

Cloud as Context: Virtual World Learning with Open Wonderland

David Parsons
Massey University
Auckland, New Zealand
d.p.parsons@massey.ac.nz

Rosemary Stockdale
Swinburne University of Technology
Melbourne, Australia
rstockdale@groupwise.swin.edu.au

Abstract

Contextual learning takes place in an environmental frame of reference, but sometimes providing suitable learning environments in the real world is not possible. Virtual worlds have the potential to provide immersive learning environments for distributed learners and enable contextual frames of reference that can open up new and previously unimagined learning spaces. These environments can provide experiences that cannot be replicated in the real world due to constraints of time, space and/or resources. However there are complex decisions to be made when selecting a virtual world platform for a specific learning environment. In particular, non-functional requirements can have a significant impact on the suitability of a given tool. In this paper we describe a virtual world learning context created using Open Wonderland to enable geographically dispersed participants to learn about agile software development techniques in a collaborative, immersive workspace. We discuss some important issues in our choice of platform, describe our learning environment, and report on the results of technical usability testing.

Keywords

Virtual world, MUVE, Open Wonderland, agile methods

1. INTRODUCTION

The last decade or so has seen an increasing interest in multi user virtual environments (MUVEs), which provide immersive 3D virtual worlds. MUVEs combine many of the characteristics of instant messaging, IP telephony and social networking web sites, but in an immersive environment where users are represented by avatars to enhance user interactivity. Educators are increasingly looking to such environments to provide new contexts for learning. Some of the perceived benefits are that a MUVE overcomes the need for learners to be physically present, and that resources that can be made available that would otherwise be expensive or difficult, even impossible, to provide. Within a MUVE we can use the power of a 3D software environment to create 'things that never were nor could be' (Brooks, 1995). For example Bainbridge (2007) suggests that undertaking scientific enquiry in virtual worlds enables us to perform experiments that would otherwise be unethical (e.g. modelling an epidemic by spreading disease amongst avatars) or impossible to administer (e.g. manipulating an environment or gathering data from very large numbers of participants.) From a different perspective, a MUVE can act like a 3D Wiki, because of its support for collaborative document

management (Waters, 2009). Thus we can see such tools as the route that can take us from Web 2.0 functionality to the meshing of 2 and 3 dimensional resources in the *Metaverse* (Smart et al, 2007). By shifting the context of our learning into the cloud, creating customised environments that can be inhabited virtually on the web, we can explore entirely new worlds of teaching and learning.

There are, however, major challenges for the educator. A number of significant MUVE educational projects have come and gone as their project funding has terminated and, as Salmon and Hawkrige (2009) note, we cannot know at this stage whether we are at the beginning of a major development in learning technology, or already towards the end of its potential. One critical factor that will influence our success in using MUVEs for education is the difficulty of going beyond those things that are relatively easy to deliver, such as virtual historical or cultural environments that avatars simply explore. Once we move into more challenging areas of delivering complex knowledge, the difficulties of building such tools may create significant barriers to progress (Atkins and Caukill 2009).

In the following section we introduce the context in which we aim to deliver complex knowledge. We follow this with our conceptual framework, and then describe our technical implementation. The paper concludes with the results of testing and evaluation and some suggestions for future work.

1.1 A MUVE Based Learning Context

In order to advance understanding and appreciation of the possibilities of delivering complex knowledge we have taken an existing interactive workshop on agile software development, currently used in a traditional face to face setting with professionals and university students, into the virtual world. Currently, this workshop can only be experienced in the real world, limiting its reach to those who are able to physically attend scheduled sessions. By building the support tools for this activity into a virtual world, we hope to extend its reach to a wider population of learners. As the system develops we also hope to provide significant process and tool support to make it easier to use than the classroom based version.

The choice of the agile workshop as our learning content is motivated by a desire to focus the *use* of teaching technology on the teaching *of* technology, an attempt,

perhaps to eat one's own dog food. Another opportunity in addressing a technology topic is that it can apply equally to students or professionals. A virtual world can provide a rich and productive environment for professional development (Waters, 2009). Since professionals often find it difficult to attend educational provision away from the workplace, virtual world learning can increase access to professional development.

1.2 A Conceptual Framework

In order to provide a conceptual frame of reference for our work, we have taken note of a number of perspectives, including resource views, immersion, game based learning and sociological and courseware perspectives. In previous work we have sketched out an analytical framework for approaching the creation of different types of learning context in virtual worlds from a resource view (Parsons and Stockdale, 2009.) The key assumption of this framework is that there are three levels of resource ranging from basic through interactive to creational. However this perspective alone is not rich enough to fully conceptualise the virtual world learning space. Therefore we have extended our analysis to consider further perspectives.

Gardner et al's (2008a) extensions to Mayes' (1995) framework appear useful in this respect. Again there are three levels; conceptualisation, construction and dialogue (application). In a virtual space these can be mapped to the concepts of immersion; psychological, physical and social. Mayes' original framework is based on the categorisation of courseware, which is divided into primary (subject matter), secondary (environment, tools and tasks) and tertiary (produced by other learners), as examined further in (Mayes and Fowler, 1999). In Figure 1 we have integrated these various perspectives and mapped them to our virtual

world workshop activity. This has helped us to understand to nature of the work that we are undertaking in ways that go beyond simply delivering subject matter (conceptualisation). We can recognise that our work already addresses some core concepts of construction and dialogue, but are also able to identify the key themes that should continue to be the focus of future work. We are also aware of further perspectives that may help to contextualise our aims and objectives.

Activities such as the one described here can address their analysis from a relatively simple emergent process perspective; 'how such a world enables and constrains distributed collaboration' (Orlikowski, 2010 p.132) However we also have an opportunity to apply an analysis from the perspective of entanglements, whereby the agency of the individuals and technology may be helped by their relationships (Orlikowski, 2010). This aspect is one that needs to be addressed in building collaborative process management within the activity.

It is also useful to frame our analysis within the context of immersive virtual worlds being an aspect of (serious) game base learning. Indeed, Wonderland is built upon the Project Darkstar game server. This can cause issues of perception, however, since 'Differing definitions of immersive learning abound and create problems when discussing the subject of educational or serious games.' (de Freitas, 2006 p.6), thus we have to be careful of the terminology we use to make it clear that whilst we may be working in a serious games environment, what we are doing is not really gaming. Nevertheless we can leverage the power of games engines for other purposes. Brooks again; 'Virtual worlds, or synthetic environments, hold great promise for training...It is ironically sadly characteristic of our culture that these promising uses will be enabled, if at all, as by products of

Framework concepts	Types of immersion	Mapping to Agile Workshop
<ul style="list-style-type: none"> • Conceptualisation • Primary courseware: subject matter • Basic resources 	<p>Psychological immersion (abstract space)</p> <p>Deliberately abstract; explorative; self-directed; experimental; multiple representations/visualisations</p>	<p>Pre workshop activities</p> <p>Need to be designed to cater for different backgrounds and experiences, access materials and experiment with tools</p>
<ul style="list-style-type: none"> • Construction • Secondary courseware: environment, tools and tasks • Interactive resources 	<p>Physical immersion (physical space)</p> <p>Deliberately concrete; realistic behaviours; manipulative, role playing; multiple viewpoints; tutor directed; expected outcomes</p>	<p>Workshop context and process</p> <p>Multiple realistic software development roles, organised by moderator, assessed products and learning outcomes</p>
<ul style="list-style-type: none"> • Dialogue • Tertiary courseware: produced by previous learners • Creational resources 	<p>Social immersion (social space)</p> <p>Deliberately situated; localised conversations; identity; reactive avatars; meeting rooms</p>	<p>Workshop environments</p> <p>Custom built context, localised team and developer rooms, co-located avatar conversations required</p>

Figure 1. A conceptual framework for virtual world learning mapped to the agile in wonderland workshop (adapted from Gardner et al (2008a) and Mayes and Fowler (1999))

our desire to be entertained' (quoted in Blundell, 2008). Nevertheless we at least have the benefit of being able to leverage game software for educational purposes, and our aim should perhaps be to try to retain the engagement, enjoyment and challenge of gaming when repurposing such platforms for more serious tasks.

2. IMPLEMENTING THE VIRTUAL AGILE WORKSHOP

The agile software workshop activity has been used in a face to face context on many occasions, and is freely available on the web (Parsons and Cranshaw, 2007). In this workshop, groups of participants work in teams to design a human powered vehicle by drawing individual features, based on user stories, on overhead transparencies. The activity takes place over 3 iterations, each one introducing new agile techniques. The intent of the workshop is that the participants come to appreciate both the meaning and value of these techniques both individually and as a cohesive set by experiencing working on a task which becomes easier as more techniques are made available. Further details on the workshop can be found in (Parsons, 2008).

Since one of the most useful features of a MUVE is that the participants can be remote, the objective of this research was to see if this workshop could be delivered within a virtual world environment so that it could be run for groups of participants who were not co-located. The minimum requirements of this activity were that:

- User stories should be available to the participants.

The system has to have some way of providing a specific set of stories to an individual user based on a selection made dynamically by the person in the role of stakeholder.

- The participants should be able to draw features of user stories.

In the original workshop, participants in the role of developers draw individual features (taken from user stories) on separate overhead transparencies. Some way of enabling participants to draw features was required.

- Participants need to be able to combine their features together at the end of an iteration

The system needs to provide some way of combining individual feature pictures into a single picture. At the end of an iteration in the live workshop, the feature transparencies are laid on top of each other to make the overall vehicle. The system needs to provide some other mechanism so that the separate drawings can be merged together to provide an overall set of features.

2.1 Selecting a Virtual World Platform

Creating a virtual world learning environment is a major investment for any educational institution, and the choice of platform is a significant commitment to the evolution of any project in this area. Using a commercial MUVE such

as Second Life requires investment in virtual land, the scale of which needs to be significant if it is to provide adequate access to virtual resources. Choice of an open source platform reduces initial outlay but implies other costs in hosting dedicated servers and providing the human resources to further develop the software.

In our research to date we have worked with Second Life, Open Simulator and Project Wonderland (now Open Wonderland) in an effort to identify the most appropriate platform for our own work. After reviewing the work of others and from our own experiences we decided to work primarily with the Wonderland platform as the best option for implementing the core functional requirements we have outlined above. Gardner et al (2008a) have outlined some of the key pros and cons of choosing Wonderland as a virtual world platform for teaching. On the plus side, Wonderland is open source and extensible, and more platform agnostic than many open source alternatives due to its Java codebase. It also enables greater control over resource access, privacy and security than the commercial Second Life with its publicly shared infrastructure. In contrast to Second Life, the primary intent of the Wonderland platform is that it can be tailored and integrated by organizations within their own infrastructures. (Gardner et al, 2008a). One of the potential issues with this is that the system works well within an organisational firewall but there may be problems providing equal access for remote users. For example in order for Wonderland to function correctly a large number of ports must be opened on the server, which may be regarded as creating potential vulnerabilities. At a minimum, Wonderland requires TCP ports for the Wonderland client web connection and the main connection to the Wonderland server, a UDP port for audio signalling and ideally another 200 UDP audio channel ports.

There are, however, a number of alternative open source virtual worlds, In addition to Wonderland, the Immersive Education Initiative's 'Platform Ecosystem', which consists only of freely available open source technologies, also includes Open Cobalt, Open Simulator (OpenSim) and realXtend, along with an enhanced descendant of the open source Second Life viewer (Media Grid, 2005). How the open source virtual world market will play out over the next few years remains to be seen. Our choice of Wonderland was partially influenced by its origins at Sun Microsystems and the support of that vendor. However in February 2010 following the acquisition of Sun by Oracle Corporation, support for Project Wonderland was withdrawn. The project has since been renamed Open Wonderland and is now entirely supported by an open source community.

Kappe and Gütl (2009) list the following non functional requirements as important in their own virtual world environment; Security, Flexibility, User Experience, Open-Source (customizable) code and Cost effectiveness.

Although their functional requirements differ from ours, these non functional requirements were also important in our selection of a particular virtual world platform. It is notable that Kappe and Gütl (2009) used these criteria to select realXtend as their platform. Therefore, as we note in our conclusion, our decision to work with Wonderland over that last two years or so was taken from a pragmatic standpoint, but this platform choice may change in the future.

2.2 System Description

The virtual world that we have created consists of a large virtual building with four separate team workspaces. Figure 2 shows the typical view of an avatar when the client is first launched. The user begins at the front of a four winged building, each wing containing a separate project team room. In the centre of the quadrangle is a large ‘integration test’ whiteboard which is used by all teams for the assessment phase of each round of the workshop.



Figure 2. The exterior of the workshop building, showing the entrance to the four main team areas and the ‘integration test’ whiteboard.

Each team workspace comprises a number of developer rooms that each contain a whiteboard and a PDF viewer showing user stories. There is also an editable whiteboard where individual features can be drawn. Figure 3 shows an avatar inside one of the team areas. Separate developer areas are visible containing the whiteboards and story boards. There is also a shared notes board in the central (shared) area.

When an iteration is in progress, developers will be drawing allocated user stories as individual features on whiteboards (one story = one feature = one drawing). Figure 4 shows an example where the user story ‘The driver must be protected from attack by wild animals’ is being implemented. The developer has drawn a cage-like structure that can be applied to the vehicle.

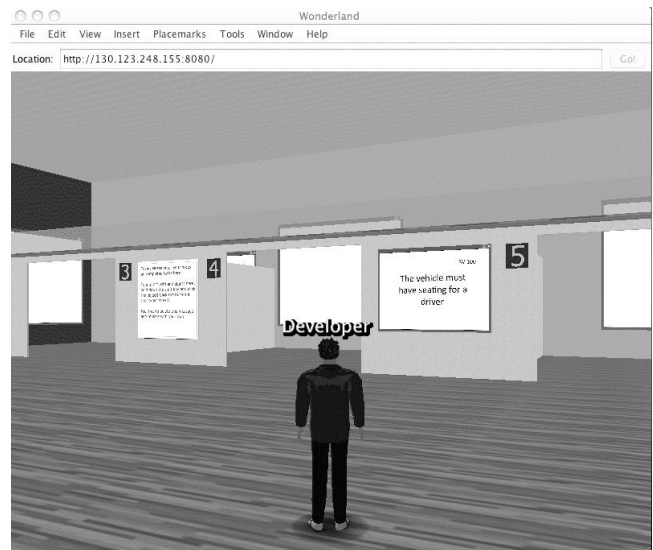


Figure 3. Outside the developer rooms, showing whiteboards, notes board and user stories in the PDF viewer.

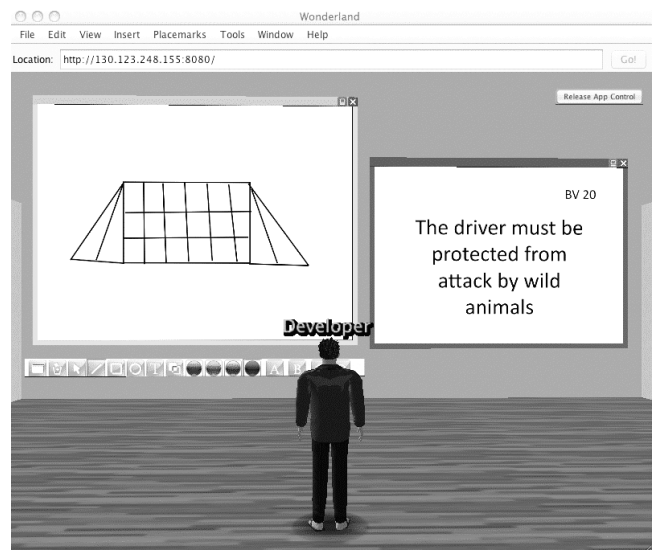


Figure 4. Implementing a user story on a developer whiteboard

Each time a developer completes the implementation of a specific user story, s/he commits that feature to the team repository by clicking the appropriate team button (A, B, C or D) on the whiteboard tool bar. This will clear the whiteboard ready for the next feature. Figure 5 shows the developer working on the next user story, ‘The vehicle must be able to travel over rough and uneven ground’. In this example the developer has drawn large wheels with caterpillar tracks. Again, on completion of the drawing, the feature will be committed to the team repository.

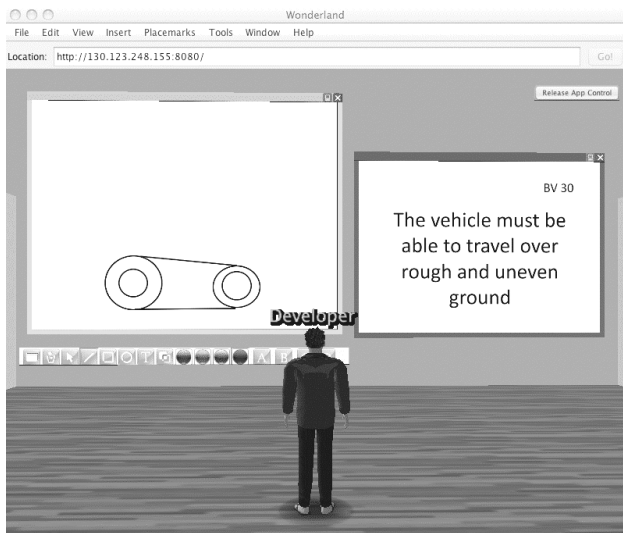


Figure 5. Implementing a second user story after the first has been committed

In a complete iteration, there will be four developers simultaneously working on a number of user stories. However to simplify the figures, the example used here uses only the two story implementations from Figures 4 and 5. At the end of the iteration, the teams will gather round the integration whiteboard. Each team's overall design, consisting of all the individual features layered together, can be viewed by selecting the team's identifier from the toolbar. Figure 6 shows the two features from our examples being combined for the team. Note that the toolbar on this whiteboard does not contain any drawing tools. It is only intended to integrate all the features from a single team and display them as a single image.



Figure 6. Combining user stories on the integration whiteboard

3. TESTING AND EVALUATION

This section describes a number of practical tests that we carried out on the initial deployment of the system. The

evaluation here is somewhat basic, focused only on some tests of system usability on different platforms, but nevertheless it may be helpful to others who wish to deploy a Wonderland virtual world for their own purposes.

3.1 First Evaluation

In our first evaluation, we deployed a fairly large virtual world and tested the configuration with a group of nine postgraduate students all working in a Windows XP lab within the same university network as our Wonderland server. The participants were asked to connect to the Wonderland server, navigate the virtual space, and interact with the design tools. The session lasted about 30 minutes. It began with an explanation of the project, its purposes, and the reason for undertaking the test. The participants were asked to carry out a number of interactions with the system (e.g. view multiple stories on the PDF viewer, draw on the whiteboard) and report back on their success in completing these tasks. The purpose of this test was (a) to ensure that the server configuration was able to support multiple users, and (b) to provide general feedback on the usability of the system. A number of performance issues were identified relating to the client side application. One of these issues was that the first time a client machine connects to Wonderland there is a large set of Java archive downloads that take place before the client starts. This improves on subsequent connections because the archives will have been locally cached. However as the evaluation progressed it became clear that the client application improved in performance over time even after this initial download, presumably as it was able to download and cache more features of the virtual world incrementally. We also observed some apparent contention with multiple simultaneous logons, whereby some participants were unable to successfully start the client on the first attempt, but were able to do so later. We suspect this may be related to some known issues with collision detection in Wonderland, with multiple avatars being created at the same point in the virtual world and colliding with each other. The suggested fix for collision detection issues is to disable it in the client, but this can only be done once the client has started. Therefore some staggering of client start-up may be required. Overall the response from the participants was that the workshop tools looked interesting but there were significant performance issues that made it hard to perform meaningful evaluation of these tools.

Following the first test session, we experimented with a number of configuration aspects, including making a much smaller virtual world to see if this affected the initial start-up issues. We had run a number of tests of system functionality within our own university network, but were also aware that the university proxy servers and firewalls could limit the practicality of running virtual world workshops for clients outside of this network. Having gone through the necessary procedures for making our server URL and ports accessible from the wider Internet, we needed to have a realistic test of whether we had successfully configured the server for public access. For

this reason the next usability test was carried out in a different university located in another part of the world.

3.2 Second Evaluation

For the second test, twelve postgraduate students voluntarily attended a session in which they brought their own laptops and attempted to connect to the Wonderland server, navigate the virtual space, and interact with the design tools. The session lasted about 90 minutes. Again it began with an explanation of the project and the participants were asked to carry out a number of interactions with the system and report back on their success in completing these tasks. The purpose of this test was (a) to ensure that the server configuration was able to support multiple users at a remote site outside of the university network firewall, and (b) to provide feedback on performance of the client software on different machines running a range of operating systems. In the tests, the following operating systems were being used by various members of the group; Windows XP, Windows Vista, Windows 7 and Linux. The test demonstrated that there were no issues with multiple external connections to our server and that all the basic connection channels were available. However we were unable to use the voice tools because of the number of ports that were open through the university firewall. Other than this we found that the performance of the system was comparable with running inside the local network. This further underlined that the main performance issues were not related to factors such as network latency or server response, but in the client side application.

Although general feedback on the system was positive in term of its intent, and the tools we were trying to provide, most of the negative responses related to aspects of the system that were generic to using the Project Wonderland client with its slow start-up. However there was also some useful feedback on where 'drag and drop' functionality was needed within the workshop tools.

A number of specific usability issues related to navigation did emerge as a result of these participants exploring the virtual space. One problem was that the modified virtual space was now too small, making it hard to navigate around the walls, whiteboards and PDF viewers. Users found themselves jammed against virtual walls or going through them, making it hard to properly work within the space. Minocha and Reeves (2010) stress the importance of designing for accessibility both in look and function. They stress the importance of having plenty of space to navigate between things, which is easy to create in a virtual world without physical constraints. Although the use of a small space in our prototype had been prompted by performance concerns, making the space smaller to test if the size of the virtual world has an effect on system performance that would impact on usability, this possibility was not borne out. As a result of our tests it became clear that most latency problems were not related to the size of the virtual world but related to critical differences in clients running

with different graphics configurations, both hardware and software.

3.3 Follow-up Testing

As a result of our test sessions a number of questions were raised which we attempted to resolve in series of follow up tests. One issue that arose was whether the usability of the client was dependent on the operating system being used despite the supposed platform neutrality of Java. In the second test the two students who were running Linux clients had the best user experience, finding the client responsive and easy to use. Two other students who were running Windows 7 were unable to successfully run the client application, which failed to fully load. Between these two extremes, the Windows XP and Vista users found the client useable, but not as responsive as the Linux client. Whilst such issues may seem relatively trivial, they were important to our evaluation group. One responded; *'Last but very important issue you need to fix...Windows 7 as I was struggling...and could not participate as fully as I wanted.'* Therefore as a result of these observations and responses we ran a number of subsequent tests, where we attempted to diagnose the apparent issues associated with the client operating system.

We re-tested the system with both Windows 7 and Linux clients and analysed the system logs. As a result of this analysis it became clear that the problem with some Windows 7 clients was due to an issue with graphics drivers so this could be easily resolved by updating the drivers. It also appeared that client performance was largely dependent on the quality of the graphics card in the computer. Running a Linux client on a standard graphics card revealed performance comparable with Windows clients. As Roebuck (2008) states, Wonderland ideally requires a 'modern game hardware' client with powerful graphics capability. Thus, in selecting a client, hardware capability (with the exception of graphics drivers) is more important than software.

Another issue that became evident in our follow up testing on different platforms was launching the client from the initial web page. Wonderland uses Java Web Start, which requires that the Java run time version 6 or above is installed on the client. In some tests we found that the browser client may not automatically map the web start file (Wonderland.jnlp) to the required 'javaws.exe.' web start application, and manual file association was required.

Table 1 summarises the contexts and issues that arose from the evaluation sessions and the possible solutions to problems that became evident during these sessions. As a result of these test and results we are now able to ensure that clients can connect to the server with minimal technical difficulty, enabling us to move onto an evaluation phase that focuses more on the user experience and learning outcomes.

Table 1. Lessons learnt from evaluation sessions

Session Details	Issue	Resolution
9 participants Single client platform Large virtual world Local site	Slow client startup	Allow time for clients to load before activity, 'prime' clients by ensuring client archive files have been previously downloaded. Reduce size of virtual world?
	Apparent avatar collision /contention issues with simultaneous logons	Stagger logons to avoid collisions, turn off collision detection as soon as possible
12 participants Multiple client platforms Small virtual world Remote site	Apparent differences in client operating system performance	Tested specific platforms, diagnosed log files, detected issues with graphics cards and drivers
	Navigation problems	Increased space between objects and walls
	Application fails to start	Ensure correct version of Java is installed and 'jnlp' files mapped to javaws.exe

4. RELATED WORK

There is a huge amount of activity in using virtual worlds for learning, so here we focus on work that relates specifically to virtual world activities concerning software engineering education and/or projects that use the Wonderland platform. Two educational projects that have previously used the Wonderland toolkit are MiRTLE (Mixed Reality Teaching and Learning Environment) (Gardner et al 2008b) and Darkstar University (Sun Microsystems 2007). MiRTLE (a project of the University of Essex) provides a virtual classroom environment. A remote lecturer is recorded with a web cam and this is beamed to the students. Students can ask the lecturer questions as if they were attending a real-world lesson. Darkstar University is a business venture to provide a virtual learning environment to universities and colleges from countries including the U.S., U.K., China, Canada and Australia.

Using the Second Life platform, some previous work has been undertaken to deliver both computer science education and software engineering games. Ritzema and Harris (2008) used the Linden scripting language within Second Life to teach introductory object oriented and event driven programming. They also used Second Life 3D simulations with more advanced learners who were working with physical adder circuits and Mealy machines in a real world lab. Virtual world simulation tools for these devices were used to support the physical world activities.

Ye et al (2007) developed two multi-player online versions of software engineering educational games from other sources, namely Groupthink (from M.I.T.) and SimSE (from the University of California Irvine.)

Bringing both virtual world software engineering activities and Wonderland together, the WikiDev3D project uses Wonderland to visualise the growth of software artefacts and their relationships over time in a project, using a city metaphor. Users can replay the development process from multiple perspectives to assist their understanding of software development processes (Stroulia, 2010.)

5. CONCLUSION AND FUTURE WORK

Our work to date has brought us from early work with Second Life, though evaluation of multiple virtual worlds, into an iterative cycle of development with Open Wonderland that has already progressed through two significant prototyping exercises with different versions of the platform. We have been able to undertake usability testing at the technical level that has guided us in our current work to bring the virtual world workshop to the verge of practical utility. There is still, however, much work to be done to be able to demonstrate the true potential of this type of learning context. Now that we have overcome most of the technical issues in deploying our workshop, we need to further refine to tools and integrate more automatic activity management prior to further user evaluation. Our current implementation lacks some of the required 'conceptualisation' content within Wonderland itself. Currently our informational materials are locally hosted in a Wiki using JAMWiki. However this material is not suitable for the required psychological immersion in an abstract space that might approach the characteristics of a suitable metaverse. Future work will include building an environment similar to Second Life's 'Orientation Island', that will enable participants to familiarise themselves with the tools and activities within the workshop prior to participating in a managed workshop process. The 'physical immersion' aspect of the system also needs further development to ensure that it fully delivers the workshop's learning objectives, while the 'social immersion' will be an interesting focus for further qualitative evaluation. We also need to be mindful of the ongoing development in virtual world platforms, and continue to monitor if Open Wonderland remains our platform of choice.

Looking further ahead, continuing technical developments mean that mobile clients to virtual worlds, including Java worlds, are a realistic prospect (e.g. Rodrigues et al., 2006). Such clients open up new potential for augmented reality virtual worlds that can blend the virtual with the actual. These developments will further challenge our ability to fully leverage their educational potential.

6. ACKNOWLEDGMENTS

This research was financially supported by the Institute of Information and Mathematical Sciences, Massey University.

We also wish to acknowledge the major contribution of research students Matthew Cotterell and Joshua MacLauchlan to the development of this project.

REFERENCES

Books:

- Atkins, C. and Caukill, M. (2009) Serious Fun and Serious Learning, in Molka-Danielsen and Deutschmann (eds.) *Learning and Teaching in the Virtual World of Second Life*, Tapir Academic Press, pp 78-89.
- Brooks, F. (1995) *The Mythical Man-Month, Essays on Software Engineering* (2nd Edition), Addison-Wesley
- de Freitas, S. (2006) Learning in Immersive Worlds: A review of game-based learning, JISC.
- Mayes, J. (1995) Learning Technologies and Groundhog Day. In W. Strang, V. Simpson and D. Slater, *Hypermedia at work: Practice and theory in Higher Education*. University of Kent Press: Canterbury

Journals:

- Bainbridge, W. (2007) The Scientific Research Potential of Virtual Worlds, *Science* 317, 472
- Blundell, B. (2008) Images in Space, *ITNOW*, November 2008, 8-11
- Gardner, M., Scott, J. and Horan, B. (2008b) MiRTLE, *EDUCAUSE Review*, 43, 5
- Mayes, J. and Fowler, C. (1999) 'Learning technology and usability: a framework for understanding courseware', *Interacting with Computers* 11, 485-497
- Minocha, S. and Reeves, A. (2010) 'Designing for Usability and Accessibility in 3D Virtual Worlds', *IEEE Learning Technology Newsletter* 12, 2 4-6
- Orlikowski, W. (2010) 'The sociomateriality of organisational life: considering technology in management research', *Cambridge Journal of Economics* 34, 125-141
- Ritzema, T. and Harris, B. (2008) The Use of Second Life for Distance Education, *Journal of Computing Sciences in Colleges* 23, 6 110-116.
- Salmon, J. and Hawkrigde, D. (2009) 'Editorial: Out of this World', *British Journal of Educational Technology*, 40, 3 401-413

Conference Proceedings:

- Gardner, M., Scott, J. and Horan, B. 2008a. Reflections on the use of Project Wonderland as a mixed-reality environment for teaching and learning. *Proceedings of Researching Learning*

in Virtual Environments. The Open University, UK, pp. 130-141.

- Kappe, K. and Gütl, C. 2009. Enhancements of the realXtend framework to build a Virtual Conference Room for Knowledge Transfer and Learning Purposes. *World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA 2009)*. Honolulu, USA, pp. 4113-4120.
- Parsons, D. 2008. The Agile Technique Hour. *9th International Conference on Agile Processes in Software Engineering and Extreme Programming*, Limerick, Ireland. p. 246-247.
- Parsons, D. and Stockdale, R. 2009. Agile in Wonderland: implementing a virtual world workshop activity, *20th Australasian Conference on Information Systems*, Melbourne, Australia.
- Rodrigues, M., Barbosa, R. and Mendonça, N. 2006. Interactive Mobile 3D Graphics for On-the-go Visualization and Walkthroughs. *SAC'06*, Dijon, France, pp. 1002 - 1007.
- Ye, E., Liu, C., and Polack-Wahl, J. 2007. Enhancing Software Engineering Education Using Teaching Aids in 3-D Online Virtual Worlds, *37th ASEE/IEEE Frontiers in Education Conference*, Milwaukee, WI, pp. 8-13.

Electronic Sources: Websites

- Media Grid, 2005. *The Immersive Education Initiative*. Available at: <http://immersiveeducation.org/> [Accessed 10 September 2010].
- Parsons, D. and Cranshaw, M. 2007. *The Agile Hour*. Available at: <http://www.massey.ac.nz/~dpparson/agilehour.htm> [Accessed 10 September 2010].
- Roebuck, K. 2008. *Project Wonderland in the Age of Immersive Education, Dr. Dobbs Mini-Summit*. Available at: http://wikis.sun.com/download/attachments/14353293/Sun_ImmersionEd_DrDobbs.pdf [Accessed 10 September 2010].
- Smart, J., Cascio, J. and Paffendorf, J. 2007. *Metaverse Roadmap, Pathways to the 3D Web*. Available at: <http://www.metaverseroadmap.org/MetaverseRoadmapOverview.pdf> [Accessed 10 September 2010].
- Stroulia, E. 2010. *WikiDev3D*. Available at: <http://blogs.openwonderland.org/2010/05/25/wikidev3d/> [accessed 10 September 2010].
- Sun Microsystems. 2007. *New Frontiers in Virtual Learning, EduConnection Newsletter*, October. Available at: <http://www.sun.com/emrkt/educonnection/newsletter/1007eduinsight.html> [Accessed 10 September 2010].
- Waters, J. 2009. A 'Second Life' For Educators. *THE Journal* January 2009. Available at: <http://www.technteach.info/asecondlifeforeducators.pdf> [Accessed 10 September 2010].