OpenBioSafetyLab: A Virtual World based Biosafety Training Application for Medical Students

Arturo Nakasone*, Siyu Tang†, Mika Shigematsu‡, Berthold Heinecke*, Shuji Fujimoto§ and Helmut Prendinger*

*National Institute of Informatics, 2-1-2 Hitotsubashi,Chiyoda Ku, Tokyo 101-8430, Japan

arturonakasone@nii.ac.jp, berthold@heinecke.me, helmut@nii.ac.jp

†RWTH Aachen University, Ahornstrasse 55, 52074 Aachen, Germany, siyu.tang@rwth-aachen.de

‡National Institute of Infectious Diseases, 1-23-1 Toyama,Shinjuku Ku, Tokyo 162-8640, Japan, mikas@nih.go.jp

§Faculty of Medical Sciences, Kyushu University, 3-1-1 Maidashi,Higashi Ku, Fukuoka 812-8582, Japan

fujimoto@shs.kyushu-u.ac.jp

Abstract—Recently, virtual world technology has been successfully used to create interesting and useful applications in the context of scientific research, ranging from visualization of static 3D representations of molecular entities to full execution of astrophysical simulations. Unfortunately, the majority of these applications are somewhat limited by the current functionality of the virtual world client software, constraining them to provide only the most basic features of user interfaces. Thus, the implementation of applications such as training programs becomes quite challenging due to the high level of interaction a user must have with the training environment itself. In order to analyze the level of usability for virtual world based applications, we will introduce OpenBioSafetyLab, our implementation of a virtual world based training application in bio-risk management, which is one of the most important areas for training systems nowadays. We also present the results of our exploratory test study with twenty-four subjects, which indicates a high degree of usability of our system, not only for the testing aspect of the training, but also for the learning aspect.

Index Terms—Biosafety, Virtual Worlds, Training Application, Virtual World Interface, User Study

I. INTRODUCTION

Despite the fact that virtual world technologies have been conceived mainly as a tool for social interaction, they are becoming an increasingly important medium for the development of powerful collaborative research applications in the context of e-Science [1]. Through the manipulation of avatars in the virtual world, users have to their disposition unique and interesting ways to interact with each other, as well as with their environment [2], [3]. For instance, Second Life [1] became a sound platform to deliver these kinds of applications mainly due to its accessibility and popularity in terms of user base [4]. Now, the need to create richer applications in a context in which information storage privacy is of utmost importance has encouraged the development of free, open source alternatives. These alternatives are allowing researchers to fully utilize and extend the capabilities of virtual world technology to deliver comparable results to more traditional, standalone applications in terms of performance and, more importantly, usability. One of the most popular open source platforms is Open Simulator (OpenSim) [2], a Second Life

compatible server application that allows developers to extend its functionality through customized loadable modules.

Early attempts to develop scientific research applications using virtual worlds dealt mostly with static 3D representations of the studied entities, such as fractals [5] or molecules [6], which were generated in Second Life. With the availability of OpenSim, complex applications that added dynamics to these representations became possible, supporting the execution of full simulation playbacks and interaction [7], [8]. However, a great majority of these applications gave a stronger emphasis to the algorithmic and data aspects of their corresponding research processes, leaving them with over-simplified user interfaces. This was due to the fact that virtual world client programs (the software that is used to navigate inside virtual worlds) lack the richness of interactivity of standalone applications, or even web-based systems. Complex applications such as training scenario managers, in which users need to be able to engage in more elaborated interaction with the environment, are difficult to be developed mainly for this reason, and require a high degree of creativity to overcome the limitations imposed by the virtual world client software itself.

One of the topics for training scenarios that has received a high degree of attention in recent years is the handling and management of pathogens as part of basic infectious disease education, a requirement for strengthening the response capability to bio-terror crises. It has been pointed out that specialized education of bio-risk training is necessary for handling infectious specimens in clinical practice and for researchers who conduct studies on these specimens. Internationally, the establishment of education that fosters the ability to properly manage bio-risk is required for experimenters to continue research activities in real-world environments. However, real-world clinical practice for each student is unrealistic, because bio-terror crises may be caused by experiments, which provokes high risk and expense. Thus, scenarios such as these can greatly benefit from the utilization of virtual world technology as a risk-free, cost-effective training environment.

In this paper, we present a virtual world based training application, collaboratively developed with the National Institute for Infectious Diseases, Japan. This application emulates an scenario in which users, through the manipulation of

---

[1]Second Life - http://www.secondlife.com

[2]OpenSim - http://www.opensimulator.org

IEEE computer society

avatars, are expected to properly handle critical biosafety related occurrences in a simulated lab environment. By having performed a preliminary test study with medical students, we will report our conclusions regarding the general usability of virtual world client programs, and make suggestions on how to improve usability for non computer scientists.

The paper is organized as follows. Section 2 describes some early and recent attempts to develop training-like applications in 3D environments. Section 3 explains the architecture of our OpenBioSafetyLab, including the generation of scenarios and the processing of user input. Section 4 describes our experimental setting and the results we obtained with medical students from Kyushu University, Japan. Section 5 presents a discussion about the usability issues we identified during the implementation and testing of the application and how they could be overcome. Finally, a conclusions and future work section will round up the paper.

II. RELATED WORK

Researchers in the field of computer graphics have always encouraged the use of 3D environments to develop training (a.k.a. learning) applications. Some of them show the potential of the environment itself to react to the multiple ways users can interact with it [9], while others concentrate on the learning aspect, ranging from the observation and manipulation of simple physical phenomena [10] to more complex scenarios involving numerous procedures [11].

Although latest advancements in medical simulation visualization suggest that there is a continuing search for high realism in the replication of the real world [12], [13], these systems are typically very expensive to build and, in most cases, sacrifice availability for accuracy. Therefore, cheaper alternatives have been evaluated. Marks et al. [14] have analyzed the possibility of using game engines as the environment to implement medical applications and, even though current game engine technology cannot offer the level of realism observed in the previously described applications, it offers interesting and cost-effective alternatives to produce fast and fairly accurate implementations. For instance, Cavazza and Simo [15] developed an application that simulates a virtual patient using the Unreal Tournament game engine.[3]

The wide range of features offered by current game engines allow medical application developers to easily create accurate scenarios based on real situations. However, most game engines are constrained by the settings defined for the specific game it was build, e.g. first person shooter games are provided with entities such as weapons and monsters and scenario building blocks more appropriate for military or space installations. In order to overcome these limitations, researchers have often made use of customized (and often costly) versions. In addition, to interact with the application, each user has to purchase a license of the game itself , which might prove impractical in most educational settings. Thus, we decided to investigate the applicability of open
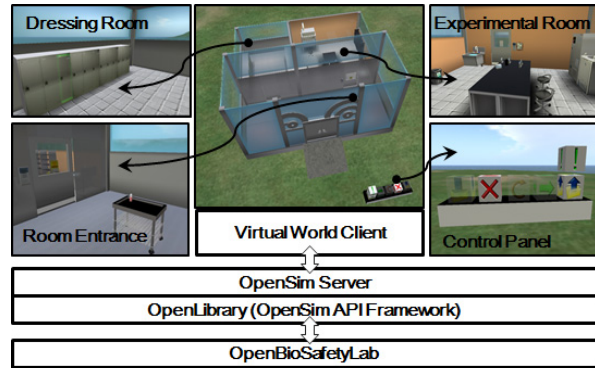

Fig. 1. OpenBioSafetyLab Application Architecture and Main Visual Interface Elements

source alternatives such as virtual world technology in the implementation of medical training systems.

III. OPENBIOSAFETYLAB – DESIGN AND IMPLEMENTATION

In this section, we will describe the main functional attributes of OpenBioSafetyLab: (a) the architecture of the application, (b) the visual setting constructed for our sample biosafety scenario, (c) the mechanism for user and system event detection and selection, and (d) the scenario format that users can utilize to create their own scenarios.

A. Application Architecture

Figure 1 shows the main architecture for our biosafety training system, along with its main visual interface elements. OpenBioSafetyLab, a Microsoft .NET based application, is the main component of the solution and is in charge of generating the virtual infrastructure in OpenSim, along with managing the scenario events, i.e. application states triggered by specific actions performed by the user. In order to let OpenBioSafetyLab control OpenSim resources and functionality in an easy way, we developed an OpenSim based API framework called OpenLibrary [4]. The main advantage for the use of such a framework lies in its simplicity, allowing application developers to manipulate virtual world resources through a very clean interface. To interact with this application, users just need to connect to the OpenSim server through either a Second Life or Hippo Opensim Viewer [5] software.

B. Visual Environment Setting

The main visual environment in OpenBioSafetyLab is divided into four interaction areas, as shown in Figure 1. The *Control Panel*, located outside the laboratory's closed area is used to initialize or cancel the training session. The *Room Entrance* area is the first area the user finds when he/she enters the laboratory and serves as the connection to the other two main sections: the *Locker Room* (an area used to pick up lab

---

[3]Unreal Tournament - http://www.unrealtechnology.com

[4]OpenLibrary Project - http://www.globallabproject.net
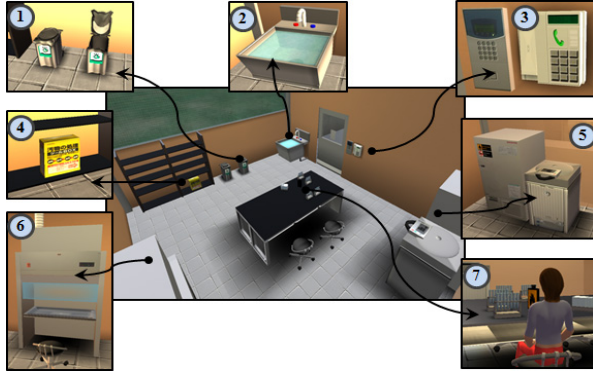[5]Hippo OpenSim Viewer - http://mjm-labs.com/viewer

Fig. 2. OpenBioSafetyLab - Laboratory Appliances and Objects: (1) Waste bin, (2) Wash bin, (3) Intercom and Phone, (4) Fluid Spill Kit, (5) Fridge and Autoclave, (6) Biological Safety Cabinet, and (7) Experimental Table

Fig. 3. Diagram of a simple emergency situation in a biosafety laboratory.

clothes) and the *Experimental Room* (the area in which the main training process occurs).

The Experimental Room was modeled from a typical biosafety laboratory with a Level 2 safety risk [16]. In this room, we implemented a set of typical artifacts that are commonly found in this kind of environment (Figure 2).

C. The Event Bucket Mechanism – Supporting Dynamic Instantiation of Events

In this section, we will describe the core algorithm of our training system, which we called the *Event Bucket Mechanism*, through a sample scenario describing an emergency procedure that is most likely to happen in this kind of laboratory (see Figure 3). In this scenario, the user, through its virtual world representation (i.e. avatar), starts the experiment by pushing a cart object into the room. This cart carries a bottle containing a toxic liquid (Figure 3-1). As soon as the OpenBioSafetyLab application detects that this action has been performed by the user, it triggers an action that results in the crashing of the bottle onto the floor and the spilling of its content (Figure 3-2). Once this action is performed, the system waits for a user reaction by preselecting which events are to be triggered from its event database or "bucket". For instance, among the possible procedures to be performed by the user, he/she can decide to: (a) put a fluid protector over the spill (Figure 3-4), (b) wash his/her hands, or (c) call the security office (Figure 3-6). If the user decides to put the fluid protector over the spill, then the application performs a new preselection of events, which in this case determines that the user can decide to put the used fluid protector in the waste bin once the spill has been cleaned up (Figure 3-5). One of the key aspects of OpenBioSafetyLab is that it does not force the user to perform a specific action, but just takes into account what the user does. He/she could even decide not to perform any action at all and finalize the training session at any moment. The rationale here is that OpenBioSafetyLab was designed as an application for testing biosafety knowledge, and not as a learning application.

An action performed by the user, which might be relevant for the execution of the training scenario or its respective
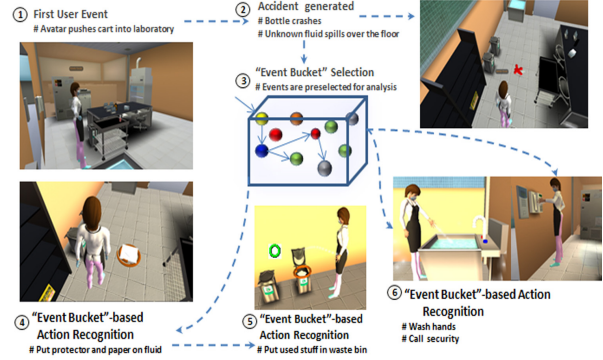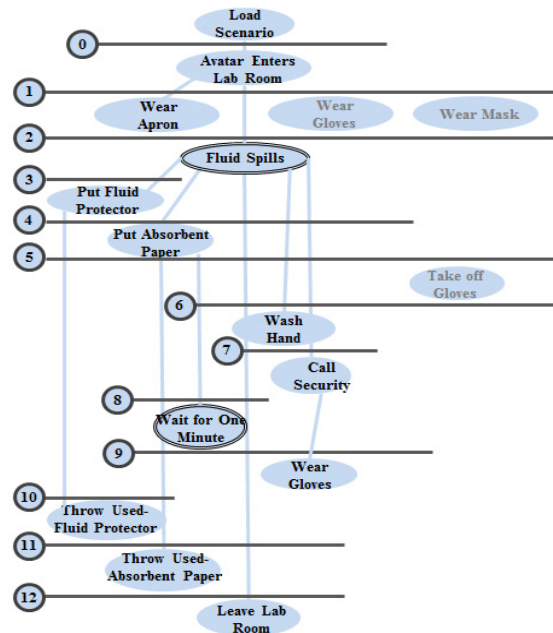

Fig. 4. Event Bucket Abstraction for a Toxic Liquid Spill Emergency Scenario (Biosafety Management Perspective). The events displayed with a double-lined border are system events, while the others are user events.

reaction, is known as an "event". Thus, all the events that the application handles for a specific training scenario are organized in a set of loosely coupled objects that we called an "event bucket". In addition, events inside the event bucket are categorized by levels, which define the requirements for a specific event in terms of preconditions. For instance, if the user tries to execute an action which is part of an event located at level N, the system verifies that the events located in previous levels (0 to $N-1$), which were defined as prerequisites of the executing event, were also executed. If any prerequiste was not executed previously, the user will still be able to execute the action, but this will not be registered as correctly executed inside OpenBioSafetyLab.

Figure 4 represents the event structure for the previously described scenario. Each level in the diagram indicates one
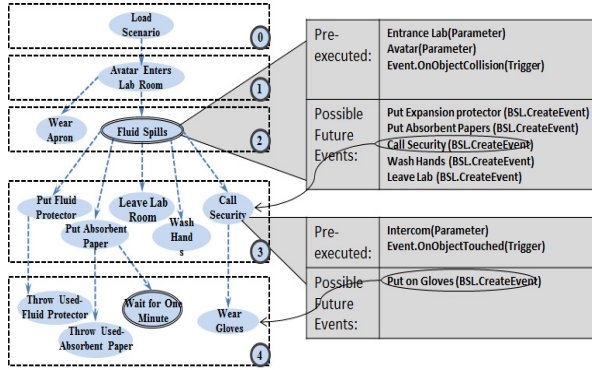
Fig. 5. Event Bucket Abstraction for a Toxic Liquid Spill Emergency Scenario (OpenBioSafetyLab System Perspective)

state in the training scenario and each event belongs to only one state. The sequence of states defined for this scenario corresponds to the sequence of actions in a standard procedure that conforms to the basic knowledge of biosafety management. Typically, there are two types of events, depending on how the event is triggered: (a) user events, if the user performs an action that is recognized as an event, and (b) system events, if the application itself performs the action.

Due to the very nature of virtual world technology, the possibilities of interaction that users may have with the application are limitless and hard to control. This would translate into high numbers of sequences of actions that are difficult to predefine in the system. For instance, after the liquid spills on the floor, a user could either try to clean the liquid with the fluid protector or wash his/her hands first or call the security office or even leave the room in panic. For this reason, the application needs to be able to intercept any potential action from the user, no matter whether they are in the correct procedural order from a biosafety perspective.

In order to simplify the creation and processing of scenario events, we redefined the concept of level to specify not only a state in the training process, but also a container of events that will be processed dynamically as the user performs actions that satisfy the preconditions for previous levels. This allows us to reduce the number of levels for a given scenario to a more manageable set. Figure 5 depicts a representation of the same training scenario described above, but from the perspective of the application.

In the case of a system event processing (i.e *Fluid Spills*), we defined a pre-executed condition that must be fulfilled to trigger this event, namely: the "Avatar" object must perform a "Collision" with the "EntranceLab" object (i.e.the avatar must go inside the experimental room). Once this action has been completed, the application will instantiate the events that are in the next level for monitoring (*Put Fluid Protector*, *Put Absorbent Paper*, *Leave Lab Room*, *Call Security*, and *Wash Hands*). In the case of a user event processing (i.e *Call Security*), the only difference is that the application will perform no action when the pre-executed condition is fulfilled

(i.e. the user touches the intercom object).

In addition to the level based event processing, the application implements a feature to indicate the user if an action has been executed in the correct sequence or if it has been executed too early in the process. This is useful to provide the user a visual feedback about his/her progress during the testing session.

IV. User Interface Evaluation

In order to evaluate the viability of our application in terms of usability, we conducted a preliminary test study with twenty-four students from the Faculty of Medical Sciences at Kyushu University, Japan. The students were aged between 21 and 29 years old, and most of them had had certain degree of experience using virtual world technology. They also had theoretical experience in handling infectious substances, but only half of them had experience on how to behave inside a biosafety lab.

A. Case Studies and Experiment Setting

In the study, we wanted (1) to find evidence on the viability of virtual world technologies to address the issue of realism for training applications, and (2) to investigate whether the use of the feedback option through the testing process would be perceived as useful by the students. Therefore, we considered two versions:

- *Feedback Version*: In this version, users are provided with the feedback value during the whole training process.
- *No Feedback Version*: Here, users never receive any kind of feedback from the system.

To test both versions, we set up two client computers in Kyushu University, which connected to corresponding OpenBioSafetyLab application servers located at the National Institute of Informatics, Tokyo. In one server, we installed the Feedback version of the application and, in the other, we installed the No Feedback version. Students were not aware of which version they were testing during the experiment, and they were randomly assigned to either version of the system.

Before performing the actual training session, users had the opportunity to get acquainted with the manipulation of virtual world objects with his/her avatar through the performance of a simple task in which he/she just needs to go inside the dressing room, take one pipette-like object, and take it to the experimental room. By performing these simple actions, we wanted the user to become comfortable with the environment.

B. Questionnaire Results

To collect the feedback from the students about their experience, we designed a questionnaire, which contained five sections: *Preconditions* (general personal information), *Acting in the Virtual Environment* (questions regarding the virtual world experience itself), *System Feedback* (questions regarding the system feedback), *Reality Experience* (questions regarding the reality sensation of the whole training process), and *User Feedback* (questions regarding the learning and entertainment experience of the application). Students from both versions

Scheme Number	Value Range
1	Very Well (1) to Not at all (5)
2	Very Easy (1) to Very Difficult (5)
3	Very Much (1) to Not at all (5)
4	Yes (1) or No (2)
5	Yes (1), Not Sure (2) or No (3)

Question	Answer Scheme	Mean	σ^2
6	1	2.83	0.92
7	2	3.35	0.65
8	1	3.46	0.78
9	3	2.58	0.79
10	3	3.67	0.89
11	4	1.17	0.39
12	3	3.38	0.82
13	3	3.21	1.02
14	3	3.42	0.93

had the same questions for all the sections, except the System Feedback. In the Feedback version, students were asked if the feedback was either helpful or disturbing, while in the No Feedback version, they were asked if feedback from the system would have been desirable. Table I shows the questions for both versions. The answer choice schemes are presented in Table II. Both sets of items were directly translated from the original Japanese format.

For all the questions, we calculated the mean and standard deviation values for the complete sample of twenty four students. These values are presented in Table III.

From the results obtained for the introductory part of the questionnaire (Questions 6 and 7), we notice that students managed to understand the whole training idea and virtual world manipulation just partially.

From the technical point of view, we were interested to know how students actually felt interacting with the virtual world client software and manipulating avatars in general. Looking at Question 8, we understand that students found the interaction with the virtual environment somewhat difficult.

To determine the impressions of the students regarding the system feedback value option, we analyzed the different answers collected in both case studies and try to determine if the feedback was actually useful for the students who got it and if it was desirable for the students who did not. Thus, we present the results obtained from our questionnaire for Questions 9, 10, and 11. The result indicates that the students who tested the Feedback version of the application consider the feedback feature as somewhat helpful and not disturbing. On the other hand, the students who tested the No Feedback version overwhelmingly stated their desire to have more feedback from the system during the training process.

One key aspect of a virtual world training application would be to deliver a environment setting in which the virtual scenario can mimic a real one with enough fidelity, so students

Question	Answer Scheme	Feedback		No Feedback		p
		Mean	σ^2	Mean	σ^2	
16	5	2.33	0.78	1.67	0.78	0.048
17	3	2.25	0.62	2.58	0.67	0.22
18	5	1.83	0.72	1.58	0.79	0.43

would feel more immersed during their experience with the system. Therefore, we present the results obtained from our questionnaire for Questions 12, 13, and 14. From these results, we speculate that students considered the level of reality of the simulation a little below acceptable, which in turn, could have affected the degree of immersiveness of their experience.

To analyze the actual contribution of the system feedback to the students' learning process, we performed a t-test analysis over the answers of both groups from questions 16, 17, and 18. The results from the t-test analysis are shown in Table IV.

We found that the only statistically significant result was the efficiency of learning, which was unexpectedly higher for the students who received *no* feedback from the system. Regarding the other questions, students from both groups seem to indicate that they were rather satisfied with the application from the entertainment point of view and that they believe this application could somehow help them to take the correct course of action in situations similar to the one presented in the OpenBioSafetyLab environment.

V. DISCUSSION AND CONCLUSIONS

In this paper, we presented OpenBioSafetyLab, a virtual world based training application for bio-risk management, which has been evaluated by users from the medical sciences in order to obtain valuable feedback of its usability.

Even though the results we obtained from our study were not completely satisfactory regarding the usability aspect of our system, they provided us with important experience on the expectations of non-IT users. From our analysis, we could identify the following issues:

- Although the system feedback was not disturbing, it was found that it was not much helpful either. Since the application was designed as a testing system (not a learning one), the appreciation of the feedback is understandable in the context of reminding them the order in which things should be performed. In this regard, people who did not have feedback would unequivocally state their desire for it.
- While the virtual lab infrastructure and objects were created as accurate as possible, the way of interaction with such objects were constrained by the virtual client itself. Therefore students perceived just a faint sense of presence. For example, touching or grabbing an object was implemented according to the functionality of the very same virtual client interface, which might be perceived as unnatural (e.g.right click and choose from a menu or clicking even when the avatar is nowhere close

TABLE I

FEEDBACK AND NO FEEDBACK VERSIONS OF THE QUESTIONNAIRE USED IN THE STUDY. THE ONLY DIFFERENCE BETWEEN BOTH VERSIONS IS LOCATED AT THE SYSTEM FEEDBACK SECTION. THE FIRST QUESTION REFERS TO A RANDOM IDENTIFICATION NUMBER STUDENTS GET WHEN FINISHING THE TRAINING SESSION.

OpenBioSafetyLab Experience Questionnaire		
Sections	*Feedback Version*	*No Feedback Version*
Preconditions	(1) Please enter the number you got at the end of the Simulation (2) What is your age? (3) What is your sex? (4) Do you have experience with virtual environments (like games)? (5) Do you have experience on how to behave in a BioSafetyLab?	
Acting in the Virtual Environment	(6) How well could you understand the introduction of the training session? (7) How well could you understand acting in the virtual environment through the introduction? (8) How well could you move/act inside the virtual environment?	
System Feedback	(9) How well did the feedback assist you? (10) Was the feedback disturbing?	(11) Would you have liked to have some feedback during the simulation?
Reality Experience	(12) How close was the training scenario to reality? (13) How close the virtual environment represented a real lab environment? (14) How immersed did you feel in the virtual environment?	
User Feedback	(15) Would you use this virtual environment again for learning purposes? (16) Was the training application effective as a study tool ? (17) Did you have fun using the training application? (18) Will the experiment help you to choose the right action in an real emergency situation?	

to the actual object). Improvements in realism will require the use of a better virtual client technology or even haptical input.

- The finding that the learning value was seen higher by subjects who actually did not receive any feedback, could be explained by the fact that the system feedback was implemented as a "reminder", and not presented by a (virtual) "teacher". This could have caused some confusion and even distraction to people that were actually experimenting with the system because of its learning feature. In any case, we think that for learning scenarios, the way feedback is provided has to be improved.

- Virtual world technologies demonstrated their potential regarding the entertainment dimension and their actual value as a knowledge communication media, despite the fact that more than half of the students declare that they do not know if they would use the system again in its current format (Question 15).

As future work, we plan to re-implement OpenBioSafetyLab as a self-learning tool, following the suggestion of our collaborators from the health sciences. We also intend to investigate ways to implement a better client interface to enhance the experience of realism.

The big vision of the OpenBioSafetyLab is to create a new standard for learning and training in the bio-risk management domain. As the first country, we are currently exploring this possibility in Japan.

REFERENCES

[1] B. Schneiderman, "Science 2.0," *Science*, vol. 319, pp. 1349–1350, March 2008.

[2] W. S. Bainbridge, "The scientific research potential of virtual worlds," *Science*, vol. 317, no. 2, pp. 472–476, 2007.

[3] H. Prendinger, S. Ullrich, A. Nakasone, and M. Ishizuka, "MPML3D: Scripting agents for the 3D internet," *IEEE Trans on Visualization and Computer Graphics*, vol. 99, no. PrePrints, 2010.

[4] S. Kumar, J. Chhugani, C. Kim, D. Kim, A. Nguyen, P. Dubey, C. Bienia, and Y. Kim, "Second life and the new generation of virtual worlds," *Computer*, vol. 9, pp. 46–53, 2008.

[5] P. Bourke, "Evaluating second life as a tool for collaborative scientific visualisation," *Proceedings of Computer Games and Allied Technology in Animation, Multimedia, IPTV and Entertainment*, 2008.

[6] "Monolith - molecular visualization for second life," URL:http://www.scilands.org/2007/04/01/molecules.

[7] A. Nakasone, H. Prendinger, S. Holland, P. Hut, J. Makino, and K. Miura, "Astrosim: Collaborative visualization of an astrophysics simulation in second life," *IEEE Computer Graphics and Applications*, vol. 29, no. 5, pp. 69–81, September/October 2009.

[8] K. Gajananan, A. Nakasone, A. Hildebrandt, and H. Prendinger, "A novel three-dimensional collaborative online platform for bio-molecular modeling," *Proceedings of the 10th International Symposium on Smart Graphics*, 2010.

[9] J.-L. Lugrin and M. Cavazza, "AI-based world behaviour for emergent narratives," *Proceedings of the 2006 International Conference on Advances in Computer Entertainment Technology*, no. 25, 2006.

[10] S. Irawati, S. Hong, J. Kim, and H. Ko, "3D edutainment environment: learning physics through VR/AR experiences," *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, pp. 21–24, 2008.

[11] S. Gerbaud, N. Mollet, F. Ganier, B. Arnaldi, and J. Tisseau, "GVT: a platform to create virtual environments for procedural training," *IEEE Virtual Reality Conference*, pp. 225–232, 2008.

[12] N. W. John, "Design and implementation of medical training simulators," *Virtual Reality*, vol. 12, no. 4, pp. 269–279, 2008.

[13] Y. Jung, R. Recker, M. Olbrich, and U. Bockholt, "Using X3D for medical training simulations," *Proceedings of the 13th International Symposium on 3D Web Technology*, pp. 43–51, 2008.

[14] S. Marks, J. Windsor, and B. Wunsche, "Evaluation of game engines for simulated surgical training," *Proceedings of the 5th International Conference on Computer Graphics and Interactive Techniques in Australia and Southeast Asia*, pp. 273–280, 2007.

[15] M. Cavazza and A. Simo, "A virtual patient based on qualitative simulation," *Proceedings of the 8th International Conference on Intelligent User Interfaces*, pp. 19–25, 2003.

[16] WHO, *Laboratory Biosafety Manual - Third Edition*. World Health Organization, 2004.