Implementing Virtual Reality Laboratory Accidents
Using the Half-Life Game Engine, WorldUp, and Java3D

John T. Bell and H. Scott Fogler

University of Illinois Chicago / University of Michigan Ann Arbor

This paper describes recent developments in an ongoing project[1-3] to develop a series of virtual reality based laboratory accident simulations, designed to impress upon users the importance of following proper lab safety procedures, and the potential consequences of not doing so. The primary goal of the project is that users will remember the *experience* of suffering a lab accident longer and more vividly than a written set of lab safety rules, and will hopefully conduct themselves in a safer manner as a result. A secondary goal is to explore a variety of different development platforms, to determine which are most practical and effective for this type of first-person educational simulation. The following sections will describe the current status of the overall project, and describe the most recently developed modules.

Project Overview

It is well known that people remember things that they experience first-hand much longer and more vividly than things that they merely read or hear[4, 5]. Specifically in the area of laboratory safety, anyone who has ever experienced a laboratory accident will remember that experience much longer than any set of written lab safety rules. However it is obviously not practical to deliberately stage lab accidents merely to drive home the importance of following proper lab safety procedures. Virtual reality, on the other hand, offers the opportunity to create realistic first-person simulations of lab accidents that will have a much stronger impact on the user than any set of written rules (although obviously not as strong an impact as real accidents.)

Virtual reality is an emerging computer interface technology designed to make the user believe, as much as possible, that what they are experiencing is "real". While no system currently available has attained that goal completely, there are many high-end systems that can suspend users' disbelief to remarkable extents, so that at least for a short while they are willing to forget that it is not real. Advanced systems achieve this level of realism using ultra-fast graphics supercomputers and specialized interface equipment, including stereoscopic displays, head-mounted displays, motion trackers, haptic feedback devices, advanced audio processing systems, and even olfactory "display" systems.[6-13]

Low-cost virtual reality[11, 14-16] does not achieve the same levels of realism and believability, but has the advantage of being within the grasp of a much broader audience, and with recent advances in consumer-grade computer graphics cards, the gap between high-end and low-end systems is rapidly diminishing. Virtual reality has successfully been applied as an information delivery system, in both the educational[17-28] and scientific visualization[7, 8, 29-35] arenas. This project strives to deliver simulations having as high an impact as possible while supporting computer equipment that is commonly available to students and other interested parties. Special devices such as head-mounted displays are supported (where applicable), but not required.

Project Status

Table 1 outlines the different safety rules and development platforms that are being implemented for this project. The items in red, (also indicated by asterisks), are the new developments since the last time that the results of this project have been reported[1]. Modules marked "Done" are available on the web for free download or direct use.

Safety Rule	Platform				
	VRML	WTK	WorldUp	Java3D	Half Life
Safety Glasses	Done	Done	Under Development		
Storage	Done	***Done**		***Under Development**	***Under Development**
No Food	1st Draft	Done			
Clear Aisles	1st Draft	***Done**		***Done**	
Gas Cylinders		***Done**	***Done**		
Proper Attire		***Done**			
Know Locations		***Done**			
No HorsePlay		***Under Development**			
Virtual Reactor	NA	Under Development	NA	NA	NA
SafeHunt	NA	Done	NA	NA	NA

Table 1: Current Status of Virtual Reality Laboratory Accident Scenarios

Descriptions of Newly Developed Modules

The following modules have been developed since the last paper was written on this project[1]. Some of the modules are complete and available for download, and others are still under development. The features and benefits of each development platform are discussed under the first module (in the paper) to use a particular platform. Because the VRML platform does not have any new modules to describe, it is described here.

VRML (Virtual Reality Modeling Language[36]) is a web-based description language similar to HTML (HyperText Markup Language.) The difference is that VRML allows for the

description of three-dimensional objects and environments, and in the case of VRML2, it also provides support for movement, interactive response, and sound. The attractive benefit of VRML is that it can be run in any web-browser having a suitable plug-in, and there are several 3D object creators (such as 3D Studio Max) that can save their work in VRML format. Our experience with VRML, however, has been somewhat mixed. We have been able to produce a few good simulations using this tool, but it has required a lot of work and does not allow students to draw extensively on their C programming background. VRML has also not lived up to its promise of cross-platform compatibility in our experience, and in general the strong interest and support that VRML enjoyed for a few years has died down. The automatic VRML generators tend to produce overly complex wasteful code, which may be less of an issue as computers have gotten faster, but which is still very difficult to maintain manually. Although we have not abandoned this platform, there are no current development activities or plans for it either.

Always Store Chemicals Properly (WorldToolKit and Half-Life)

This safety rule states that all chemicals should be stored in proper locations when not in use. There are special storage locations for flammable and radioactive materials; acids and bases should not be stored together, and of course, chemicals should not be left laying around the lab benches. This rule was first addressed in the VRML platform[3], has recently been completed in the WorldToolKit format, and is under development using the Half-Life game engine.

In all three platforms the basic scenario is that a number of bottles have been left carelessly on the lab bench, and it is the responsibility of the user to clean up the lab by putting each of the bottles in its proper location. If the user succeeds in cleaning up the lab in a timely fashion, then they receive kudos for their efforts, but if they fail to do so, (either because they misplaced materials or because they simply failed to clean up at all), then the lab blows up.

The WorldToolKit platform is the one that we have been most successful and productive with in our development group. It is a library of C language functions designed for the development of virtual reality programs, produced by the Sense8 Corporation[37]. Students write C language routines to drive the simulations, calling on the WorldToolKit functions to handle low-level graphics and device management details. This platform is one the students are very comfortable with, based on the C programming skills they have learned in class, and it is relatively easy to develop new simulations by building on our accumulated experience with previous work. WorldToolKit also offers the benefit of multiple platform support, which we have pursued with limited success. (WorldToolKit has versions for MS Windows, Silicon Graphics, Sun, HP, and Linux. Efforts to port the lab accident simulations from the Windows platform to the other platforms have not yet been completely successful, but are still underway.) WorldToolKit supports advanced virtual reality hardware devices, such as head-mounted displays, stereoscopic displays, and even the CAVE™[9, 30]. The VR lab accidents are written to support many of these devices, but do not require their use. The down side of WorldToolKit is that the binary programs have to be downloaded and installed on the users' computers, and separate versions must be created for each desired hardware platform. Figure 1 illustrates three scenes from the WorldToolKit implementation of the proper storage lab accident simulation.
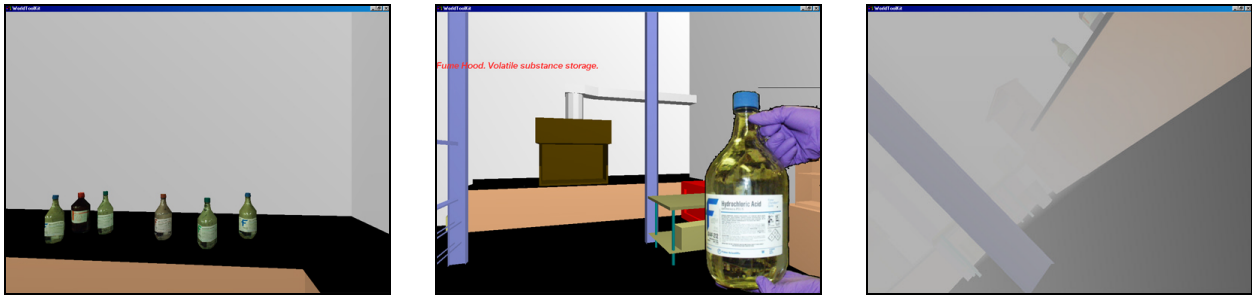
Figure 1: WTK Simulation - (A) Bottles on the bench to be cleaned up. (B) Carrying a bottle of acid to the fume hood. (C) Weak and dying after the explosion.

Half-Life is a new development platform for the VRUPL group, and the first simulation using this new platform is still under development. Half-Life is a commercially available computer game that normally involves wandering through underground dungeons and shooting bad guys with machine guns and similar weapons. However it is also possible to develop new environments and simulations, (modifications, or "mod"s), which can then be driven by the Half-Life game engine[38-42]. Half-Life includes a graphical development and programming environment that allows one to quickly develop new environments, and includes support for sophisticated lighting effects, actions triggered by users' movements, and perhaps most importantly for the lab accidents, explosions. It also includes human characters that can be programmed to interact with the user, and has the added benefit of being very popular and well known within the primary target audience (college age students) of the lab simulations. The Half-Life development environment also includes a large number of useful components, including radioactive symbols, lab coats, a wide variety of textures, and furniture such as desks and fume hoods. Images of the Half-Life simulation can be seen in Figure 2.

Figure 2: Half-Life Simulation - (A) General lab furniture. (B) Fume hood and lab bench with bottles. (C) Explosion, and blood on the floor.

Always Keep Aisles Clear (WorldToolKit and Java3D)

This safety rule focuses on the need to keep laboratory aisles, particularly those leading to exits, clear and free from obstructions. The VRML version was one of the first simulations developed for this project, and needs to be re-done. New developments this year are that this simulation has been completed in both WorldToolKit and Java3D formats. The scenario in both of these implementations is that a fire breaks out in the laboratory blocking easy access to the door through which the user entered. If the aisles are clear, then the user can quickly exit through an

alternative route, but if they are cluttered, then it requires extensive additional effort to exit the lab, possibly at the cost of the user's life. The two new simulations are dramatically different from each other, and are also different from the original VRML simulation. Images from both versions of this simulation are shown in Figure 3.

The WorldToolKit simulation involves a laboratory with two exits, and requires the user to choose ahead of time whether the rule is being followed or not. A fire breaks out near the door through which the user entered, forcing the user to make some potentially difficult choices as to how to exit the laboratory. If the aisles are clear, the best option is to run through the remaining open doorway. If not, the user must choose between taking the time to move the clutter blocking the doorway or attempting to run through the fire to safety, either of which can be hazardous to the users health. The lab fills with smoke as the simulation proceeds, so remaining in the lab is not a good choice in any case. This simulation is also equipped with a "health meter", and provides different feedback messages at the end depending on what choices the user has made and how they performed overall.

The Java3D simulation has only the cluttered option, which initially has a single clear path across the room. When the simulation starts, the user is taken along the clear path to the far corner of the room, and then a fire breaks out blocking the easy path back. The user must then wind their way through a maze of tables and clutter to attain the exit. A conclusion screen plays back the path that the user took, and compares it to the most direct route in an uncluttered lab. Time comparisons are shown, indicating how much time the user lost due to the clutter in the lab. There are six different versions of the simulation, which differ by the placement of clutter (and the resulting maze layout.) The initial version is selected randomly, and then users can select specific versions for repeat runs.

Java3D is another new development for the VRUPL lab, and is based on the Java programming language. It consists of a package of Java classes that support the description of three dimensional objects, environments, and actions, including user interactions. Because Java3D is a relatively new development, there are not a lot of resources available, other than on-line documents[43]. (A search of both Amazon.com and the O'Reilly web site for "Java3D" found zero matches for either source.) The student who developed the Java3D lab accident had learned basic Java in a programming course, but was entirely on his own regarding the Java3D extension. The result was a pretty good simulation that took a long time to develop due to the steep learning curve and lack of available references. Java3D again has the benefit of being runnable in any web browser for which the user has the appropriate plug-in.
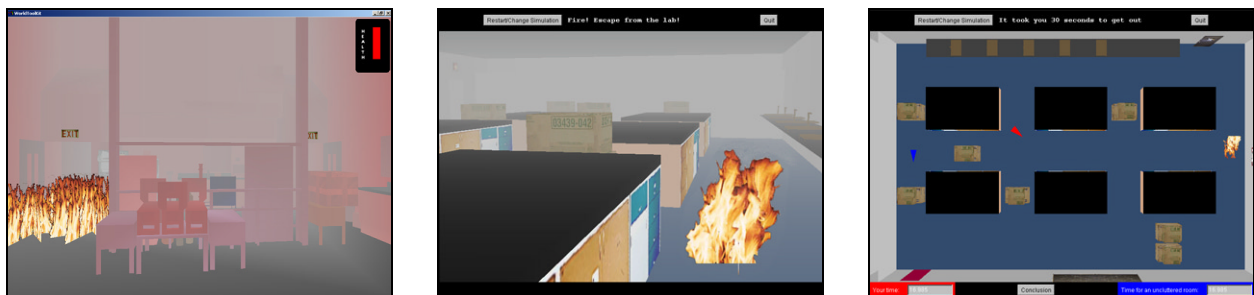

Figure 3: (A) WTK - Both exits blocked and smoke thickening. (B) Java3D - Fire, smoke, and clutter in the lab. (C) Java3D - Followup comparing good and bad exit routes.

Securely Fasten Gas Cylinders (WorldToolKit and WorldUp)

This safety rules centers around the fact that compressed gas cylinders can turn into torpedoes should the nozzle ever break off. For this reason compressed gas cylinders should always be securely fastened to bench or cylinder rack. In these simulations there is a laboratory containing two gas cylinders – one that is securely fastened, and one that is not. As the user enters, the unfastened cylinder falls over, breaking the nozzle and causing it to fly around the room causing havoc. One of the many things that gets broken is the nozzle on the secured cylinder, which then proceeds to shoot a jet of high-pressure gas harmlessly up at the ceiling. At the end of the simulation, the cylinder shoots straight at the user's face, followed by an image of a tombstone. This simulation was developed in both WorldToolKit and WorldUp by the same student who developed the no food allowed simulations on the WorldToolKit platform. (Note: One reviewer wanted to know what our lab walls were made of, because the one time he had seen a gas cylinder go ballistic it had shot straight through the nearest wall and landed some distance away, as opposed to bouncing around the room. The developers of this simulation have never had the (mis)fortune to experience a broken gas cylinder first hand.) Figure 4 shows scenes from both the WorldToolKit and WorldUp versions of the gas cylinder accident simulation.
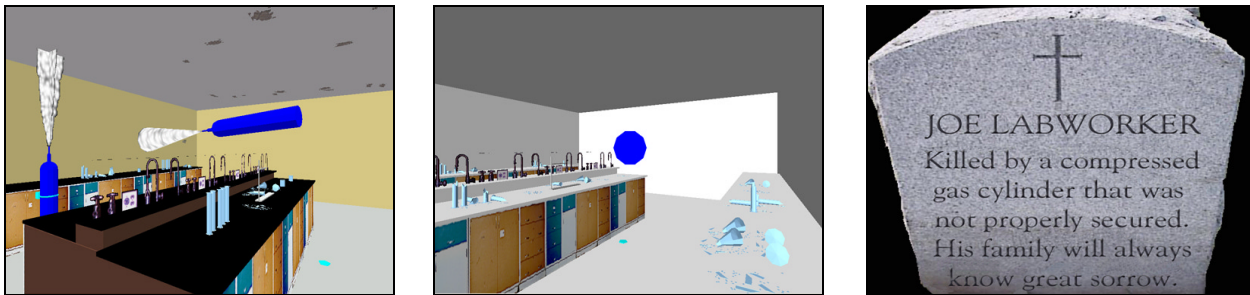

Figure 4: (A) WTK - Secured and unsecured gas cylinders. (B) WorldUp - Oncoming cylinder ends the simulation. (C) Both versions reach the same conclusion.

WorldUp is another product of the Sense8 Corporation, and is in fact built upon the WorldToolKit library. WorldUp presents the developer with a graphical development environment, as opposed to the extensive C Programming required when using WorldToolKit directly. WorldUp developers predominantly drag, drop, click, and fill out forms with design parameters. There is also support for visual basic scripts, which are attached to objects or the scene graph in the form of tasks. One of the attractive benefits of the WorldUp platform is that the simulations can be run directly from a web browser, using a plug-in that is freely available from the Sense8 web site.

WorldUp is supposed to be an "easier" development environment than WorldToolKit, and so our initial approach was to assign newcomers to the lab to work on WorldUp simulations. This approach did not work well for us, possibly because no one in the lab (including the lab director) had experience with WorldUp, and so newcomers had no one to turn to for help. More recently, we tried assigning an experienced and highly competent WorldToolKit programmer to develop some WorldUp simulations, and she was able to successfully develop a WorldUp simulation for the gas cylinder safety rule, which she had previously developed in WorldToolKit.

Always Wear Proper Attire in the Laboratory (WorldToolKit)

For this simulation the user must first dress for work, deciding whether to wear sandals or more substantial shoes, and whether or not to wear a lab coat. Upon entering the lab, accidents occur that expose the user's feet and/or arms to harmful chemicals. If they are wearing proper attire, then they have a certain amount of time to remove the affected clothing and activate the lab safety shower to wash off any remaining residue. If they have not chosen proper attire, then their arm and/or feet quickly dissolve down to the bone, which is hopefully an experience that they will not soon forget! Figure 5 shows four scenes from this simulation.
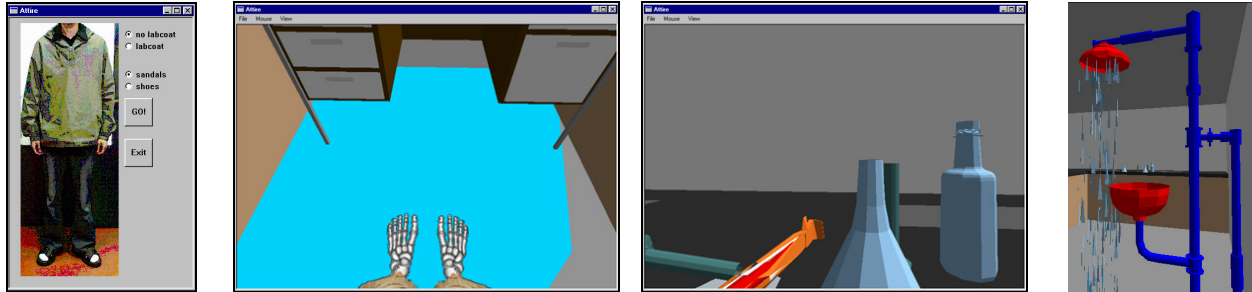

Figure 5: (A) User selects attire before entering the lab. (B) Sandals were not a good choice! (C) Neither was neglecting a lab coat! (D) A safety shower is available.

Know the Locations of Fire Extinguishers and All Exits (WorldToolKit)

In this simulation, (shown in Figure 6), a fire breaks out that blocks both the exit door and easy access to the fire extinguisher. If the user has paid attention as they entered the lab, then they can access the extinguisher through an alternate route and extinguish the flames in time to escape the lab safely. If they are not able to do so in a timely fashion, however, then they are quickly overcome by the smoke and flames. As a complicating factor, the second exit to the lab is blocked by clutter, which is immovable in this simulation.

Figure 6: (A) Users must note the fire extinguisher as they enter the lab, (B) Because it is hard to see it through the flames later. (C) The consequences of not finding it can be fatal.

No Horseplay Allowed in the Laboratory (WorldToolKit)

This simulation (which is still currently under development) has been postponed for some time because it requires the presence of human characters in the laboratory. Humans are much more difficult to simulate than lab benches and beakers because they require a much greater level of detail and realistic movement before they are believable in a simulation. The two breakthroughs that have made this possible are the increased graphics power of modern personal computers, and the human figures that were extracted from other development platforms. Work is still proceeding on achieving realistic motion as the character moves in the laboratory. The general scenario for this simulation is that someone throws a ball in the lab, which knocks a co-worker into the lab bench full of chemicals, and then onto the floor, as shown in Figure 7

Figure 7: (A) A thrown ball hits the scientist, (B) Knocking him to the ground, (C) and severely injuring him.
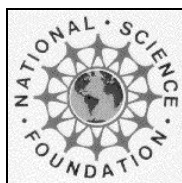
Accessing and Running the Simulations

All of the completed simulations are freely available on the VRUPL web site[44] (Follow the "VR Lab Accidents" link), and are run in a variety of ways depending on the development platform. The VRML and Java3D versions are run directly from the web browser, using the appropriate plug-in. (Links to access the necessary plug-ins are provided.) The WorldToolKit versions must be downloaded onto the user's computer and installed – The MS Windows versions of these all use the standard InstallShield™ installation method, which also provides un-installation capability. The WorldUp simulation can either be downloaded and installed or run directly in the browser, again using a suitable plug-in.

Performance depends primarily on the performance of the graphics card, and to a lesser extent on the performance of the computer. The WorldToolKit and WorldUp simulations in particular are based on OpenGL graphics, and so a modern OpenGL graphics accelerator card (e.g. GeForce) is highly recommended. There may be problems with Windows 95/98 on computers using DirectX graphics instead of OpenGL graphics. One caveat is that Sense8 products recognize the presence of an accelerated card by checking for a 16-bit color setting, which means that if the color setting is set to anything else (e.g. 8, 24 or 32 bit color), then Sense8 will perform graphics calculations in software rather than hardware. For OpenGL systems, set the colors to 16-bit to ensure use of the hardware acceleration; For DirectX graphics cards, setting the colors to 24 or 32 bit can sometimes prevent Sense8 from attempting to use hardware acceleration, thereby avoiding the associated problems.

Acknowledgements

This project was supported, in part
by the

National Science Foundation

Opinions expressed are those of the authors
and not necessarily those of the Foundation

Bibliographical References

1. Bell, John T. and H. Scott Fogler, "No Food Allowed - The Latest Virtual Reality Laboratory Accident", *Proceedings of American Society for Engineering Education Annual Conference,* Montreal, Quebec, Canada, American Society for Engineering Education, 2002.
2. Bell, John T. and H. Scott Fogler, "Virtual Reality Laboratory Accidents", *Proceedings of American Society for Engineering Education Annual Conference,* Albuquerque, NM, American Society for Engineering Education, 2001, http://www.cs.uic.edu/~jbell/Professional/Papers/ASEE2001.pdf.
3. Bell, John T. and H. Scott Fogler, "Virtual Laboratory Accidents Designed to Increase Safety Awareness", *Proceedings of American Society for Engineering Education Annual Conference,* Charlotte, NC, American Society for Engineering Education, 1999, http://www.cs.uic.edu/~jbell/Professional/Papers/asee_safe.pdf.
4. Dale, Edgar, *Audio-Visual Methods in Teaching,* Third Edition, Holt, Rinehart, and Winston, 1969.
5. Kolb, D. A., *Experiential Learning: Experience as the Source of Learning and Development*, Prentice-Hall, Englewood Cliffs, NJ, 1984.
6. Aukstakalnis, Steve and David Blatner, *Silicon Mirage: The Art and Science of Virtual Reality*, Peachpit Press, 1992.
7. Brooks, Frederick P., Jr., *et al.*, "Project GROPE - Haptic Displays for Scientific Visualization", *Computer Graphics: Proceedings of SIGGRAPH '90,* **24(4)**, pp 177-185, 1990.

8. Byron, S. and C. Levit, "The Virtual Windtunnel: An Environment for the Exploration of Three-Dimensional Unsteady Flows", *Proceedings of IEEE Visualization '91*, 1991.

9. Cruz-Neira, Carolina, Dan J. Sandin, and Thomas A. DeFanti, "Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE", *Proceedings of Siggraph 93,* New York, ACM Press, 1993.

10. Eglowstein, Howard, "Reach Out and Touch Your Data", *BYTE,* , 1990.

11. Larijani, L. Casey, *The Virtual Reality Primer*, McGraw Hill, 1994.

12. Pimental, Ken and Kevin Teixeira, *Virtual Reality: Through the New Looking Glass,* Second Edition, Windcrest Books, 1995.

13. Sutherland, Ivan, " The Ultimate Display", *Proceedings of the IFIP Congress,* , 1965.

14. Jacobson, Linda, *Garage Virtual Reality*, Sam's Publishing, 1994.

15. Stampe, Dave, Bernie Roehl, and John Eagan, *Virtual Reality Creations*, The Waite Group Press, 1993.

16. Wodaski, Ron, *Virtual Reality Madness*, Sams Publishing, 1993.

17. Bowman, Doug A., *et al.*, "The Educational Value of an Information-Rich Virtual Environment", *Presence,* **8(3)**, pp 317-331, 1999.

18. Brelsford, John W., "Physics Education in a Virtual Environment", *Proceedings of Human Factors and Ergonomics Society 37th Annual Meeting*, 1993.

19. Bricken, Meredith, "Virtual Reality Learning Environments: Potentials and Challenges", *Computer Graphics,* **25(3)**, 1991.

20. Byrne, Chris, "Virtual Reality and Education", HITL Technical Report No. R-93-2, Human Interface Technology Laboratory, Seattle, WA, 1993.

21. Brooks, Fred, "Virtual Reality in Education: Promise and Reality Panel Statement", *Proceedings IEEE Virtual Reality Annual International Symposium (VRAIS '98),* , p 208, 1998.

22. Emerson, Toni and Debra Revere, "Virtual Reality in Training and Education: Resource Guide to Citations and Online Information", *HITL Technical Publications,* **B-94-1 (1997 Revise)**, 1994.

23. Helsel, Sandra, "Virtual Reality and Education", *Educational Technology,* , 1992.

24. Loftin, R. Bowen, Mark Engelberg, and Robin Benedetti, "Applying Virtual Reality in Education: A Prototypical Virtual Physics Laboratory", *Proceedings of IEEE Symposium on Research Frontiers in Virtual Reality,* Los Alamitos, CA, IEEE Society Press, 1993.

25. Pantelidis, Veronica S., "Virtual Reality and Education: Information Sources", ftp:ftp.hitl.washington.edu/pub/scivw/citations/VR-ED.html.

26. Seidel, Robert J. and Paul R. Chatelier, *Virtual Reality, Training's Future?*, Plenum Press, New York, 1997.

27. Staff, "Immersive VR Tests Best", *CyberEdge Journal,* **4(6)**, 1994.

28. Youngblut, Christine, "Educational Uses of Virtual Reality Technology", *Technical Report IDA Document D-2128,* , 1998.

29. Bajura, Mike, Henry Fuchs, and Ryutarou Ohbuchi, "Merging Virtual Reality With the Real World: Seeing Ultrasound Imagery Within the Patient", *Proceedings of SIGGRAPH '92*, ACM SIGGRAPH, 1992.

30. Cruz-Neira, C., *et al.*, "Scientists in Wonderland: A Report on Visualization Applications in the CAVE Virtual Reality Environment", *Proceedings of IEEE 1993 Symposium on Research Frontiers in Virtual Reality*, 1993, http://www.evl.uic.edu/paper/template_pap_project.php3?indi=133.

31. Cruz-Neira, Carolina, *et al.*, "Integrating Virtual Reality and High Performance Computing and Communications for Real-Time Molecular Modeling", *Proceedings of High Performance Computing '95*, 1995.

32. Friedhoff, Richard, *Visualization: The Second Computer Revolution*, Abrams, New York, 1989.

33. Johnson, A., *et al.*, "SANDBOX: An Interface to Scientific Data Based on Experimentation", *Proceedings of Fifth Eurographics Workshop on Visualisation in Scientific Computing,* Rostock, Germany, 1994, http://www.evl.uic.edu/paper/template_pap_project.php3?indi=127.

34. VACCA, John, "NASA Projects: Part 1", *VR World,* , 1995.

35. Warwick, Kevin, John Gray, and David Roberts, *Virtual Reality in Engineering*, IEEE, London, 1993.

36. Carey, Rikk and Gavin Bell, *The Annotated VRML 2.0 Reference Manual*, Addison-Wesley, 1997, http://rikk.best.vwh.net/Book/.

37. Sense8 Corporation, "Sense8 Corporation Web Site", http://www.sense8.com.

38. MetaMod, "MetaMod main web site", http://www.metamod.org.

39. Broome, Jeffrey, "botman's bots - Half-Life bot development", http://www.planethalflife.com/botman/.

40. Ciragan, Mete, "MilkShape3D Tutorials", , 2003, http://www.swissquake.ch/chumbalum-soft/ms3d/tutorials.html.

41. Cook, Steve G., "Shadowman's Half-Life SDK2 Resources", http://planethalflife.com/hlsdk2/.

42. Wavelength, "Your Source for Half-Life Editing", http://www.thewavelength.net/.

43. Sowizral, Henry, Kevin Rushforth, and Michael Deering, *The Java 3D API Specification*, Addison-Wesley, 1998.

44. Bell, John T., "Virtual Reality Undergraduate Projects Laboratory Web Site", http://www.vrupl.evl.uic.edu.

Biographical Information

John T. Bell

(Lecturer, Department of Computer Science, University of Illinois, Chicago, 312 413-9054, JBell@uic.edu, http://www. cs.uic.edu/~jbell) Dr. Bell holds graduate degrees in Chemical Engineering and Computer Science, and combines both skills by developing computer simulations of chemical engineering related phenomenon. Recent work has focused on the application of interactive immersive three-dimensional display technology (a.k.a. VR) to the presentation of scientific, engineering**,** and educational information.

H. Scott Fogler

(Vennema Distinguished Professor of Chemical Engineering, University of Michigan, H.Scott.Fogler@umich.edu, http://www.engin.umich.edu/dept/che/people/fogler.html.) Dr. Fogler is the author of four books and over 160 research publications. Awards include the ASEE Meriam-Wiley Distinguished Author Award (1996), the University of Michigan Distinguished Faculty Achievement Award (1996), the AIChE Warren K. Lewis Award (1995), and the Chemical Manufacturer's Association National Catalyst Award (1999).