3D Spaces in Software Engineering: From K-12 to Life Long Learning

Heinz Dreher
School of Information Systems
Curtin Business School
Bentley, Western Australia
h.dreher@curtin.edu.au

Torsten Reiners
Institute of Information Science
University of Hamburg
Hamburg, Germany
reiners@econ.uni-hamburg.de

Naomi Dreher
School of Information Systems
Curtin Business School
Bentley, Western Australia
n.dreher@curtin.edu.au

Carl Dreher
School of Information Systems
Curtin Business School
Bentley, Western Australia
c.dreher@curtin.edu.au

Abstract: Despite continuing debate regarding the utility of virtual worlds, their use has continued to pervade the worldwide marketplace. An important consideration is that the current K-12 student cohort comprises digital natives who are clearly the major group of users represented in virtual worlds, either in games or social online communities (Prensky 2001). Increasingly this cohort of students will expect to use virtual worlds to learn, and educators are wise to prepare. Indeed 3D Spaces will likely become one of the major platforms for distant learning as well as for virtual teams of software engineers to both work and socialize. In the context of software engineering, this paper discusses the benefits for early movers to integrate 3D Spaces in the classroom, project meetings and vocational training. The focus is set on life-long learning and the specific methodology of 3D Space integration is demonstrated over all stages – from childhood to the workplace.

Introduction

The pros and cons of virtual worlds, synthetic worlds, alternative realities, or in short, 3D Spaces are discussed by experts from almost all fields (Delwiche 2006; Dickey 2005). Without a doubt, 3D Spaces provide new possibilities for learning scenarios, especially in distance education or blended learning. Instead of having podcasts and digital documents with little-to-no interactivity or socializing opportunities for students, a 3D Space can be synchronously explored by all participants (e.g., students and lecturers) with a high degree of communication and interaction. Examples are found in various classes around the world demonstrating that a virtual classroom with compulsory attendance is viable even though the students might be situated in several countries: see Lamont (2007), Conklin (2007), Erlenkötter et al. (2008), Gregory et al. (2009), and Dreher et al. (2009a) for some examples.

Many first movers and visionaries (some of whom started in the inflated expectation phase of the Gartner Hype Cycle; see The Equity Kicker 2007) are responsible for the hype of 3D Spaces. This is especially the case for the so-called social virtual worlds that were being hyped around 2006 (Hayes 2009). Due to large media coverage and many (false) expectations, the corporate growth in virtual worlds became too large for demand and resulted in

the predicted climax of the hype cycle. Nevertheless, in many cases the decrease in coverage caused a consolidation of experience and reconsideration of strategy to maximize the return on investment – regarding money and especially time. Instead of focusing on the obvious social advantages of virtual worlds (by using the media for meetings, communication and presentation of classes in the classical format of slide presentations), ideas and expert knowledge from all over the world have been applied to create advances in learning models, scenarios, and experiences. As predicted by the Gartner Hype Cycle and discussed by Hayes (2009), 2009 is the year of "Education and Business Maturity", a forecast that can be observed simply by having a look at current projects in institutions world-wide and by seeing the funds being invested in existing or new technologies and application development. For example a major project by T-Systems (2009, p. 40) introduced the device EduKey which "gives students and teachers secure, biometric access to the EduNex portal from any PC with an Internet connection, via a personal profile. This enables self-defined, flexible learning using the multimedia and interactive education platform." Another example pertinent to educational institutions is the integration of technology such as SmartBoards to facilitate an immersive learning experience through interaction with virtual 3D content in a new and very intuitive fashion.

We continue with a short overview of virtual worlds before we describe below (in Section "Virtual Worlds in the Context Software Engineering") our current exploration of 3D Spaces focusing on software engineering from several perspectives: (1) allowing K-12 students to have first experience with programming environments, (2) teaching a class in software development and implementation at a university level, (3) teaching in the context of life-long learning or offering training / recruiting courses for companies, and (4) software development projects in distributed (virtual) teams or off-shoring scenarios. In Section "3D Spaces and Immersion", we briefly discuss how 3D Spaces support immersion and how this can be exploited. The conclusion discusses the requirements for further integration of 3D Spaces and presents a short overview of our next research focus.

Virtual Worlds

The number of virtual worlds is growing, however we need to distinguish between the anticipated usage and target audience. A closer look at the market research by KZERO (2008) reveals that the largest share is currently in the age group below 18 years. Next to some other famous worlds like Habbot Hotel, World Of Warcraft, or Club Penguin, Second Life had the biggest media impact (covering both its advantages and disadvantages) and was probably the most visible virtual world to the mass market. One of the major advantages and therefore distinctions is the freedom in Second Life, where every user can have an avatar that is capable of becoming a creator of his or her own content, either on their own land or by being part of a community such as educational institutions or companies. This freedom caused the immense growth of virtual worlds but also the discussion of limits and exposure to the so-called darker side of freedom. While content in the Internet is generally even more pervaded with subjects like pornography than the major virtual worlds could ever be, the perception is much higher in a 3D Space because the user receives an immersive feeling of the environment, which in contrast to an Internet site is not restricted in viewing distance. That is, web pages are not visible at the same time – even though they might be in the same paths on the server – whereas 3D Spaces are open and visible. Therefore, the presence of content is pushed on a different level that needs to be considered when starting a virtual course or company. However it is generally comparable to the real-world, where you have an influence on the neighbors as well as the access to your property.

Second Life is still the virtual world of choice when it comes to freedom in designing and implementing content without external tools or restrictions in complexity, structure, or code. Note that some restrictions are still given but these are of minor importance in comparison to other running and accessible worlds. Nevertheless, the authors stay current with the developments in the virtual world market and this paper will have a closer look at worlds like Project Wonderland and Croquet, which are providing interesting technology but are still in the alpha or beta stage. Project Wonderland is described in Section "Monitoring Off-Shore or Outsources Software Development". The 3D Space Croquet is very promising due to the architectural design as a peer-to-peer virtual world similar to concepts from the music and video sharing community, where the traditional server-client architecture is replaced by a decentralized model; see The Croquet Consortium (2009) as well as Becerra-Fernández et al. (2008) for examples with Croquet. Nevertheless, due to the current alpha status of Croquet, we will concentrate on Project Wonderland in our description of life-long learning scenarios.

Virtual Worlds in the Context Software Engineering

In the post punch card era of software engineering, programming has traditionally been conducted in 2D text-based environments (i.e., text editor coding programs like Eclipse). Third-generation coding programs were developed to be more user friendly by utilizing of the graphical user interface (i.e., through use of menus, forms, and diagrams); examples are Visual Basic and Raptor (Carlisle, 2009). More recently the 3D environment available in Virtual Worlds is being used to assist the process of conceptualising, developing, and implementing software programs; see Dreher et al. (2009b). We discuss this topic below with regard to teaching software engineering, monitoring off-shore software development, and on–the-job (vocational) training.

Starting Early to Teach Programming

3D Spaces are well known to students currently in K-12 education due to the popularity of the computer game industry amongst students. Examples include dedicated gaming consoles (e.g., Sony Playstation or Nintendo Wii), PC-based games (e.g., The Sims or World of Warcraft), and social online communities (e.g., Club Penguin or Habbot Hotel), which involve content sharing and multi-player components. Important pedagogical benefits can come from channeling students' motivation and experiences with computer games into education. Projects include, for example, the famous beer game where students operate a brewery to learn about the effects in a supply chain; see Jacobs (2000). For more information about games in education (i.e., serious gaming), see Barab et al. (2004), Michael and Chen (2005) and Serious Games Institute (2009). With respect to software engineering, there are some approaches that combine gaming and 3D worlds for young learners. One example is ALICE (2009), which is a 3D programming environment designed to teach an object oriented programming language like C# or Java by using drag and drop to build a program out of a set of instructions or variables etc. The environment teaches fundamental concepts and structures that are required to implement simple and complex programs; the results are immediately visualized for a deeper understanding of the relationship between source code and its actual effects during execution. Due to the design of the interface, ALICE represents a perfect environment to start software engineering education at a very early age by using the tools for storytelling, gaming, or doing visual presentations. This playful approach to introduce complex subjects at an early stage in students' educational life can be used to ignite their interest for later enrollment in scientific courses or participation in student projects.

Similar to language education it is important to continuously provide education in software development throughout students' lifespan because consistent application facilitates age-appropriate skill acquisition. For programming, a common approach is to use specialized environments like Eclipse and to start teaching programming languages like Java or C#. Nevertheless, the rather technical interface and complex development process can scare students with less experience as they assume the requirement of deep knowledge in various fields like mathematics or sciences. Another approach could be offering programming classes in a 3D Space with appealing scenarios that present tailored opportunities to learn programming. As before, a 3D Space would encourage students to do more than just write source code, it could allow them to become creative in how they realize and represent the solution. This is especially so if the 3D Space provides the same functionality as the environments the students use in their spare time for entertainment. One relevant example is a virtual world developed by Linden Lab called Teen Second Life; see Teen Second Life (2009). It offers the same functionality as Second Life (e.g., creation of objects, programming, collaboration) but is reserved for *safe* usage by being restricted to certain (young) user groups. This is done to prevent bad influences and problems that occurred with Second Life during its hype phase.

Teaching Software Engineering in the Virtual Classroom

The classic software engineering class comprises both theoretical and practical elements. Regarding theory, students have to understand the complexity of software development projects and the methodology to support all steps from the idea to the finished project. In contrast, the practical component includes important capabilities such as communication, social interaction, and team project management. These practical elements are difficult to teach and learn without having the students participating in a larger project. The role of learning through doing is highly valued in a constructivist approach to education (Woolfolk 2006). Therefore, most programming students are required to develop and implement specific software in a team, where they first start with the specifications and finish with a tested application. From the authors' experience with teaching programming, we have observed that the

following factors have a critical influence on student outcomes: previous experience with and knowledge of programming; effective or ineffective communication in project teams and with the client; socializing within the project team; motivation to engage in the process; and helping students to think creatively to identify or generate solutions.

In our experience these critical factors can be beneficially influenced by educational design and through use of 3D Spaces. At Curtin Business School, the current authors have redesigned a third-year information systems project course by using Second Life as an immersive systems development environment set in the context of an existing research commercialization project, an automated essay grading program (Williams & Dreher 2004). Students with prior experience in programming report that it is easy to learn Second Life's programming language (Linden Scripting Language, LSL), while those that are less confident are assisted by menu-based object editors and a comprehensive library of scripts that are continuously being updated by Second Life residents via a wiki. Communication is critical in systems development, both within the project team and with the client. Second Life enhances communication within the team because it has built-in text and voice chat functions (that also interface with the 2D web), and the 3D environment allows all team members to simultaneously work on the project and see results in real time. This latter point also assists communication with the client because it is easy to quickly and iteratively develop demonstration models to check and refine system specifications. In our experience, project outcomes are enhanced when students develop both intrinsic motivations for the project and friendships with their team members. We have observed that students engaging in systems development projects in Second Life are commonly very interested by the flexible, fun possibilities that the 3D Space presents, and they often form positive working alliances and even enduring extra-curricular friendships that started purely online and subsequently moved into the offline world. Regarding the identification of solutions to problems, perhaps it is the combination of intrinsic motivation and social synergy that promotes creative thinking and task-focused problem solving in students working in Second Life; they appear genuinely interested in the in-world applications and can show impressive initiative in identifying creative solutions to system development problems. Naturally, these experiences prepare students for industry experience, and the benefits we have observed can also apply to industry applications such as monitoring off-shore or outsourced projects.

Figure 1 (overpage) shows the west-side of *Australis 4 Learning,* where the Curtin Business School has its virtual world representation in Second Life. The two buildings in the background are the headquarters of the whole project; the building style is used to visualize the grading process of essays, first by the traditional method using human markers (the old style represented by the sandstone building) and by applying the software MarkIT (the new style represented by the modern business building). In the foreground, several platforms are shown that are assigned to student groups within the class. The close distance of the groups and the public development of their solutions is important for the class with respect to motivation. The students receive the same tasks, which are more or less given by properties of the anticipated solution rather than a specific description, and are required to develop everything on *Australis 4 Learning* due to building rights. Therefore, students can see the other groups' outcomes during the development process and are able to adapt their solution, for example by integrating something they learned through observation or to improve their construction by turning in something more advanced than the other group (i.e., motivation by having competition and trying to be the best).

Monitoring Off-Shore or Outsources Software Development

Outsourcing describes the movement of business processes from one company to another (generally involving subcontracting), while off-shoring stands for physical movement of labor to another country (generally the same company); see Friedman (2007). In the majority of cases, the motivation is to save resources, improve efficiency, or to focus on core competencies at the originating companies' locations. Nevertheless, several drawbacks can result, especially considering personnel and project management, communication and team work, including identification with the product, service or business. The following selective list shows some possible problems.

Communication: Being at one location allows regular face-to-face meetings to discuss the progress and expected problems. In face-to-face meetings the communication is immediate and includes emotions, gestures, or accentuation to express more about the status than just the words. Furthermore, the direct contact allows immediate reactions from the conversational partners to identify problems and work on solutions. In the context of off-shoring or outsourcing the (naïve) solution to keep up with the personal contact involves regular traveling to meet at one location, but this involves additional expenditure (time as well as cost).
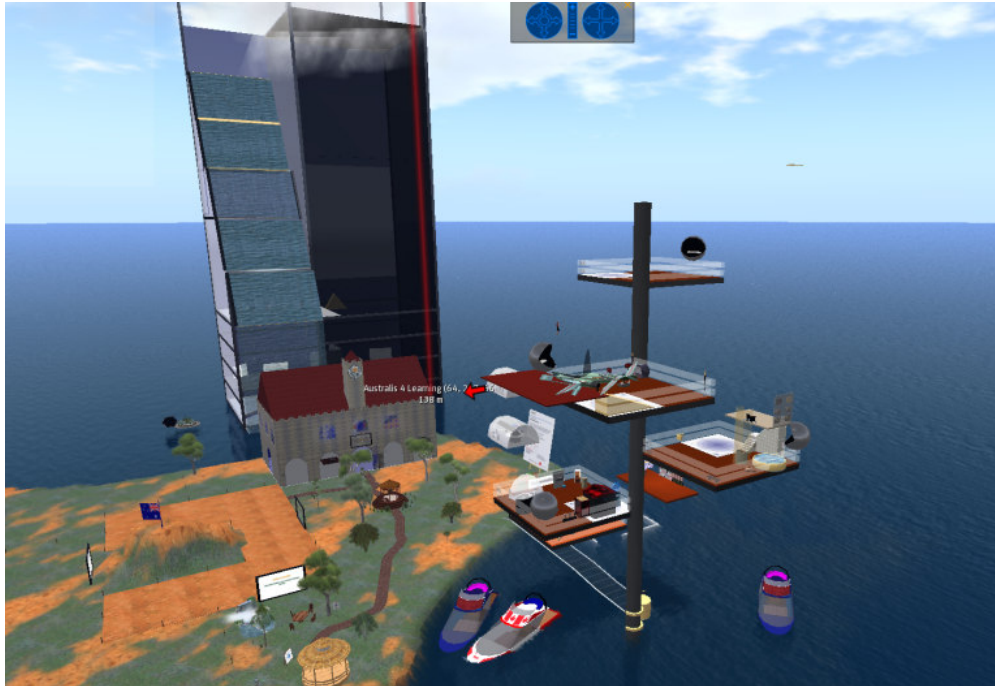
Figure 1: Impression of the Island *Australis 4 Learning* in Second Life showing the students' platforms.

Hierarchy: Distance as well as the lack of personal meetings might have negative effects on the management hierarchy because the responsible manager is probably 'far away' and not able to check the process at any moment. This is also true when it comes to virtual teams that comprise members from various locations and companies. Note that virtual teams from different companies might cause other problems because team members tend to work in favor of their *own* company rather than for the goals anticipated by the team, especially if the goals imply discrepancies for the team member.

Quality Control: The first stages of the software engineering process (specification, concept. Draft) as well as core implementations are typically done by (expensive) in-house experts. Afterwards, mainly to reduce the implementation costs, defined modules are often contracted / out-sourced to other companies or IT groups instead of using the in-house software engineers. The disadvantage is that the quality is related to the module definition as well as the specific programming team, and usually tested at milestones or on delivery. Daily snapshots are technically possible but would require an enormous workload.

3D Spaces provide concepts that solve or at least lower the above-mentioned problems. Similar to classrooms, virtual worlds provide a platform for communication and socializing with far deeper immersion than can be provided by 2D team meeting software or collaboration software. Avatars could be modeled close to the real appearance such that the recognition of team members is possible by audio and visual characteristics. The meeting in a virtual space allows synchronous communication and direct reactions as is possible in face-to-face meetings. In addition, 3D Spaces can be used to visualize information (e.g., display business graphs, 3D objects or processes) by using the building capabilities of the virtual world.

Another advantage of 3D Spaces is the 'physical' context of the meeting. While video or telephone conferences are relatively anonymous with respect to the company, 3D Spaces allow a unique location with representative furniture and room accessories that transport the team affiliation. This concept can be transferred to virtual offices in the same building as the meeting room where the team members have their own office following the design of a corporate identity. Similar to the real world, where individuals become part of the company's philosophy by working in the offices, employees can have an immersive experience in the virtual world and might identify themselves with the team rather than their originating company.

Regarding an anticipated scenario, a company needs some applications and decides to have the software developed in a country with lower costs for programmers. In this case one problem might be that the software has bugs or does not follow certain quality guidelines with respect to design, coding, performance or stability. As a result, the developer and the programmer need to communicate regarding the required changes. This might repeat in

several cycles until the software is finished and fulfills all specifications. Further costs for face-to-face meetings or post-processing might occur.

3D Spaces can be used to have regular meetings to verify the process as well as quality and allow a prompt intervention in case of problems. Nevertheless, there are specific 3D Spaces that have advantages over others as they provide more advanced technologies in addition to the general communication channels. One upcoming virtual world with a focus on such capabilities is Project Wonderland (Fig. 2), which is supported by Sun Microsystems. The Client-Server architecture is an Java-based open-source environment that provides a closed 3D Space similar to Second Life. It includes high-fidelity, immersive audio, object interaction and, most important, live applications such as web browsers, word processing or programming environments. In particular the integration of x-term based applications is an innovative form of collaborative work, which cannot be achieved in such a way by Remote Desktop from Microsoft for example. To illustrate the advantages, the following example describes a short project that is currently implemented as a prototype together with a software development company.
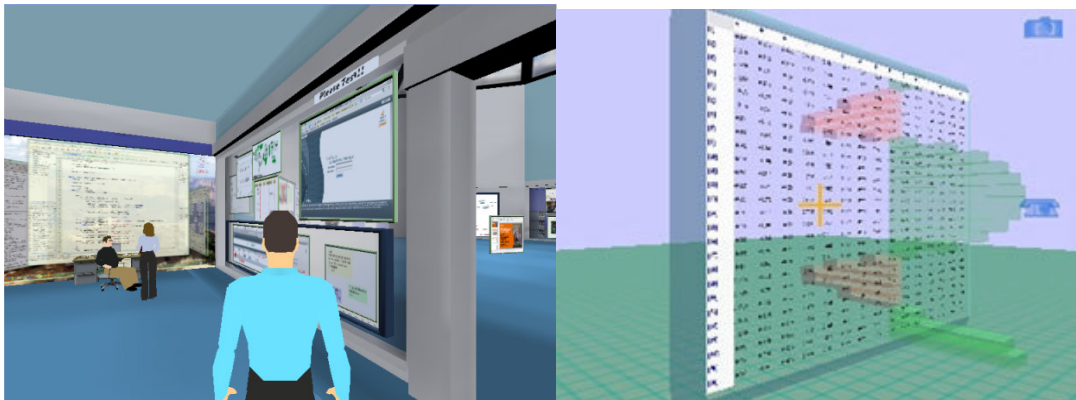
Figure 2: Project LG3D screenshots (MD 2007).

The project management is located in Germany where most of the system architecture is specified and implemented. Non-core components are produced off-shore at a partner in an East-European country. The current development process involves several face-to-face meetings where milestones are discussed, further coordination is done by video conferences and remarks in the source code. Nevertheless, collaborative work on the software is not possible. Currently a project room is created similar to the example in Figure 1. Besides communication components (e.g., phone, chat, browser) some advanced applications provide collaboration (i.e., Whiteboard, OpenOffice and Eclipse). The whiteboard is similar to the web-based equivalent where everyone in the room can draw simultaneously on it. OpenOffice and Eclipse are used to show the documentation as well as the software development environment. The advantage is that all attendees can gather in the virtual room and 'observe' the programmer and his or her work in progress. Mistakes or code that are not in compliance with the specification can be immediately determined and, if necessary, modified by taking control of the application (clicking in the window). The same applies to OpenOffice where a group can share documents, such as the task list with the deadlines. Note that the common tools are still provided (conferencing, document exchange, chat) and the shared application is an extra provided in the 3D Space.

Training on the Job

The scenarios in last two sections introduced technologies that support the learning process, especially in cases where the learners are distributed over many locations. Consider the following situation: A company is migrating to a different software system, which requires employees to participate in special training units. During this time the employees are not in their offices but in classrooms, which are generally located further away and require traveling and overnight stays. With 3D Spaces, we can define new learning scenarios where the class meets online in a virtual representation of the classroom. The employee is able to either stay in the office or even at home while being in a virtual classroom that is equipped with tables and chairs for the participants as well as having the required applications placed on their virtual desks or the virtual walls. Additional equipment that is a familiar part of real classrooms (e.g., flip charts) is accessible, as are tools for group or individual communication. The tutor can do live demonstrations on the software, which is seen by everyone in the room. Furthermore, participants can ask

questions in different ways: voice, chat, pointing at the application or even by taking the control of the application to demonstrate the problem. The real advantage is that training units can easily be setup. Each participant gets his own wall with the application where a certain exercise is to be solved, such as defining a format in the office application or implementing a class in Eclipse. If a participant has a question, the tutor can come over and share the application by taking control (click in the window) and assist by writing the first lines of the code for example. Due to the immersive audio system, they can even use voice talk, which is programmed to mimic real vocal communication and is less audible as the distance increases, thereby not disturbing other participants.

3D Spaces, Constructivism, and Immersion

If a learning theory is applied to 3D Spaces, it is generally constructionism; see Dede (1995), Riedl et al. (2005), or Coffman and Klinger (2008). The constructionist perspective suggests that learning can happen most effectively when students are actively creating things. This is particularly relevant to virtual worlds because they provide a new freedom for students to perform tasks in 3D Spaces that would not be possible otherwise, for example experimenting with architecture, design sculptures or visit historical places. Nevertheless, other theoretical perspectives such as behaviorism are also relevant to 3D Spaces which are a great environment to observe the behavior of others in various situations. For instance, class students can observe the lecturer performing certain tasks which are then repeated immediately. Regarding the application sharing in Project Wonderland, a scenario for a training unit in Java Programming could be designed in the following way: The lecturer opens the Eclipse programming environment in Project Wonderland as a shared application and demonstrates the software development process. Just as is possible on a desktop application, the students can follow each step - especially the steps that are not directly related to typing the source code, such as the arrangement of tools, setting of the environment, or tricks to improve the editing process (e.g. macro usage or application of wizards to create code segments). Observing the lecturer this way in the 3D Space presents advantages over similar demonstrations in physical classrooms (visibility, communication of the avatars without interfering with the lecturer, recording of the demonstration by taking a video recording). For the lecturer, the same technology provides an opportunity to monitor and correct students' work. After the demonstration by the lecturer, every student can open the shared application Eclipse at any virtual wall to practice program development. The lecturer can follow the progress and provide feedback as soon as mistakes become obvious.

Immersion is best described as the experience of becoming part of the 3D environment where the virtuality starts to feel real; see Educause Learning Initiative (2006), Kemp and Haycock (2008), and Clarke et al. (2008). However, the immersive elements are often limited to interactions of the avatar with the 3D environment rather than integrating the real user or real-world; e.g. actions in the virtual world have effects on the real world (or vice versa) or it does not matter if something is either done in the virtuality or the reality. Therefore, we started several projects at the University of Hamburg and Curtin Business School to achieve an even deeper integration of the student into the 3D Space. The first project is in the context of automated essay grading (Dreher et al. 2008), where the submission, grading and analysis can be done in both virtual and real worlds. Note that SLoodle (Kemp 2007) has similar mechanism to turn homework in but does not provide automated marking or analysis. Furthermore, in the Curtin Business School automated essay grading project the 3D Space is used to visualize and simulate the process of the automated assessment from different perspectives.

A project at the University of Hamburg involves students experimenting with hi-tech equipment that is also low-cost (see Lee 2008). Even in primary and secondary school, SmartBoards are used to improve the interaction, and therewith, the immersion of the students into the environment. As shown in Figure 3, the user can interact directly with the objects by using their fingers to drag, rotate or insert annotations (in these examples the user can rotate a chemical molecule or walk through the streets of Paris). The same interactivity is possible with Second Life and other 3D Spaces and this is the first step of merging both worlds (i.e., physical and virtual). Using 3D glasses can extend the impression that both worlds merge because the 3-dimensional view provides a seamless transition from the physical world into the virtual world. Furthermore the University of Hamburg students are working on tools to overcome certain restrictions of the SmartBoard, such as where only one finger can be detected at any moment (i.e., there is no multi-touch interface like there is with the Apple IPhone). Based on Lee's (2008) ideas and using the Wii-Remote control (that allows any wall to become an interactive display), students extended the 2-dimensional interpretation of the input device by adding another remote and tracking the distance from the wall. Even though the resolution is not comparable with professional equipment, it was demonstrated that the immersive feeling can be increased dramatically by investing less than USD$50.
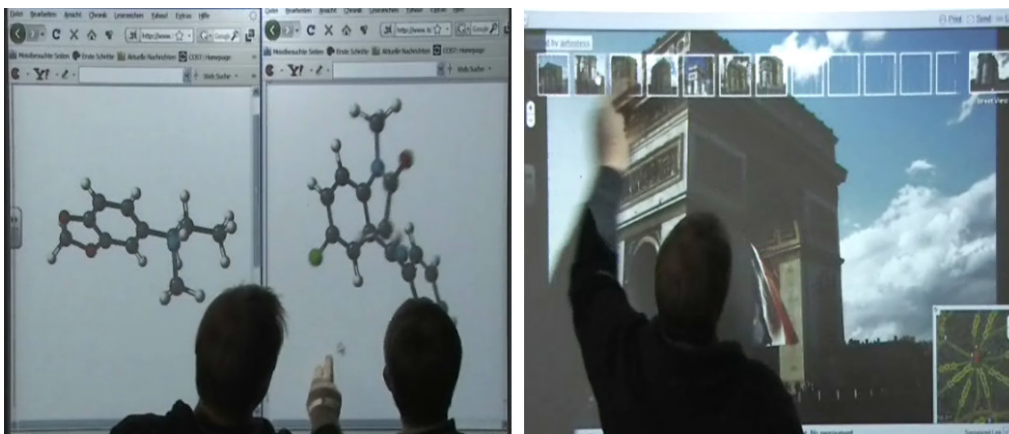
Figure 3: Demonstrating interaction on a SmartBoard: showing molecules in 3D (left) and allowing students to have a look at landmarks in different European cities (right)

Another form of immersion occurs when avatars interact with the real world. All 3D Spaces provide many channels for communication within the 3D Space but also to other virtual worlds and networked applications and devices. For instance avatars are able to call – comparable to Skype by voice or instant messages – real world people to inform them about events or other requested information. In addition to interfaces that display information about the current online status in a given 3D Space, we are working on several tools to enable advanced tracking and observation of the 3D Space to allow visualization and interaction in the real world; e.g. creating heat maps of the islands that show the paths of avatars, logs of changes by students, inter-world communication. We predict that in the years to come the boundaries between the real and virtual worlds will blur; for businesses in particular, being able to contact individuals via multiple media (both real and virtual) will become increasingly valuable.

Conclusion

This paper has demonstrated three scenarios in the context of information systems and software development, these being education, virtual projects and vocational training. A common element in each of these contexts is that the protagonists are distributed all over the world and yet still share the same experience at the same time via participation in 3D Spaces. Virtual worlds have been around for quite some time but have always had a difficult standing, especially compared to the Internet and the Web 2.0. In the beginning the fast growth produced some nice treasures which got hidden under the huge amount of content that flooded all virtual worlds. However a time of consolidation has started and we expect innovative approaches to appear over the next few years. This innovation might be achieved purely in-world by using all possibilities of the 3D Space rather than simply transferring the 2D content and Web 2.0 functionality into the virtual environment. Additionally, innovation could proceed by breaking the invisible wall between physical and virtual realities and making 3D Spaces part of the real world. This would enable people to interact seamlessly with each other and with virtual objects as they traverse the physical world, or teleport to specific virtual locations that are better suited for solving real-world problems. Similarly if we need to develop or test 3D objects or investigate complex machines, we can open a window to a virtual world, drag the required objects from our desktop into the 3D Space and interact with them alone or in a collaborative way. Accordingly 3D Spaces can powerfully augment our capacity to interact and/or achieve our goals in less time than might be required otherwise.

New technology needs time, but a survey of managers in Internet companies by Novomind revealed that the subject of virtual worlds was better known than even Skype; even though 50% of respondents classified virtual worlds as overrated hype, they were aware of the advantages. 3D Spaces are used world-wide within impressive projects, but they are still in their infancy. We have presented viable scenarios that highlight the utility of the 3rd dimension online. However users of virtual worlds commonly limit themselves by using only the basic functions. In contrast there are also many researchers, early adaptors and adventurers that are engaging in pioneering exploration of virtual worlds and have discovered huge potential in 3D Spaces for education, virtual teams, and vocational training in software engineering.

Furthermore, software engineering is not a scenario commonly associated with 3D Spaces. This paper demonstrated that an additional dimension online can support the user at all stages of the software development

process. The configuration of the 3D Space according to the needed requirements can be done in most cases with only a few clicks and facilitates fast integration and maximum of flexibility. By using a game-like environment, a virtual world can provide powerful support and motivation for young students to start learning programming. Virtual worlds also allow experts to work collaboratively by providing a common view of the development tools that is independent of physical distance. Finally, 3D Spaces facilitate vocational training because the employee need only log on to their virtual world avatar in order to participate in a vocational training course, and the virtual experience still facilitates socializing and sight-seeing opportunities that are valuable aspects of a real-world business trip. In closing it seems both natural and beneficial for software engineers and programming students to be studying, working, and training in a virtual environment - the digital nature of virtual worlds is so very intrinsic to the vocation of software engineering, and the innovative nature of 3D Spaces adds value for individuals, organizations, and indeed the industry.

References

ALICE. (2009). *An Educational Software that teaches students computer programming in a 3D environment*. Retrieved March 30, 2009, from http://www.alice.org

Barab, S., Thomas, M., Dodge, T., Carteaux, R., & Tuzun, H. (2004). Making learning fun: Quest Atlantis, a game without guns. *Educational Technology Research & Development, 53*(1), 86-108.

Becerra-Fernández, I., Madey, G., Prietula, M., Rodríguez, D., Valerdi, R., & Wright, T. (2008). Design and development of a virtual emergency pperations center for disaster management research, training, and discovery. *Proceedings of the 41st Hawaii International Conference on System Sciences*, 1-8.

Carlisle, M. (2009). *Welcome to the RAPTOR home page*. Retrieved April 5, 2009, from http://www.usafa.af.mil/df/dfcs/bios/mcc_html/raptor.cfm

Clarke, J., Dede, C., & Dieterle, E. (2008). Emerging technologies for collaborative, mediated, immersive learning. In J. Voogt & G. Knezek (Eds.), *International Handbook of Information Technology in Primary and Secondary Education* (pp. 901-909). New York: Springer.

Coffman, T., & Klinger, M. B. (2008). Utilizing virtual worlds in education: The implications for practice. *International Journal of Social Sciences, 2*(1), 29-33.

Conklin, M. S. (2007). *101 uses for Second Life in the college classroom*. Retrieved March 30, 2009, from http://warburton.typepad.com/disruptive/documents/SL_handout.pdf

Dede, C. (1995). The evolution of constructivist learning environments: Immersion in distributed, virtual worlds. *Educational Technology, 35*(5), 46-52.

Delwiche, A. (2006). Massively multiplayer online games (MMOs) in the new media classroom. *Educational Technology & Society, 9*(3), 160-172.

Dickey, M. D. (2005). Three-dimensional virtual worlds and distance learning: two case studies of Active Worlds as a medium for distance education. *British Journal of Educational Technology, 36*(3).

Dreher, C., Reiners, T., Dreher, N., & Dreher, H. (2009a). 3D virtual worlds as collaborative communities enriching human endeavours: Innovative applications in e-learning, *Accepted for publication in the proceedings for the IEEE-DEST 09 conference*. Istanbul, Turkey.

Dreher, C., Reiners, T., Dreher, N., & Dreher, H. (2009b). 3D virtual worlds enriching innovation and collaboration in Information Systems research, development, and commercialization, *Accepted for publication in the proceedings for the IEEE-DEST 09 conference*. Istanbul, Turkey.

Dreher, N., Dreher, H., & Reiners, T. (2008). Design and integration of an Automated Assessment Laboratory: Experiences and guide. *Proceedings of ED-MEDIA 2008: World Conference on Educational Mul- timedia, Hypermedia & Telecommunications, Vienna, Austria*, 2858–2863.

Educause Learning Initiative. (2006). *Seven things you should know about virtual worlds*. Retrieved March 30, 2009, from http://www.educause.edu/eli

Erlenkötter, A., Kühnlenz, C.-M., Miu, H.-R., Sommer, F., & Reiners, T. (2008). Enhancing the class curriculum with virtual world use cases for production and logistics. *Proceedings of E-Learn 2008: World Conference on E-Learning in Corporate, Government, Healthcare, & Higher Education, Las Vegas*, 789–798.

Friedman, T. L. (2007). The world is flat 3.0: A brief history of the Twenty-first Century. New York: Picador.

Gregory, S., Reiners, T., & Tynan, B. (2009). Alternative realities: Immersive learning for and with students [working paper]: University of Hamburg, Germany, and University of New England, Australia.

Hayes, G. (2009). *The Virtual Worlds Hype Cycle 2009: Business, education, marketing, social media, virtual worlds*. Retrieved March 30, 2009, from http://www.muvedesign.com/the-virtual-worlds-hype-cycle-for-2009

Jacobs, F. R. (2000). Playing the beer distribution game over the internet. *Production and Operations Management, 9*, 31-39.

Kemp, J. W. (2007). *Prim Drop Alpha: YouTube video demonstrating a drop box*. Retrieved March 30, 2009, from http://www.youtube.com/watch?v=VUUnnLm-AG4

Kemp, J. W., & Haycock, K. (2008). Immersive learning environments in parallel universes: Learning through Second Life. *School Libraries Worldwide, 14*(2), 89-97.

KZERO. (2008). *Research*. Retrieved March 30, 2009, from http://www.kzero.co.uk/blog/?page_id=2092

Lamont, I. (2007). *Virtual reality and higher education: Another perspective*. Retrieved March 30, 2009, from http://terranova.blogs.com/terra_nova/2007/05/teaching_in_vr_html

Lee, J. (2008). *Creating tech marvels out of a $40 Wii Remote*. Retrieved March 30, 2009, from http://www.ted.com/index.php/talks/johnny_lee_demos_wii_remote_hacks.html

MD, D. (2007). *Do they have Croquets in Wonderland?* Retrieved April 5, 2009, from http://www.javalobby.org/java/forums/t101778.html

Michael, D., & Chen, S. (2005). *Serious games: Games that educate, train, and inform*. Florence, KY: Cengage Learning Services.

Prensky, M. (2001). *Digital natives, digital immigrants*. Retrieved March 30, 2009, from http://www.marcprensky.com/writing/Prensky%20-%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf

Riedl, R., Bronack, S., & Tashner, J. (2005). Innovation in learning assumptions about teaching in a 3-D virtual world, *International College Teaching Methods and Styles Conference*. Reno, NV.

Serious Games Institute. (2009). *Serious Games Insitute home*. Retrieved March 30, 2009, from http://www.seriousgamesinstitute.co.uk

T-Systems. (2009). *Trends EduKey*. Best Practice 1/09. Retrieved March 30, 2009, from http://download-tua.sczm.t-systems.de/ContentPool/en/StaticPage/59/76/94/597694_Complete-Edition-ps.pdf?client=t-systems.de

Teen Second Life. (2009). *Official site of the free 3D online virtual world for teens*. Retrieved March 30, 2009, from http://teen.secondlife.com

The Croquet Consortium. (2009). *Croquet Consortium main page*. Retrieved March 30, 2009, from http://www.opencroquet.org/index.php/Main_Page

The Equity Kicker. (2007). *Second Life is riding the Hype Curve*. Retrieved March 30, 2009, from http://www.theequitykicker.com/2007/07/25/secondlife-is-riding-the-hype-curve

Williams, R., & Dreher, H. (2004). Automatically grading essays with Markit©. *Journal of Issues in Informing Science and Information Technology, 1*, 693-700.

Woolfolk, A. E. (2006). *Educational Psychology* (10th ed.). Sydney, NSW: Pearson.