Virtual Reality: At the Service of GEN-IV, and V, and ….,

Rizwan-uddin[1], Nick Karancevic[1] and Sukru Tikves[2]

[1]*Department of Nuclear, Plasma and Radiological Engineering*
University of Illinois, Urbana, IL 61801, USA

[2]*Department of Computer Science and Engineering*
Hacettepe University, Ankara, Turkey

Tel: (217) 244-4944; Fax: (217) 333-2906;
E-mails: rizwan@uiuc.edu nick@karancevic.com stikves@yahoo.com

Abstract — *Generation II nuclear reactors have benefited significantly from experience gained over the last few decades. In the United States, average down time of these reactors has dropped significantly, leading to an increase of total electricity output despite an almost stagnant number of units. Both, planned (re-fueling) outages as well as unplanned outage times have dropped. Next to longer core life, better training and operator experience are the prime reasons behind this improvement. Nuclear engineers are factoring these experiences into the design of most of the next generation of nuclear reactors. However, with the help of virtual reality representation of these new designs, additional optimization at the design time as well as personnel training after the completion of the reactors is possible. Virtual reality systems can help in optimized layout for operational as well as component replacement exercises. In addition, these systems can also be used for operator training. Furthermore, re-fueling personnel may also be trained on these systems to reduce the downtime. Since GEN-IV designs are still in development stages, the best approach would be to develop a general framework of virtual reality system that can be tailored to specific designs, as needed. As a first step toward this direction, a virtual reality framework is being developed for a generic nuclear reactor. A four-walled CAVE, operating at the National Center of Supercomputing Applications at UIUC campus, is being utilized for this development. Some background material and results of the initial stages of development are presented here.*

I. INTRODUCTION

Many GEN-IV candidate designs are currently under investigation. Technical issues related to material, safety and economics are being addressed at research laboratories, industry and in academia. After safety, economic feasibility is likely to be the most important criterion in the success of GEN-IV design(s). Lessons learned from the designers and operators of GEN-II (and GEN-III) reactors must play a vital role in achieving both safety and economic feasibility goals. Though there are numerous lessons, we will focus on two of them.

It is well known that despite the fact that no new reactors have come online, the total electrical power generated by nuclear power plants (NPPs) in the US over the last 20 years has continued to increase. Longer core life has been the primary reason. Two other important factors that have contributed to an increase in total electricity generated by nuclear are: 1) significant reduction in refueling time over the last twenty years; and 2) better operator training that has led to reduction in planned and un-intended downtime of GEN-II reactors. Economic feasibility of GEN-II reactors would be much worse today if it was not for the improvements in the overall availability of NPPs due to these two factors. Without any doubt, reducing the number of refueling as well as reducing the downtime during refueling have played an extremely important role in making GEN-II reactors economically competitive in today's de-regulated environment. This is reflected in several proposed designs with a much longer core life (with as much as 30-40 year core life) and in continued efforts to reduce refueling time.

In addition, reduction in safety and maintenance related shutdowns from the early days of PWR and BWR operations has also contributed to increased availability of NPPs. Realizing that the total cost associated with the downtime, in many cases in NPPs, can be much more than the cost of the repaired part, there is ample reason to believe that significant gains can be achieved by minimizing the downtime due to repair and replace operations. In addition to repair and maintenance, downtime also results due to accidents. Minimizing number of accidents that lead to operations at lower power or to complete shutdown, is obviously a desirable goal. [The "political" cost of accidents to nuclear power is probably much more than the economic cost.] Most nuclear accidents have been attributed to human factors (errors). Hence, in addition to an optimized design for refueling, better personnel training for online and offline operation and maintenance, and better training to minimize accidents can also play an important role in the optimization of GEN-IV reactors. Other industries have also faced similar challenges. Corresponding issues in other industries have been addressed by taking better advantage of the latest technological developments.

Many technologies that have helped the automobile, aeronautics, civil and, to certain extent, chemical industries in becoming more efficient revolve around better use of computing power that has become available over the last 2-3 decades. Computing power is already playing a significant role in better simulation of the physical processes leading to lower conservatism, and hence improving the efficiency of NPPs. This is evident from the large body of work on coupled neutronics-thermalhydraulics simulations, much more detailed simulation of the core physics, shift toward transport and Monte Carlo methods instead of diffusion theory, etc.

However, other technologies that are now available due to increased computing power, such as computer aided design and manufacturing, 3D modeling, etc., have also allowed significant improvements in the design departments of the automobile, aeronautics, civil and chemical industries. Specifically, these technologies have played a significant role in planning layout and operator training. A case can be made of the compact automobile designs of the early days that led to cramped conditions under the hood, which in turn made repairs much more time-consuming and expensive. Extreme examples include necessity to remove several components before a $5 belt could be replaced. In comparison, newer designs, carried out with the help of CAD software, lead to much *cleaner* layout of the parts, and a "mechanic-friendly" environment under the hood.

Though some efforts have been put in the context of GEN-II reactors and somewhat more with GEN-III reactors, nuclear industry has not taken full advantage of these technologies. In the case of nuclear power, in addition to layout planning, these technologies may also significantly help in operator training, and training of personnel in maintenance and repair departments. They may also offer better design optimization to more efficiently address regulatory issues such as plant safety and fire control.

While the field of 3D modeling has significantly matured, the next frontier is the full exploitation of *virtual reality* environments [1]. Efforts are already underway in conventional fields of engineering in this direction. For example, a project is sponsored by the Deere Co at Iowa State University to develop and evaluate a virtual environment for use in training, assembly, and maintenance methods development [2]. Such design-time optimization can also significantly benefit GEN-IV reactors.

There is currently very little research and development in the nuclear arena in these areas. Virtual Reality (VR) systems are being developed and applied at Argonne National Laboratory (ANL) to address issues from several different branches of science and engineering, including nuclear [3]. A detailed 3D model of an AP-600 reactor is being developed. [Other applications of this technology at ANL include visualization of CAD data, crashworthiness experiments and computational combustion.] Work is proceeding in France on developing computer-aided motion planning for nuclear maintenance. Researchers are modifying off-the-shelf 3D CAD modeling tools to suit the needs of nuclear industry. Specifically, fully functional CAD and computer aided motion (CAM) capabilities are now considered essential over the life of the NPP to "evaluate the feasibility of new maintenance tasks." It also is considered a useful tool for "communication and negotiation between all parties involved in a maintenance operation" [4]. Japanese and S. Korean researchers are also working along similar lines. Korea Electric Power Research Institute is devoting significant resources to develop a virtual control room [5]. The fact remains that only relatively few groups in the nuclear field are taking advantage of these developments.

II. TECHNOLOGIES AVAILABLE

Several technologies are now available that can be utilized in the development of an effective virtual environment framework to optimize GEN-IV reactor designs. Some of these are discussed below.

II.A. Graphical User Interface

GUIs or Graphical user interfaces (at the 2D level) are now commonplace in nuclear software. However, there are still many applications that rely on old-style input/output interfaces. 2D GUIs will be selectively used to simulate the perception of (2D) monitors and other 2D displays commonly used in the control rooms. These can be easily developed using, for example, visual basic or other similar packages.

II.B. CAVE (CAVE Automatic Virtual Environment)

The CAVE (so named after the "Allegory of the CAVE" in Plato's Republic) was first developed at the Electronic Visualization Laboratory (EVL) at the University of Illinois in Chicago. A CAVE has now been operational at the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign for many years. The CAVE (CAVE Automatic Virtual Environment) is a projection-based VR system that surrounds the viewer with 4 screens. The screens are arranged in a cube made up of three rear-projection screens for walls and a down-projection screen for the floor; that is, a projector overhead points to a mirror, which reflects the images onto the floor. Four basic components that comprise the CAVE are: the computers; the graphics systems; the tracking system; and the sound system [6].

The CAVE (see Fig. 1) works by reproducing many of the visual cues that brain uses to decipher the world around. Information such as the differing perspectives presented by the eyes, depth occlusion, and parallax (to name a few) are all combined into the single composite image that one is conscious of, while the rest is decoded by the brain to provide the depth cues. The CAVE must reproduce all of this information in real-time, as one moves about in the CAVE. The projected images are controlled by an SGI Onyx with two Infinite Reality graphics pipelines, each split into two channels. Sixty different images are displayed every second on each of the CAVE's four walls at very high resolution.


Fig. 1. A 3D virtual environment.

The CAVE's graphics system is broken into two parts: the images projected onto the CAVE's walls, and the shutter glasses that make the images appear three-dimensional (Fig. 2). The images projected onto the walls of the CAVE are provided by four Electrohome CRT projectors that surround the CAVE [6].

Two perspective-corrected images are drawn for each frame: one for the right eye and one for the left. Special Stereographics' CrystalEyes liquid crystal shutter glasses are worn that ensure that each eye sees only the image drawn

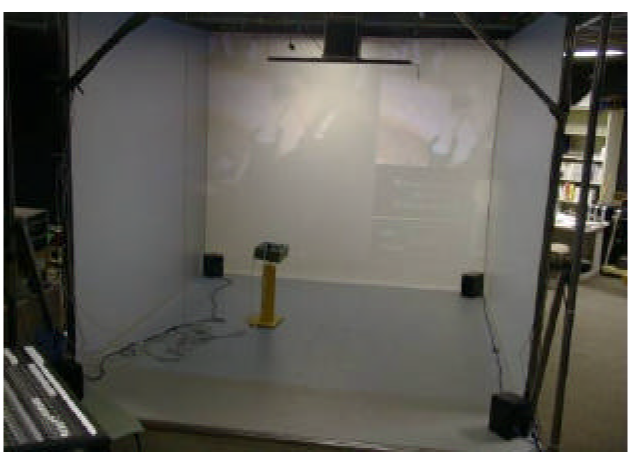
Fig. 2. The CAVE’s visual and audio output hardware.

for it. This creates a stereoscopic effect where the depth information encoded in the virtual scene is restored and conveyed to the eyes of those using it. The CAVE has an extremely advanced tracking system that enables it to constantly track the position and orientation of the special tracked glasses (Fig. 3). The six-degrees-of-freedom head-tracking device is mounted on the special glasses. The user stands inside the CAVE wearing these special (tracked) glasses. The person wearing the tracked glasses controls the viewpoint of the CAVE. They can look around the corner of an object, step behind it, look underneath it, or anything else that they could do in real life. A wand (a 3D mouse; also called a *CAVE-wand*) with buttons is the interactive input device (Fig. 4). The CAVE Wand is also attached to the Tracker Control Unit via a wire and allows the user to walk around (with the joystick in the middle) or interact with the virtual world through the push buttons. Currently, there are two types of wands. Both use the Ascension Flock of Birds tracking system [1], but have different control devices. The primary (new) wand has three buttons and a pressure-sensitive joystick. It is connected to the CAVE through a PC that is attached to one of the Onyx's serial ports. A server program on the PC reads data from the buttons and joystick and passes them to the Onyx. The older wand just has three buttons, and is attached to the mouse port of the Onyx. There are also "simulated" tracking options available, using either the keyboard and mouse or a spaceball. The use of one or another is transparent to the CAVE programmer, since it is defined in the CAVE configuration file. Systems typically have two sensors, one for tracking the user's head, and another for the wand [6].

In addition to its extraordinary graphics capabilities, the CAVE also provides superb audio facilities. It has an eight channel audio system with state-of-the-art digital audio support that is controlled by an advanced sound mixing board [6].

For testing, one can run the CAVE using any number of walls simultaneously. The number of CAVE walls used does not affect the program. The CAVE library determines which

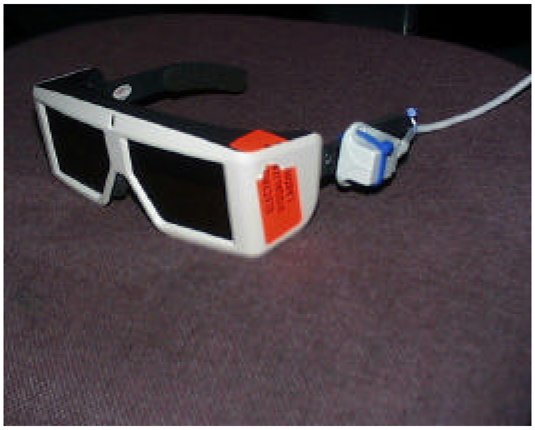

Fig. 3. These glasses enhance the user's perception of a 3D environment

walls are to be used and does the necessary setup when the program starts.

Currently there are many CAVEs operational around the world. One on the campus of University of Illinois is located at the Beckman Institute for Advanced Technology in Urbana, and is used by NCSA's Visualization and Virtual Environments Group to conduct various types of research in the fields of virtual reality and scientific visualization.

Fig. 4: The CAVE's input device, CAVE Wand, is an advanced 3D mouse/joystick with more degrees of freedom than a conventional device for a similar purpose.

II.C. ImmersaDesk

The ImmersaDesk is a drafting-table format virtual prototyping device. The *original* ImmersaDesk had a 4 x 5-foot rear-projected screen tilted at an angle of 45 degrees. The size and position of the screen gives a wide-angle view and the ability to look down as well as forward. Except for the display device the same hardware is used as in the CAVE. User wears CrystalEyes stereo glasses. Stereo emitters are placed behind the screen. ImmersaDesk can use the same tracking systems that the CAVE uses. New models of ImmersaDesk are increasingly using flat screen monitors instead of projector-based screens [7].

II.D. VISBOX

Developed by some members of the CAVE user group at the NCSA, VISBOX (Fig. 5) is a poor-man's CAVE. It is a single wall system at a fraction of the cost of a CAVE. However, some of the features, such as the head tracking system, are superior to those available in the more expensive CAVE. User in the VISBOX cannot turn her face by 90 degree and still find immersed in the same environment, as is the case in the CAVE. However, applications are fully compatible with CAVE. Hence, development can be carried out on VISBOX, and then fine-tuned in the fully immersive

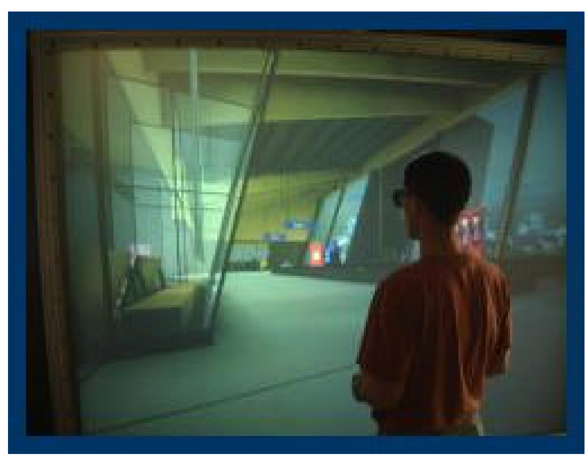

Fig. 5: The VISBOX is very similar to CAVE, but only has one display plane

environment of the CAVE. VISBOX runs on a single LINUX machine.

II.E. Simulators

CAVE simulators have been developed for PCs. These are available from NCSA and will provide the platform for routine development and testing of components of the virtual reality framework.

II.F. Future of 3D display technology

One must worry about the compatibility of the developed system with the technology of the future. Trying to predict the future of these technologies may be hazardous to one's credibility. However, certain trends can be followed. As is already happening with ImmersaDesk, the future CAVEs are also likely to be based on large sized monitors or tiled-monitors (an array of flat screen monitors) for each wall. NCSA's Visualization and Virtual Environments group has designed and installed a 20-tile scalable display wall. The *Wall* provides a display surface that is 4096 x 3840 pixels, useful for showing high-resolution imagery and animations, or for showing multiple information sources side-by-side. At 12 feet across and 9 feet high, the *Wall* easily accommodates small teams of people working together. [A significant challenge in building a tiled wall is to create a seamless image.] Such *Walls* may eventually replace the current generation of CRT projected CAVE walls.

3D display monitors have already been developed, and are likely to become economically feasible over the next several years. Tools being developed will rely on state of the art in 3D display technology, while conforming to the most widely implemented current standards, thus increasing the likelihood of the project's compatibility with the next generation 3D monitors currently under development. Though the details of operations of 3D monitors will not be known for some time, it is very likely that applications developed using 3D solid modeling packages, such as AutoCad and Pro-Engineer, will be either directly—or at least after passing through a "translator"—viewable on these next generation of monitors.

II.G. Streamlining the Develop-Integrate-Test Efforts

Even though several different choices for a graphics library are available, OpenGL [8], seems to be the most widely supported one across different computer platforms and graphics cards, in addition to being one of the most advanced sets of routines for rendering graphics. OpenGL provides ample control of detail, and the ability to re-use code. In terms of the language for the actual computer code, or the source code, the most widely used languages for OpenGL are C and C++, both of which have been standardized and remain very popular. Since C++ is a superset of C, and its object-oriented structure can be very useful in implementing the hierarchical nature of the program, this is an ideal choice for the language.

However, it is realized that not all development will start from scratch. Hence, from the development of 3D CAD models to displaying them in the virtual environment, we also plan to take advantage of other languages that are supported by CAD packages. Currently there are relatively few standards. Virtual Reality Modeling Language (VRML) [9] seems to provide a common ground to start. Unfortunately, unlike OpenGL, VRML remains a set of standards not implemented in hardware. Therefore, it cannot be used on its own. Moreover, efforts are also needed to develop a toolbox that can streamline the projection of all *vrml* files into the CAVE-like environments without significant input from the user. CAD packages (at least some of them) do allow the users to save their models in *vrml* format, and given the toolbox mentioned above, these models can then be easily ported to the CAVE.

Fortunately, different visualization tools discussed above are somewhat "compatible." Development carried out on one can be ported to others as necessary. However, there is a need to streamline the transfer of 3D CAD models that can be developed in software packages like

AutoCad, Pro-Engineer etc., into the 3D virtual environment.

A second item to be addressed is how to streamline the process of displaying the results of numerical simulations (like system variables such as temperature, neutron flux, heat generation rate, etc.) as part of the 3D CAD model. In this regard, past work carried out at NCSA toward the development of standards for file formats will be very useful. Specifically, we plan to rely on Hierarchical Data Format (HDF) project of NCSA [10]. It involves the development and support of software and file formats for scientific data management.

Additional issues related to representation of dynamic data in the virtual environment will be discussed elsewhere.

III. RECENT WORK

Work has progressed along several directions. Here we report the results of some recent developments.

III.A. Integrating the Standards

A program is being developed to provide a virtual display of data in a nuclear reactor's core. Such data can be provided by any outside source, such as another program or database, or an actual reactor's readings. The program will also provide a natural or instinctive way of changing various parameters; for example, moving the control rods by moving a lever. The basic method to achieve this goal includes creating a simulated control room, with customizable panels to hold all the instrumentation, and panel holders to hold the panels. In addition, future developments will include such details as a simulated view of the inside of a core, or a cooling tower, as a function of the core values passed on to the program. The code is being written in C++ and is based on OpenGL.

The program is divided into three major sections: the dynamically linked library (*dll*), the main program (*exe*), and the configuration data (*txt*). The purpose of the *dll* is to enumerate and evaluate a given function. Individual functions can be added to it for the purpose of reading a file, getting a value from a different program, or calculating a value from any given mathematical function. The *exe* draws what is actually displayed on the screen. It contains the functions to draw such items as a gauge, a panel with gauges, a structure containing panels, or an entire control room containing customizable components. Each of the components being drawn is defined in the *txt* by such parameters as dimensions, and the index of the function in the *dll* that should be used to evaluate, for example, the position of the meter needle. Also, each component that can be drawn by the *exe* includes one or more default pre-defined values for testing purposes. At this time, since the program is still being both designed and tested, *txt* files are not being used, and all of the parameters for the components to be drawn are hard-coded (a part of the code rather than user input) into the *exe* file. An anticipated method of editing the *txt* is by using a separate (GUI) application explaining all of the parameters that need to be supplied.

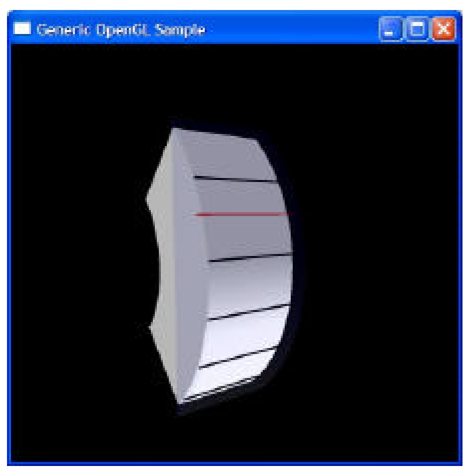

Fig. 6: A virtual cylindrical gauge

The following is a more detailed description of the program. Design of a basic component to interact with the *dll*—in this case, a cylindrical gauge—is described first (Fig. 6). For now, it only draws a customizable number of divisions. When completed, it will be possible to cover the face of the gauge by any picture selected by the user, and could include anything from several divisions, to labels, numbers, and scales. The gauge includes pre-defined values for such customizable items as the radius to the glass, or the needle, or the face. However, as of now the particular values of those parameters being used have been hard-coded, and for convenience defined as set A. The user would specify in the *txt* file that a cylindrical gauge A is needed at a certain position, with a given length, width, face, and index of the function to be used for evaluating the needle's position. The gauge is a sample component; others such as a knob, lever, CRT, etc. are being developed.

Next, a panel is created that holds different components. To test the link between the panel and individual components, several default sets of parameters were defined to respectively display 1, 2, 4, or 8 of the aforementioned gauges. A picture of the panel is shown in Fig. 7.

Naturally, the next component to be built is a place to install the panel. This could be a table (control panel) or a wall. These are created using the general category of "cylinder" (two parallel plane surfaces and walls connecting them). Pictures of a 3D control panel and a segment of a wall holding various instrumentations are shown in Figs. 8 and 9, respectively. An actual control room with instrumentation panels that can be arranged by the user, is shown in Fig. 10.

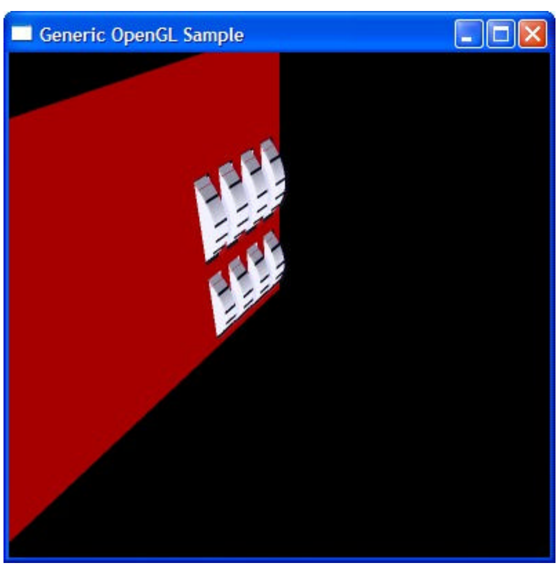

Fig. 7: The panel used to hold various instrumentation components being tested with 8 gauges.

The universal potential of the simple design of a cylinder with any user-defined polygon as the base is being exploited. For example, it is used to create a more universal code, and to draw a room—the ceiling and the floor are the base of the cylinder, and the sides of the cylinder are the walls of a room. The code, previously used to draw a single panel, is extended to draw any given side of a cylinder, and the routine to draw the control panel, as seen in Figs. 8 and 9, is extended to a more general code to draw any given cylinder.

The complexity of the program necessitates that the parameters used to draw the graphics (number of gauges, dimensions, etc.) be separated from the actual rendering code. The next major step is to create a platform-independent implementation of a user-friendly interface to change those parameters. This step will also help the development and implementation of the *txt*.

After the steps described above have been completed, we will focus on such details as choosing more artistic colors and textures rather

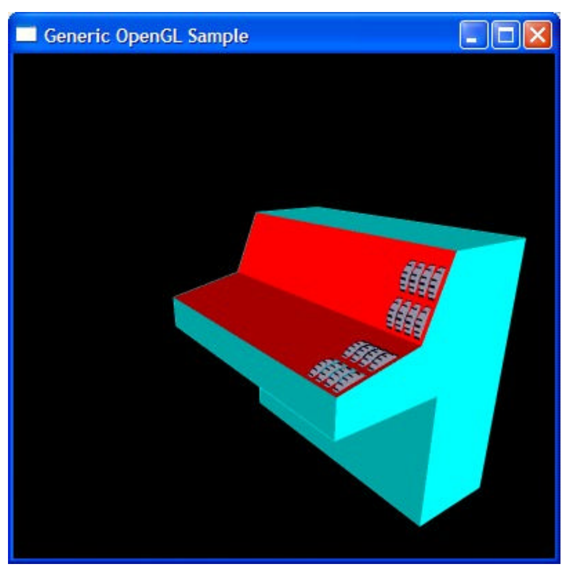

Fig. 8: A sample control panel with instrumentation.

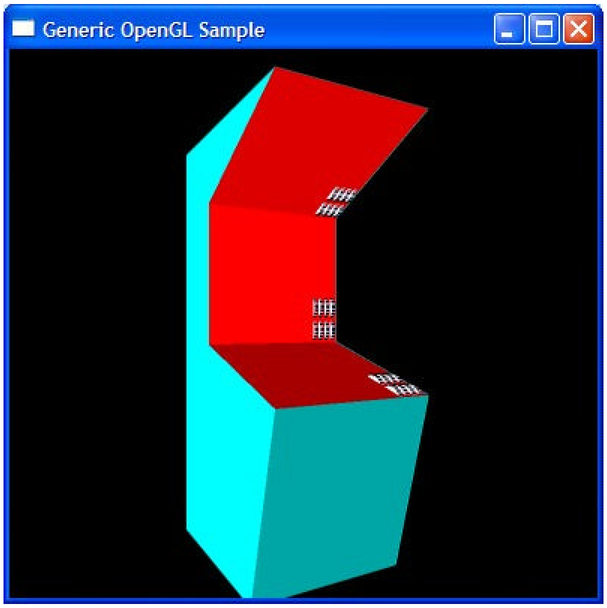

Fig. 9: Another sample control panel with instrumentation.

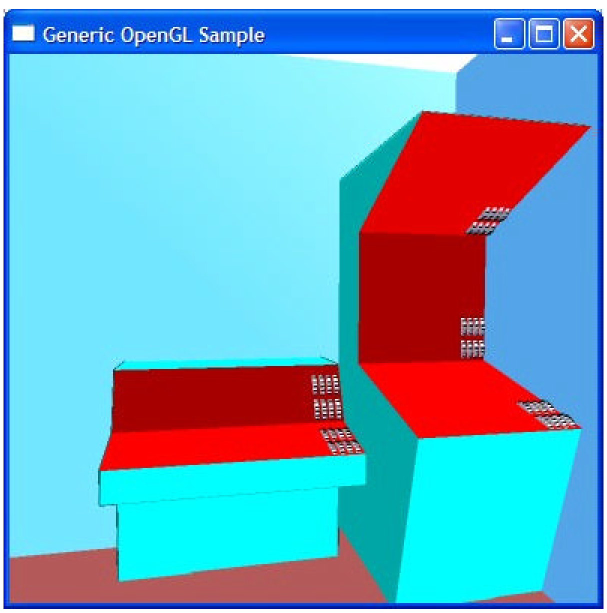

Fig. 10: A sample control room in early stages of development.

than using whatever will provide the best contrast between the components for more efficient debugging and development. The ability to use pre-selected images to emulate material textures, and to provide more control over what is displayed on gauges, will be incorporated.

In the CAVE, a fully developed virtual control room will be used to test the ergonomics of the setup as well as for operator training.

III.B. Displaying VRML files and data in the CAVE

Codes have been written that allow a user to display numerical data from 3D simulations in the CAVE. A generic capability has been developed that allows data written in a class of HDF-4 standard files to be easily displayed in the CAVE. At this time only limited manipulation (rotation, translation, cutting planes, etc) has been incorporated. In addition, another conversion routine has been written that allows *vrml* files to be displayed in the CAVE (Fig. 11). Models created in CAD software can be saved in the *vrml* format, and then easily displayed in a virtual environment after passing through the above described code.

IV. CONCLUSIONS

Technology, safety, and efficiency of nuclear reactors have significantly evolved since the initial discovery and subsequent development of the nuclear power technology. However, safety still remains one of the major concerns, and one of the best ways to improve this aspect of the nuclear industry is by improving the ergonomics and human factors. One of the essential steps in this direction is to be able to model as many aspects of a nuclear power plant as possible, with maximum flexibility, and minimal costs. Fortunately, computer technology has advanced enough that most modeling and simulation tasks of nuclear power plants can be carried out on an average personal computer. This fact has been exploited to develop various codes to simulate core operation. However, recent advances in the field of graphics processing for personal computing have not yet been utilized to their full potential. The goal of this on going project is to assemble currently available information and standards into a code which can be used for virtual reality simulation of a reactor site to enhance training of reactor operators, or the overall power plant ergonomics.

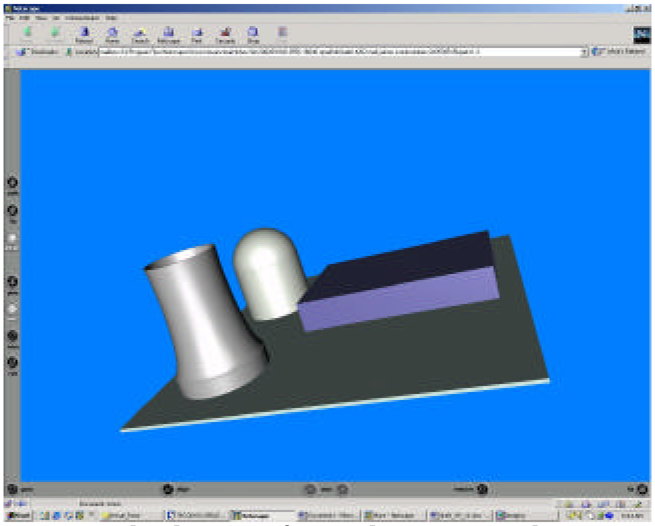

Fig. 11: The layout of a nuclear power plant developed using a standard 3D CAD model development package.

ACKNOWLEDGEMENT

Some of the pictures are taken from web sites of NCSA at UIUC and EVL at UIC. Research is funded in part by a grant from the Department of Energy.

REFERENCES

1. W. SHERMAN and A. CRAIG. *Understanding Virtual Reality*. Morgan and Kaufmann (2002).

2. Iowa State University's Virtual Reality Application Center's website http://www.vrac.iastate.edu/

3. Argonne National Laboratory web site: http://www.re.anl.gov:80/vr/virtualreality.html

4. E. SCHMITZBERGER and J.L. BOUCHET. "A Test Case of Computer Aided Motion Planning for Nuclear Maintenance Operation." *ICONE* (2001).

5. The Avant-Garde Simulation Technologies in Korea Electric Power Research Institute, Simulator Team, http://www.kepri.re.kr.

6. National Center of Supercomputing Application (NCSA) web site: www.archive.ncsa.uiuc.edu/VR/VR and http://cave.ncsa.uiuc.edu/

7. Electronic Visualization Laboratory, University of Illinois at Chicago (UIC), website: http://www.evl.uic.edu/home.html

8. OpenGL official website: http://www.opengl.org

9. VRML repository, maintained by the Web3D consortium: http://www.web3d.org/vrml/vrml.htm.

10. National Center of Supercomputing Application (NCSA); HDF group: http://hdf.ncsa.uiuc.edu/

2003 International Congress on Advances in Nuclear Power Plants

May 4-7, 2003 • Congress Palais • Córdoba, SPAIN

Proceedings of ICAPP '03

Table of Contents

Author Index

Forward

Sponsors

Organizers

Search this CD-ROM

CD-Rom Help

Exit