STUDENTS’ EVALUATIONS OF 3D WORLDS

Paul D. Chandler
Research Fellow in Multimodal Pedagogy
University of New England
paul.chandler@une.edu.au

3D virtual worlds are with us now. Systems such as Second Life, OpenSim, Active Worlds, Open Wonderland and World of Warcraft are known to teachers and students alike. Some schools have established an online 3D presence in one or more of these worlds, and students have been engaged in exciting projects building virtual worlds. For instance:

• Corpus Christie Catholic High School presented at AARE 2009 on their project to use 3D multi-user virtual environments as a tool for transition to secondary education
• Northern Beaches Christian School were prominent at ACEC 2010 with their explorations into Second Life.
• Sylvania High School presented at “Listen2Learners” in 2010 on their virtual café learning space: researching sustainable cafés and learning spaces, training for virtual world construction, reflecting on design processes and designing and constructing a sustainable café learning space.
• Stafan Scutt and Dale Linegar are reporting at this conference of extensive experience over some five years.

My perspective is this: if I want students to explore in (or with) a 3D virtual world, the technology needs to work without too much fuss and the students need to be able to find their way around. Ideally, I’d like:

• to give the students a memory stick with the software on it and they can commence exploring immediately (ie no installation required – take the technician out of the equation)
• allow them to take it home to further explore; apart from portability on a memory stick, there are two sticking points:
  o need an internet connection which is fast enough and not restricted
  o platform dependency (Mac at school + PC at home = problems)
• not to be troubled by what “issues” (or people) might be encountered who are not part of the school community
• not to set up and run a server (eg OpenSim) myself

A “universal solution” is not presently available, but I am in the process of looking into three systems with students, which is what is reported here.

State of the art (or not)

As others have observed, the present state of the 3D virtual worlds is somewhat similar to where we were about 20 years ago with accessing online services or the internet. That is, you have a dedicated service provider (once – this would have been a service like AOL, now – a service like Second Life) and on your computer you have a relatively lightweight client (once – a modem terminal program, now – a Second Life client). All the heavy duty work is done at your service provider’s end, and your client is the ‘window’ into it. If you are unable to be online you cannot access your online world. If your service provider becomes too busy or fails, you can’t get online either. This is the client-server approach, and is presently the dominant model for accessing 3D virtual worlds. As others have observed, interaction between objects and avatars in a virtual world is primarily through geometric proximity. Ideas of “ownership” and “interaction” are a poor resemblance to the real world. In other words, not only does the “architecture” of systems for 3D virtual worlds have limitations, there is still much scope for developing the internals of the world itself.
An alternative to the client-server approach is the peer-to-peer (P2P) model. Even though much of Web 2.0 activity is increasingly “centralised” (e.g., Facebook, Skype, Twitter), the P2P model is reasonably well known. The WWW is, in essence, P2P because anyone can add a new host which can serve pages; BitTorrent filesharing is entirely P2P; Diaspora* is social networking using a P2P architecture.

There is considerable interest in implementing 3D virtual worlds using a P2P architecture (e.g., Solipsis, OpenCobalt, OpenQwaq).

For students in today’s classrooms and homes, the practical implications of a P2P arrangement are that all they need is a computer with the software installed – “their” virtual world exists on their computer, in the same way that a word processing document sits in a folder somewhere. They do not need continuous access to an online service, a high-speed link to that service, a login to that service, etc. The “peer-to-peer” possibility arises when students are able to share their virtual world with other users – something like creating a wiki.

The exploration

It is fair to say that P2P 3D worlds are in their infancy. They may offer a more accessible means of exploring virtual worlds to a wider range of schools and students than arranging for subscriptions to online services, overseeing online behaviour or managing servers. But before we can devise educational projects which use this technology, we must firstly ask some fairly fundamental questions:

- do students find them easy to use?
- can their “find their way around” easily enough?
- what features do they readily stumble into?
- what features are more obscure and necessitate induction?
- what are the difficulties which they encounter?
- what are the possibilities they can see?

This investigation involves making three P2P 3D virtual world environments available to students, and challenging them to explore, collaborate, learn and make recommendations. Eight year 5/6 students have been engaged to explore three different software platforms. These students are from a government primary school, the majority of them being in the “e-learners” group, and therefore used to self-directed learning, completing their work online, and exploring technologies as necessary.

OpenCobalt (http://www.opencobalt.org)

Providing the inspiration for this project, it is the only genuine P2P virtual world, but also the least developed. We have not explored this with students as yet (hoping that a more stable and useful product will be available by the time we get to it). It is multi-platform and does not require installation. It runs “straight off the stick”. There are (at this stage) no “sample worlds”, but you can import objects directly from Google’s library of kmz objects. The introductory videos on the site are well worth looking at.

KindleLabSolo (http://kindlelab.com/home/?content=solo)

This is a re-packaged Snowglobe (SecondLife/Opensim browser) and Opensim server, which runs “straight off the stick” with one click, but is only for Windows systems. In practice, some unreliability has been found when moving from computer to computer, possibly because some data (created while the programs are running) is actually stored in working directories on the local computer. Nevertheless, if you want a nearly instant “second life experience” without worrying about logins, security or internet access, it’s very clever.
We have had most experience with KindleLabSolo. Students have found it quite easy to learn how to do the following:

- Navigating and flying
- Creating a snapshot
- Building a object
- Moving an object
- Working with the inventory (body parts, clothes, etc)
- Creating a landmark
- Importing objects
- Not becoming a ghost
- Returning “home”

They found it more challenging to work out how to

- Not becoming a ghost

Frustrations were experienced when a world would work one day, and not the next, for no apparent reason. That is, login would be denied, the client would take ‘infinite’ time to log in, the client would crash moments after logging in. In these cases, we were unable to find a ‘fix’ so simply started again. About half the group suffered at least on instance of this; they were not fussed, but that would easily change if they had lost school work!

We’d hoped to be able to inter-connect our worlds. Setting up subsequent users on any server is quite easy. Finding the changes to the configuration files to allow connections from computer other than the local one wasn’t, so finding a neat solution to that remains on our wish list.

**Solipsis** ([http://www.solipsis.org/](http://www.solipsis.org/))

A French initiative whose development has concluded. It is only available for Windows platforms and requires straightforward (but extended) installation\(^1\), so it is not portable. The user must explicitly run two servers\(^2\) and then a client. Interaction with one another’s worlds requires editing a configuration file\(^3\), something which would certainly be messy with dynamically allocated (and therefore changeable) IP addresses. So it really is single-user unless you know what you’re doing.

We have only begun to explore solipsis, but I rather suspect that students will find it very easy to use, but deem it suitable for junior students. It would also seem that all the ‘world building’ must take place in a 3D editor outside of the world environment, so it will actually prove limiting in that respect.

**Conclusion**

We are still exploring, so the jury’s out. A final verdict will probably depend on how user-friendly and robust the current generation of OpenCobalt proves to be, and how easily KindleLabSolo can be configured to accept traffic from a non-local machine.

I remain convince that current 3D virtual world technologies are not the ‘high point of the art’, and we should keep alert for developing technologies.

**References** (in addition to information found on the Web sites above)


\(^1\) The installation seems to assume that the library MSVCP71.DLL will be already in a ‘sensible place’, but as often as not I have had to obtain a copy from another computer and place it in Solipsis’ navigator folder.

\(^2\) One of these, the worldsserver, by default listens on localhost, which means it won’t respond to any request from another machine. This is good for security, but if you are hoping for any interaction between users, it needs to be started with command line options H 192.168.0.2 -P 8550

\(^3\) The URLs for the worlds servers is defined in uiworlds.html in the worldsserver folder, and needs to be edited manually.