback to student

The Moodle page displaying the practical tasks is updated. In this practical session the student is required observe the DC motor operation and bring up the magnetic fields showing how they influence circuit behavior. As the student completes each

individual task they receive feedback from the Second Life client (Figure 7) and the Moodle page.

Figure. 7 Feedback to student on progress in practical 1

The lecturer can also review the student's progress and level of understanding by reviewing both the level of practical tasks carried out. This demonstrative example shows how virtual worlds and virtual learning environments can be integrated to create top down, narrative led, user learning experiences where user interactions inside the virtual world can be recorded and retrieved for analysis and assessment.

V. FETCH, DECODE, EXECUTE CYCLE OF A CPU

It was the decided to create a more conceptually difficult demo which would help students understand and visualize how the fetch, decode and execute cycle of a CPU functions. Each component of the cycle was broken and replicated as a series of interacting life sized buildings in Second Life and a demonstration created that illustrates the steps undertaken in a CPU during the execution of a single instruction. This highlights the interchanging of control and operand information between the various building blocks of the CPU e.g. program counter (PC), program store (PS), instruction register (IR), operand memory (OM), data mux (DM) and arithmetic and logic unit (ALU). The example demonstration illustrates the six basic CPU steps in the multiplication of two operands. For example, step 1 highlights the role of the PC in providing the index to the PS. In this example the starting hex value of '00' is used to index the PS (Fig. 8).
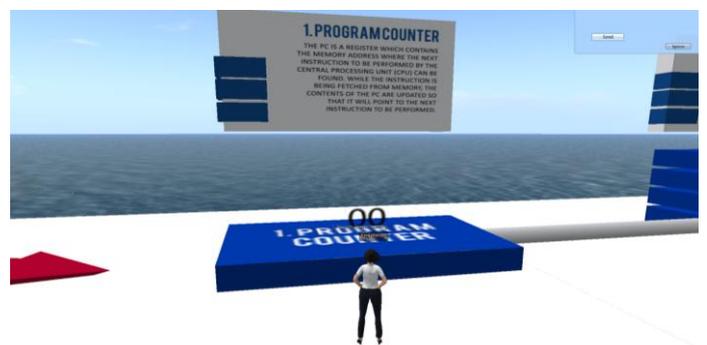

Figure 8 Program Counter with starting hex value of 00

In step 2 the PS outputs the 8-bit contents of its memory (hex value '68') as defined by the index '00' (Figure 9).

Figure 9 Program Store with hex value 68 highlighted

This is fed to the IR in step 3 where it is partitioned into four 2-bit values as A (00), B (10), C (10) and D (01) (Fig. 10).

Figure 10 2 bit values displayed above the Instruction Register

Each of the four 2-bit values plays an important role in the control of the CPU. For example, in step 4 the 2-bit values A and D are used to index the OM. In this example the decimal values '4' and '6' are output from the OM to the data mux where they will be fed into the ALU (Figures 11+12).
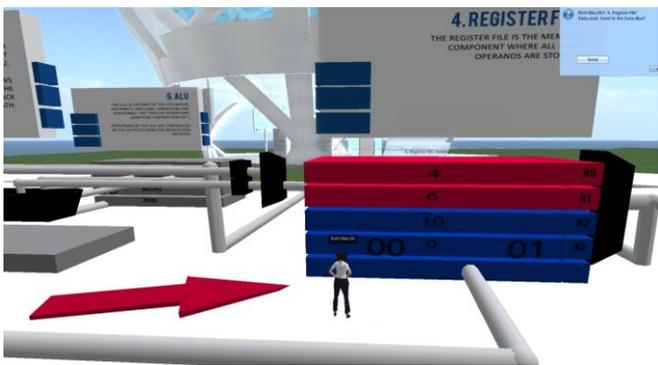
Figure 11 Decimal values 4 and 6 indicated on the register file

Figure 12 Decimal values 4 and 6 being transferred from Reg. to Data Mux

Step 5 outlines how the 2-bit value B (10) informs the DM that the resultant value from the ALU is to be stored back in the OM when the ALU is finished (Figure 13). Finally, in step 6 the 2 bit value C (10) informs the ALU that a multiplication operation is to be performed on the data present at its inputs. The ALU performs the multiplication of 4 x 6 to provide the computed decimal value of 24 (Figure 14).
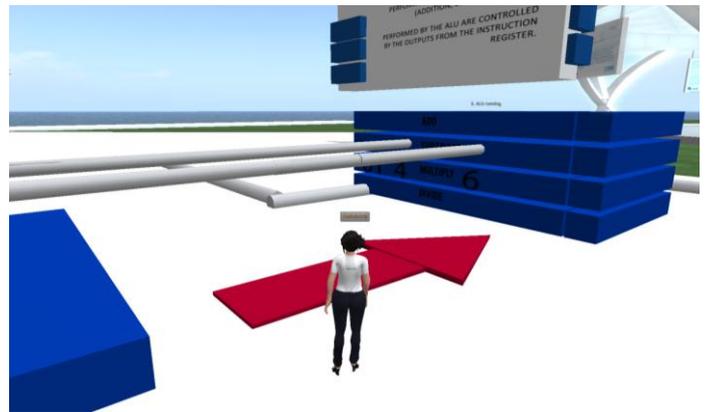

Figure 13 Decimal values 4 and 6 being transferred from Mux to the ALU

These six steps illustrate how the CPU performs a single multiplication operation however addition and other arithmetic operations experience the same sequence of tasks although the instruction data will be different.
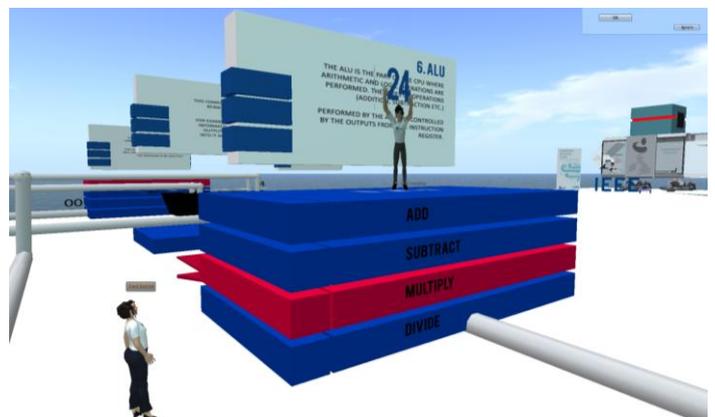

Figure 14: Result 24 displayed from the ALU

To assist in evaluating a student's understanding of the CPU operations and interchange of data a further demonstration was created with an error introduced in step 6 where the data value C being fed to the ALU is different from the original value stored in the IR of step 3 (Figure 15). For example, this time step 1 highlights the PC hex value of '01' which is used to index the PS. In step 2 the PS outputs the value '40' as defined by the address value '01'. Step 4 shows the IR binary contents of '0100 0000' where A='00', B='00', C='00' and D='01'. In this step the 2-bit values A and D are used to index the OM and read the decimal content values '6' and '4'. Step 5 outlines how the 2-bit value B (00) causes the resultant output of the ALU (i.e. when the result is computed) to be fed back into input 1 of the ALU (feedback path options is selected). A simulated fault occurs in step 6 where instead of issuing the C value of '00' (add operation) to the ALU, the C value of '01' (subtract operation) is sent which causes the ALU to perform a subtract operation. This is illustrated in the demo where the output of the ALU subtract operation (6-4) is the value '2' (Figure 16).
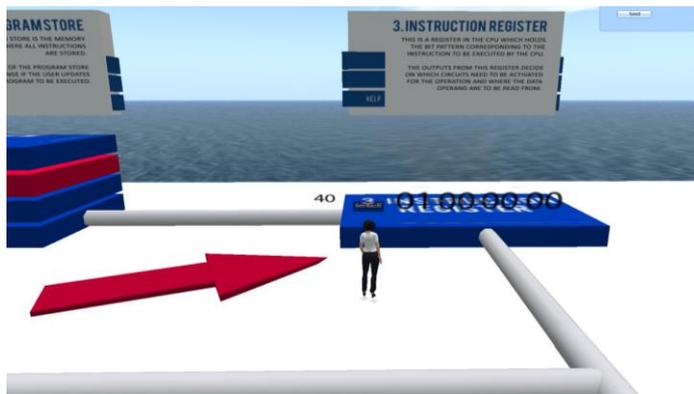

Figure 15 2 bit values displayed above the Instruction Register

Figure 16 Incorrect result displayed on the ALU

This is incorrect as the IR told the ALU to perform an addition operation (6+4) where the resultant decimal value should be '10'. The focus of this fault is to encourage student to examine the IR where the binary instruction is divided into its four 2-bit patterns. This is often the most difficult aspect of the CPU as this is where the instruction information is used to control the OM, DM and ALU components. By tracing back through the sequence of steps, and by comparison with the steps in the 'working' demos, students will be able to see the

fault occurring in the output of the IR data. At the end of the "faulty demo" the student is questioned on their understanding of what went wrong is this simulation. If they answer correctly then they are congratulated. If they answer incorrectly then they are asked to rerun the simulation and take the test again. All of these interactions are recorded in Moodle for review by the student and member of academic staff. This simulation will be used for teaching in the forthcoming academic year and student feedback will be recorded and assessed to evaluate if this approach is an effective way to teach.

VI. INTEGRATING HARDWARE IN VIRTUAL WORLDS

To integrate external hardware into virtual worlds requires the use of a communication protocol that works within the constraints of what is offered by Second Life. In this instance communication and data flow is facilitated by the use of XML-RPC to send data from external hardware devices to virtual objects inside Second Life. This can be achieved by implementing a method that connects to a dedicated Second Life RPC server and then pushes the data through the channel created using PHP scripting with associated Linden Scripting Language (Figure 17).

```
$xmldata = "<?xml version=\"1.0\"?>
<methodCall>
<methodName>llRemoteData</methodName>
    <params><param><value><struct>
        <member>
            <name>Channel</name>
            <value><string>".$channel."</string></value>
        </member>
        <member>
            <name>IntValue</name>
            <value><int>".$intvalue."</int></value>
        </member>
        <member>
            <name>StringValue</name>
            <value><string>".$strvalue."</string></value>
        </member>
    </struct></value></param></params>
</methodCall>";
sendToHost("xmlrpc.secondlife.com", "POST", "/cgi-bin/xmlrpc.cgi", $xmldata);
```

Figure 17 RPC sender php script

External hardware devices are controlled and accessed using a C++ program in conjunction with PHP scripting to allow the hardware to transmit data directly to Second Life utilizing winsock2 to send XML-RPC requests to the Second Life RPC handler (Figure 18).

```
RpcRequesterSL req1("Robotic arm web auto");
// First, we retrieve the SecondLife internal channel to reach the appropriate object
if ( req1.retrieveChannel() ) {
    // Then we send integer and string values
    req1.sendData(12, "string value here");
}
```

Figure 18 C++ RPC sender

The XML-RPC method provided by Linden Labs relies on a RPC server for communication and additionally an internal mechanism that passes messages to appropriate objects using a unique random channel allocated on initialization.

```
remote_data(integer type, key channel, key message_id, string sender, integer ival, string sval) {
    if (type == REMOTE_DATA_REQUEST) { // handle coming requests
        llOwnerSay("I just received string: \""+sval+"\" and integer: "+(string)ival);
    }
}
```

Figure 19 RPC receiver LSL script

The sequence diagram shown in Figure 20 describes the 3 main steps in sending data to virtual objects.
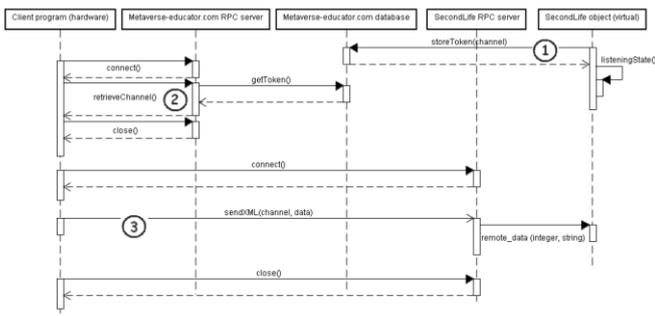

Figure 20 Sending data into Second Life

To retrieve data send from a virtual object it is necessary to use an immediate server. This is required as the Second Life "Server" has to reach the "Client", is this case the PC controlling the external hardware (Figure 21). To do this a HTTP request must be made to the immediate server and the resultant data stored in server database until retrieved.
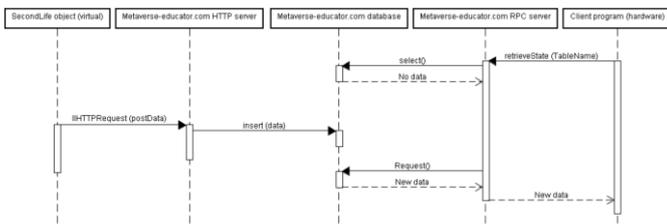

Figure 21 Retrieving data from Second Life

This architecture allows the creation of some highly interactive demonstrations. Figure 22 shows this architecture been used to capture temperature and light values from networked wireless sensors. Figure 23 shows how this data is subsequently visualized in a virtual representation of the building inside a virtual world. The white lights in the building show where lights are on and the green lights show color coded temperature readings.
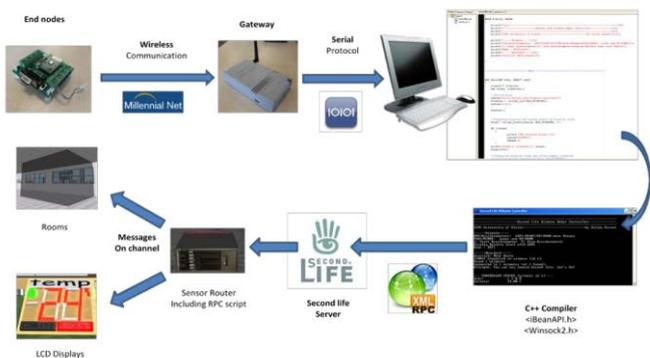

Figure 22 External wireless sensor data captured and visualized

By extending the approach slightly it is possible to utilize more advanced and complex devices. Figure 24 shows an architecture for using a Wiimote controller in the physical world to manipulate a virtual robot arm inside a virtual world. Figure 25 shows the Wiimote controller been used to manipulate and control a virtual robot arm.
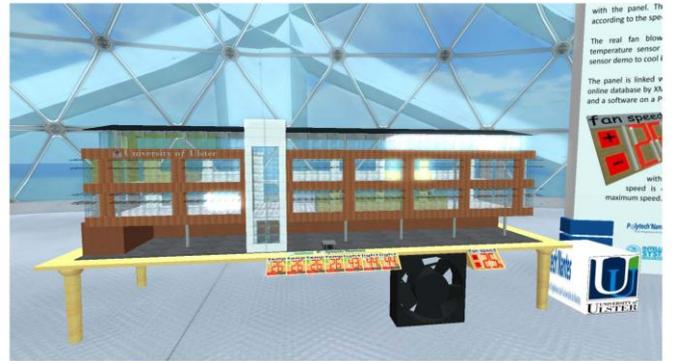

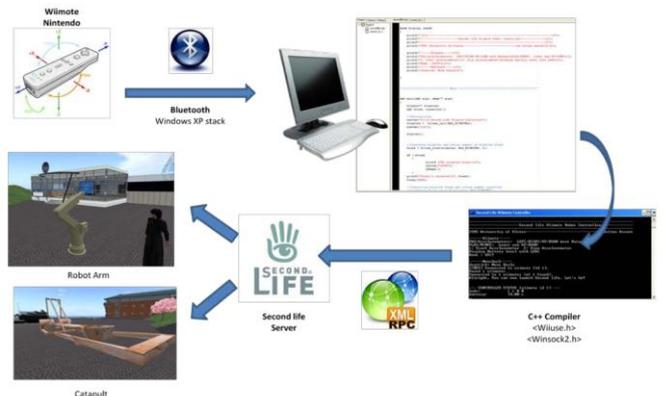Figure 23 Data Visualization in a virtual world

Figure 24 Architecture for Wiimote data capture

Figure 25 Remote manipulation of the robot arm using the Wiimote

These examples show how data can be captured from external devices and rendered/visualized inside virtual worlds allowing a physical entity to manipulate a virtual entity. It is reasonably easy to extend this architecture to facilitate two way communications where interactions with physical objects influence virtual objects and in addition the manipulation of virtual objects cause a change in external physical objects.

Figure 26 shows a hardware washing machine simulator with its virtual representation recreated inside a virtual world. The hardware simulator allows students to create control algorithms to program different aspects of a washing machine operation e.g. selection of wash and spin cycles and control of motor speed. Figure 26 shows the synchronization of the real and virtual simulators where spin cycles selection and button operations are replicated in both the virtual and real worlds. Two way control is facilitated here e.g. resetting the device in the real world will result in a change in the virtual world while pressing the virtual buttons in the virtual world will cause changes in the external hardware.
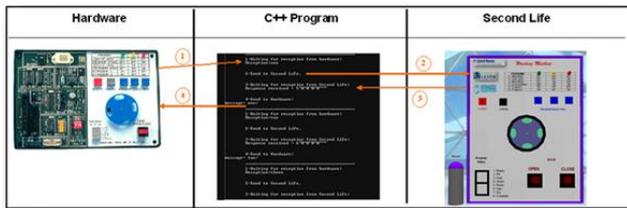

Figure 25 Integrating a machine simulator in a virtual world

Figure 26 Synchronized virtual and real washing machine simulator

Extending the SLOODLE tracking architecture discussed in section 4 to record these hardware interactions allows user interactions in both the real and virtual worlds to be recorded for subsequent review and assessment (Figure 27).

Figure 27 User interactions recorded into SLOODLE learning environment

This functionality and the ability to record and later access user interactions will allow for the creation of more complex simulations similar to the Fetch, execute and decode demonstration in section 5 but with the addition of external hardware and devices.

VI. CONCLUSION

This paper provided an overview on ongoing research at the Intelligent Systems Research Center, University of Ulster, Northern Ireland into the use of virtual worlds and virtual learning environments for teaching. Engineering Education Island was introduced and the Fetch, execute and decode CPU simulation showed that it is possible to make complex and highly interactive demonstrations using in-world tools that can be used for teaching. The paper also demonstrates that it is possible to integrate virtual learning environments and virtual worlds to harness relative strengths of each platform e.g. the course management features of virtual learning environments and the immersive/highly interactive nature of virtual worlds to create engaging learning experiences for students. In addition the ability to integrate, capture and visualize data with two way communication with external hardware devices offers many opportunities to review current approaches to the design of remote laboratories.

REFERENCES

[1] Achterbosch, L., Pierce, R. and Simmons, G. (2007) Massively multiplayer online role-playing games: The past, present, and future. Computers In Entertainment 5:4 — article no. 9

[2] Livingstone,D.,Kemp J.and Edgar,E., Multi-User Virtual Environment & 3D Virtual Learning Environment, ALT-J. 16(3):139-150. 2008

[3] Alexander, B. (2006) Web 2.0: A new wave of innovation for teaching and learning?. EDUCAUSE Review 41:2 , pp. 32-44.

[4] Barab, S. A., Thomas, M., Dodge, T., Carteaux, R. and Tuzun, H. (2005) Making learning fun: Quest Atlantis, a game without guns. Educational Technology Research and Development 53:1 , pp. 86-107.

[5] Dede, C., Nelson, B., Ketelhut, D. J. and Bowman, C. (2004) Design-based research strategies for studying situated learning in a Multi-User Virtual Environment. 6th Int. Conf. on Learning Sciences Santa Monica, California

[6] Dickey, M. D. (2003) Teaching in 3D: Pedagogical affordances and constraints of 3D virtual worlds. Distance Education 24:1 , pp. 105-121.

[7] Mason, H. and Moutahir, M. "Multidisciplinary Experiential Education in Second Life: A Global Approach", in Proceedings of the Second Life Education Workshop at the SL Community Convention, San Francisco, August 20th, 2006. Last accessed 10th August from: http://www.simteach.com/SLCC06/

[8] Second Life. Last accessed 03/2010 From: http://www.secondlife.com

[9] Kirriemuir, J. (2007) An update of the July 2007 'snapshot' of UK higher and further education developments in Second Life Eduserv Foundation , Bath

[10 Kemp, J. and Livingstone, D. (2006) Putting a Second Life 'Metaverse' skin on learning management systems. Paper presented at the Second Life Education Workshop at SLCC San Francisco

[11] Kirriemuir, J. (2007) An update of the July 2007 'snapshot' of UK higher and further education developments in Second Life Eduserv Foundation Bath

[12] Kemp, J. and Livingstone, D. (2006) Putting a Second Life 'Metaverse' skin on learning management systems. Paper presented at the Second Life Education Workshop at SLCC San Francisco

[13] Callaghan MJ, Harkin J, Scibilia G, Sanfilippo F, McCusker K, Wilson S, (Jun 2008) "Experiential based learning in 3D Virtual Worlds", Proceedings of Remote Engineering and Virtual Instrumentation (REV 2008) Conference, Dusseldorf, Germany, June 2008

[14] Melton, J. (2008). Need an LMS? Try the Open Source Package Moodle. Journal of Instruction Delivery Systems 22(1), 18-21.

[15] Slator, B. M., Hill, C. and Del Val, D. (2004) Teaching computer science with virtual worlds. IEEE Transactions on Education 47:2 , pp. 269-275.

[16] Mason, H. and Moutahir, M. (2006) Multidisciplinary experiential education in Second Life: A global approach. Paper presented at the Second Life Education Workshop at SLCC San Francisco

[17] Livingstone, D. and Kemp, J. (2006) Massively multi-learner: Recent advances in 3D social environments. Computing and Information Systems Journal 10:2 — School of Computing, University of Paisley

AUTHORS

Michael Callaghan is a Lecturer in the School of Computing and Intelligent Systems at the University of Ulster, Northern Ireland. He holds a Bachelor of Technology in Electronic Engineering and a Master's of Science in Computing and Design, and is a member of the IEEE and ACM. He leads the Serious Games and Virtual Worlds research team at the Intelligent Systems Research Center at the University of Ulster.

Kerri McCusker is a graduate of the University of Ulster and is currently employed as a research associate in the area of virtual worlds at the Intelligent Systems Research Center within the University of Ulster.

Julio Lopez Losada is a graduate of the Universidad de La Coruña in Spain and is currently a placement student in the Intelligent Systems Research Center within the University of Ulster.

Dr Jim Harkin is a Lecturer in the School of Computing and Intelligent Systems at the University of Ulster. He holds a Bachelor of Technology and a PhD in Electronic Engineering from the University of Ulster, and is a member of the IEE. He is a member of the Intelligent Systems Research Center within the University of Ulster.

Dr Shane Wilson is a Lecturer in the School of Computing and Intelligent Systems at the University of Ulster. He holds a Bachelor of Science in Computer Science and a PhD in Informatics from the University of Ulster. He is a member of the Intelligent Systems Research Center within the University of Ulster.