Int. J. Engng Ed. Vol. 14, No. 5, pp. 316–321, 1998
Printed in Great Britain.

0949-149X/91 $3.00+0.00

World Wide Web-based Simulation*

TAMIE L. VEITH
Biological Systems Engineering Department, Virginia Tech, Blacksburg, VA 24061, USA
JOHN E. KOBZA, C. PATRICK KOELLING
Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061, USA.
E-mail: tveith@vt.edu, jkobza@vt.edu, koelling@vt.edu

World Wide Web (WWW) simulation is a new engineering tool with enormous potential to impact engineering education. By combining the modeling capability of traditional simulation tools with the communication capability of the Internet, WWW-based simulation provides wide accessibility, controlled access, efficient maintenance and increased integration. This paper describes WWW-based simulation, reviews its development and educational applications, and summarizes the current state of the art in this rapidly evolving area.

INTRODUCTION

SIMULATION is an essential engineering tool used by both students and practitioners to gain insight into a system's behavior. Simulation tools allow students to model a system, understand it, and then explore alternate designs. The disadvantages of traditional simulation software, including its expense, lack of portability and long learning curve, can limit its use in courses that do not have simulation as a major component. World Wide Web (WWW) simulation has the potential to overcome these limitations by combining the modeling power of traditional simulation languages with the ease of communication provided by the Internet. Benefits of WWW-based simulation include wide accessibility, controlled access, efficient maintenance and increased integration.

The key to this synergistic combination is the object-oriented structure of the Java™ WWW-based programming language. Object-oriented programming is particularly well suited for simulation modeling [1]. An object-oriented program consists of a number of *classes*, each one a module defining some aspect of the program's process. Activating a particular class creates an *object* of that class. The class acts as a template, with its attributes and behaviors defining the object. As the program runs, the objects are created and destroyed on an as-needed basis and interact by passing parameters back and forth. Object-oriented programming languages reduce the need to pass multiple parameters sequentially through a program as is often necessary in procedural languages such as Pascal or FORTRAN. Additionally, by using separate modules to define unique attributes, object-oriented programming allows easy modification and code reuse. This results in efficient, cost-effective model creation and maintenance [2, 3].

Java closely resembles the C and C++ object-oriented programming languages with a few alterations, such as the removal of pointers, specifically intended to make Java more robust. Java further simplifies use for the end user, as well as the programmer, by supporting graphical user interfaces and a number of menu formats. The result is a simple, efficient, platform-independent programming language that is ideal for creating WWW-based programs. Using Java one can create small programs called applets that are embedded into an HTML document and viewable on any Java-compatible browser. The applets are compiled into a set of bytecodes, or machine-independent processing instructions. An interpreter within a Java-compatible browser translates this set of bytecodes, and executes the applet [4]. Java's high portability enables an applet to be accessed through numerous platforms on almost any computer connected to the Internet.

WWW-based simulation packages, particularly when written in Java, provide several beneficial features that can be grouped as follows:

Wide availability
- allows access on many platforms without recompiling;
- allows access at distant sites without transporting hardware or software;
- allows access outside normal business hours.

Controlled access
- protects against unexpected change and duplication of original model;
- allows 'copy exactly' model distribution;
- enables individualized access through passwords on unrestricted machines;
- allows limited time-span access.

* Accepted 15 September 1998.

Efficient maintenance
- enables frequent modifications to be made and instantly distributed;
- reduces error potential when updating and distributing models;
- eliminates virtually all on-site maintenance;
- allows modifications and implementations to be made through the server.

Increased integration
- interfaces instantly with existing WWW browsers;
- requires only a WWW browser capable of viewing Java programs;
- encourages communication and interaction through the WWW.

The following sections present these characteristics in the context of three examples that go beyond the basic queuing models currently provided by WWW-based simulation packages to explore future possibilities of WWW-based simulation. The different approaches, both in and out of the classroom, that have been taken during the development of WWW-based simulation are then discussed. Finally, a summary is given of the current state of WWW-based simulation and a potential view of its future.

EXAMPLES AND BENEFITS

This section presents three examples as a reference for discussing the benefits of WWW-based simulation. The first example involves a simulation model used in an engineering problem solving class. The second is a marketing situation where a simulation model is used to demonstrate the advantage of new products to customers. The third involves the use of simulation models by businesses.

Engineering education

As part of an engineering problem-solving class, a student is given a simulation model of a manufacturing line that consists of several machines in sequence. The cost of holding partially completed items, revenue associated with completed items, and the relationship between production rates and per-unit production costs are embedded in the simulation model. The student specifies the production rates at the different machines and runs the simulation. The model shows items moving through the manufacturing line, the queues of items at the different machines and the customers picking up completed items at the end of the line. After execution, the simulation computes an average per-unit profit or loss for the manufacturing line. Through repeated interaction with the simulation the student identifies the production rates that maximize the long-run profits.

This simulation model could provide a valuable educational experience, but there are several issues potentially preventing its use by large numbers of students. First, the computer used by the student may need to have the simulation language installed in order to run the model. If this is the only simulation to be used by the student, there is little advantage in buying and installing the software (even for a relatively inexpensive student version). Supposing the student could be provided with an executable version of the simulation, this executable file might not be compatible with all the computers used by the students in the course. Providing the simulation at a specific location, e.g., laboratory computers, limits its accessibility and availability. Interacting with the simulation may also require the student to learn the simulation language, which may not be relevant to the course's learning objectives. If the software is updated to change the underlying costs, how are these updates distributed and how do you ensure all students use the latest version?

These difficulties exist not only within a single course, but also across the engineering educational community. For example, one university may provide accessibility by obtaining site licenses for software or requiring students to purchase a standard computer configuration, but these actions may exclude students at other universities from using the model. WWW-based simulation can help to address these issues by providing wide accessibility, controlled access, efficient maintenance and increased integration.

Wide availability. The portability of a WWW-based model onto numerous platforms enables the user to run the model from multiple locations without requiring additional hardware, software, file downloads, or code recompilation. Furthermore, since the Internet is usually available twenty-four hours a day, access to a model is not limited by time constraints. This provides economical widespread availability of an educational tool. With WWW-based simulation, a student can use any computer having a Java-capable WWW browser to access the website for the course and run the Java applet containing the simulation. The simulation interface appears the same as a typical website and requires no special training to use.

Controlled access. Since Java does not allow users to save applets, a person must have access to the original model or package source code to permanently change a Java-based simulation model or duplicate the simulation package. This ensures users access the latest version of the model and reduces the potential for inadvertent or unauthorized alterations. This benefit can be particularly useful for teachers asking students to individually modify and analyze a particular model. If desired, access to the site can be controlled using passwords or network restrictions. Such access restrictions are placed on the model or the WWW site, not on the machine itself, ensuring uniform access privileges regardless of platform. An educator might use this option to

restrict the access of copyrighted material to class members.

Efficient maintenance. Via the Internet, the creator/maintainer of a WWW-based simulation model can permanently modify the model's source code by accessing the serving computer from almost any WWW-linked location. The model can then be made instantly available over the WWW and users notified, if necessary, through e-mail. This allows a teacher or teaching assistant to maintain the site from his or her office or while away on business. After maintenance, the updated model is instantly available, simplifying version control and minimizing learning disruptions.

Increased integration. Programming software and Java-compatible browsers that support WWW-based simulation packages are available through the Internet. Many such viewing browsers are currently free to educators, students, and researchers or are included in standard office software packages. Once the browsers are installed, no additional software or downtime is required before accessing a simulation model.

Since many users are familiar with navigating WWW browsers, it is easier to learn the simulation software. Various WWW-based tools enable a user, while working with a model, to send messages, answer questions, explore a subject area, and research a particular topic in depth. For example, each WWW-based simulation model is linked to a hypertext markup language (HTML) page. This page can contain e-mail links, help or problem request forms, connections to relevant search engines, and links to previous models. The HTML page can also provide supplemental information in the form of textual instructions, images, sound clips, and other appropriate multimedia tools. Internet tools such as list-serves, electronic bulletin boards, or chat rooms can further enhance user communication.

An educator can provide both technical and course-related information along with the simulation model. Technical assistance might include HTML instructions on how to read and use the specific model, appropriate parameter ranges to employ, and a list-serve dedicated to usage questions and difficulties. These tools would allow more class time to be spent on the objectives of the exercise, promptly notify all users of any technical issues that may arise, and help prevent users from focusing on the technicalities of the tools rather than learning the course concepts. The WWW site might include homework assignments related to the simulation project in the format of forms to be completed by each student and sent to the instructor via the WWW. Advanced websites might include video clips from the actual manufacturing line being modeled to help the student relate the coursework to the real world. Links for references and related WWW sites could be used to encourage the students to explore simulation in connection with this and other engineering problems, while an e-mail link to the instructor would allow the student to propose new ideas resulting from such exploration and interact frequently with the instructor.

Product sales

Immediate demonstrations of the benefits of new products can be vital to the growth of competing companies, whether suppliers or consumers. Customer feedback regarding a supplying company's products can help the marketer focus its efforts. Likewise, promptly accessible, well-explained product previews help a consumer company function and expand smoothly.

Demonstrating, via a WWW-based simulation model, the ways a new product can enhance the customer's system is effective for both the marketing company and the customer and encourages suppliers to market new products in a cost-effective, consumer-aware manner. For example, a production-line machine supplier might use WWW-based simulation modeling to demonstrate a more efficient machine. By providing two simulation models, one based on the current machine and one on the new product, and by allowing customers to input model parameters for their particular production runs, suppliers can demonstrate the benefits of the new products.

By using WWW-based simulation, the company can make a single copy of the models available over the WWW instead of mailing demonstration software to potential customers. This reduces costs and avoids customer frustration with installation and potential hardware compatibility problems. The company's marketers are able to reach more people, especially individuals and small businesses who might otherwise be overlooked.

E-mail and other relevant WWW links provided at the marketing site along with the simulations could help customers learn about the company, locate additional products, and direct questions to the most helpful source. Likewise, survey forms and search engines built into the preview help the marketer find out more about the needs and concerns of their customers through customer feedback, orders, and complaints and can allow quicker, individualized responses. Combining these types of WWW-based tools with simulation models describing the product provides an interactive connection between the marketer and both potential and existing customers.

Business performance

Today's fast-paced, corporate world challenges businesses trying to stay ahead of competitors and of the market. The key to success for organizations is giving decision makers access to the latest information. WWW-based simulation can do this by providing effective economical tools.

Consider a mutual fund company that develops a stock market econometric model to forecast

interest rates. Fund managers could use the model to predict portfolio performance, determine buy/sell actions and advise investors. The benefits of using a WWW-based model include a single copy of the model that can be updated continuously as economic indicators become available. Since the model is proprietary, the company would limit access to the site by using passwords or network restrictions. Fund managers can access the current version twenty-four hours a day from almost any computer (subject to possible network restrictions). Output from the model can then be used as input to portfolio management software that can identify weakness of the current portfolio and identify opportunities for improvement.

APPROACHES

In the 1980s Jack Thorpe, in conjunction with the United States Department of Defense, created SIMulator NETworking (SIMNET), a networked system of computers running a single simulation program [5]. SIMNET was an attempt to make the use of simulators and simulation techniques more feasible for military defense operations. In this system each user computer acts as a simulator, allowing multiple users to experience a single simulation program simultaneously in a safe, cost-effective environment.

The success of SIMNET resulted in a standardized simulation networking protocol, Distributed Interactive Simulation (DIS). DIS allows computerized simulators to communicate and act in synchronicity through the Internet. Although DIS began in the military environment, it is now being used increasingly often in non-military applications such as air traffic control, intelligent vehicle highway systems, and interactive, multi-user computer games. Work is underway to incorporate some aspects of Internet markup and programming languages such as Virtual Reality Markup Language (VRML) and Java to the DIS protocol. This will increase the abilities of DIS systems to define behavioral information and visual representations as users enter and exit the simulation environment.

The WWW and its existing structure of interconnected hypermedia provide a relatively high-level, user-friendly interface for the Internet. This has prompted the growth of WWW-based simulation, combining non-Internet and the above-mentioned Internet simulation techniques with current WWW-based technologies. The following researchers have provided key contributions in the area of WWW-based simulation.

Irene Neilson discusses the use of the Interact Simulation Environment (ISE), created as an aid to teaching engineering students [6]. The ISE embeds simulation models into a graphical user interface while specific commands are placed into related HTML documents. Because of differences in students' background computer knowledge, the ISE is set up to utilize simulation as a modeling tool without requiring the student to interact with the actual simulation interface. A student may save 'snapshots' of the simulation model at a given state for later reference or to use in discussion with others. However, the ISE, as currently developed, requires the original simulation model to be stored directly on the user's machine in order to access 'snapshots' of that model. Using Java to develop the ISE would allow platform-independent model interpretation and eliminate the need for multiple storage of the simulation, as well as the resulting increase in file maintenance.

Rodney Cole and Scott Tooker [7, 8] have developed WWW-based tutorials to assist physics students. Making these simulation models available over the WWW greatly expands the range of possible access locations. Like ISE, the physics tutorials allow students to see interesting cases of a given simulation model without requiring prior knowledge of the parameters defining these cases or of the background programming languages involved. The use of familiar WWW browsers such as Netscape virtually eliminates the amount of time necessary for distributing and learning viewing software.

The tutorials use Apple's OpenDoc Frameworks to provide a basic simulation environment. OpenDoc supports modular development of models, allowing existing tutorials to be tailored to meet the needs of different educational grade levels. Cole and Tooker state that future tutorials will combine OpenDoc components, HTML, and multimedia effects created using Java. OpenDoc and Java are complementary tools; OpenDoc components can be written entirely in the Java language, or Java applets can be embedded into the components. However, using OpenDoc in this method voids the platform-independence of Java. OpenDoc applications and OpenDoc components, such as those created by OpenDoc Frameworks, must be downloaded onto the user machine with a working copy of OpenDoc. OpenDoc is currently available only for the Macintosh platform.

The importance of the WWW as a platform for interactive learning is supported by T. Singh [9]. In particular, he discusses the WWW in connection with the added cross-platform benefits of Java. Singh demonstrates Java applets for teaching electromagnetics and theories of dipole-antenna. These tutorial applets are embedded in instructive HTML pages and allow some degree of user interaction with the presented models.

Using simulation models based on the process interaction paradigm, Rajesh Nair demonstrates WWW-based simulation through a unique combination of Java applets and query-driven databases [10]. Actions by the user send queries to a database that then initialize the appropriate applet. The database stores data created by the running simulation and supplies these data back to the applet as necessary throughout the simulation run. Using a database in this manner helps

circumvent some security issues currently imposed by Java applets. Nair's simulation model applets incorporate JSIM, a library of Java classes specifically created for this simulation project. The user interfaces provided for the example applets are simple and self-explanatory. However, animation is interrupted periodically as the applet pauses to connect with the database on the server. Simple, descriptive statistics are available for the simulation run after the user chooses to exit the simulation. Unfortunately, the window displaying the model closes before the statistics appear, so the user is unable to refer back to the model.

SimKit, created by Arnold Buss and Kirk Stork [11], runs as an applet on the WWW and takes advantage of Java being object-oriented by using the event graph design approach. SimKit models discrete-event simulations and is particularly geared toward military applications. The main simulation facilitating package, Javasim, is designed to allow for expansion in order to accommodate frameworks for various types of simulations. Javasim makes extensive use of Java interfaces, which add defined behaviors to identified classes without imposing the hierarchical structure of class inheritance. This structure allows a modeler to add customized tools into SimKit without making changes to Javasim. SimKit permits user interaction through a detailed, model entry form. By combining with a Java-based graphing package designed by Leigh Brookshaw, SimKit allows a useful output of statistics and graphs.

Much of the preceding research provides only limited, WWW-based simulation capabilities, motivating the development of Netsim, a general, discrete-event, WWW-simulation package currently available through the WWW at http://hokies.ise.vt.edu/netsim/ [12]. Netsim was created as a Masters Thesis project in an attempt to promote simulation as an advantageous and available educational tool for today's computer-assisted, standard and distance-learning classrooms.

Programmed entirely in the Java language, Netsim offers all the previously discussed benefits of WWW-based simulation while overcoming the limitations presented in the current WWW-based simulation literature. For example, a model created in Netsim appears as an applet on an HTML page and is available over the WWW, taking full advantage of the portability of Java. No special software or code is needed on the user's machine.

Netsim implements discrete event simulations using event graphs [13]. This allows users full interactive modeling capabilities with a large amount of flexibility in defining and customizing models. A programming interface provides a blank template with text fields for the various parts of a simulation model, such as event name and state variables. Any type of basic simulation model may be entered into this interface, using an event graph. While knowledge of simulation, particularly the event graph approach, is useful in designing a model in Netsim, no knowledge of Java or simulation modeling is required to enter or modify the model.

A second interface allows the user to run the simulation model. This interface not only provides start, pause, and stop capabilities and data output, but also an animation of the model. Model animation allows viewers a visual demonstration of how the system is operating over time. This is an important feature of Netsim in that users of all educational backgrounds can use this WWW-based simulation tool to visually understand the operation of a system or process.

CONCLUSIONS

Simulation has an important role in engineering education. It provides a means to understand and make predictions of real-world systems while allowing numerous simplifications to the process under observation. Simulation aids in engineering problem solving and system design, particularly in cases where it is impractical or unsafe to manipulate the real system until the optimal solution, is achieved.

With WWW-based simulation, educators are able to incorporate simulation models into their courses without the cost and inconvenience of requiring special equipment or materials and without spending significant time teaching students to program and manipulate the model. Outside the classroom, as well, WWW-based simulation offers valuable cost savings, efficiency, and computer system flexibility to independent and corporate business professionals.

By accessing simulation models through the WWW, students and practitioners alike are able to ensure a common reference and to communicate with each other, building on understandings and ideas. Obtaining and using WWW-based simulation can become as simple as using a favorite WWW browser, with problem-specific help and technical advice a mouse-click away. As technology continues to advance, WWW-based simulation is likely to become much more widespread making simulation, itself, a common communication tool.

REFERENCES

1. J. Banks, J. S. Carson II and B. L. Nelson, *Discrete-Event System Simulation*, 2nd ed., Prentice-Hall, NJ (1996).
2. A. Freeman and D. Ince, *Active Java: Object-Oriented Programming for the World Wide Web*, Addison-Wesley, Harlow, England (1996).
3. J. A. Joines and S. D. Roberts, Design of object-oriented simulation in C++, *Proc. 1996 Winter Simulation Conference*, ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, Coronado, CA (1996) pp. 65–72.
4. L. Lemay and C. L. Perkins, *Teach Yourself Java in 21 days*, Sams.net, Indianapolis, IN (1996).
5. D. Fullford, Distributed interactive simulation: it's past, present and future, *Proc. 1996 Winter Simulation Conference*, ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, Coronado CA (1996) pp. 179–185.
6. I. Neilson, R. Thomas, C. Smeaton, A. Slater, and G. Chand, Education 2000: implications of W3 technology, *Computers and Education*, **26**, (1996) pp. 113–122.
7. R. Cole and S. Tooker, Physics to go: Web-based Tutorials for CoLoS physics simulations, *Technology-Based Re-Engineering, Engineering Education, Proc. Frontiers in Education FIE'96 26th Annual Conference*, Salt Lake City, UT (1996) pp. 681–683.
8. S. Tooker and R. Cole, Using OpenDoc to create low-cost physics simulation tools for secondary and higher education, *Technology-Based Re-Engineering, Engineering Education, Proc. Frontiers in Education FIE'96 26th Annual Conference*, Salt Lake City, UT (1996) pp. 688–689.
9. T. Singh, M. Zhu, U. Thakker and U. Ravaioli, Impact of World Wide Web, Java and virtual environments on education in computational science and engineering. *Technology-Based Re-Engineering, Engineering Education, Proc. Frontiers in Education FIE'96 26th Annual Conference*, Salt Lake City, UT (1996) pp. 1007–1010.
10. R. S. Nair, J. A. Miller and Z. Zhang, Java-based query driven simulation environment, *Proc. 1996 Winter Simulation Conference*, ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, Coronado, CA (1996) pp. 786-793. http://www.cs.uga.edu/~rajesh/JSIM/jsimdistribution/docs/API_users_guide.html.
11. A. H. Buss and K. A. Stork, Discrete event simulation on the World Wide Web using Java, *Proc. 1996 Winter Simulation Conference*, ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, Coronado, CA (December 1996) pp. 780–785.
12. T. L. Veith, Netsim: a Java-based, WWW simulation package, Master's thesis, Virginia Tech, VA (1997).
13. L. W. Schruben, *Graphical Simulation Modeling and Analysis: Using SIGMA for Windows*, Boyd & Fraser, Danvers, MA (1995).

Tamie Veith is a USDA National Needs doctoral fellow in the Biological Systems Engineering Department at Virginia Tech. She received her BA in mathematics from Reed College and her MS in industrial and systems engineering at Virginia Tech. Her research interests include theoretical and biological systems applications of operations research with particular interest in computer modeling and Internet applications.

John Kobza is an assistant professor of industrial and systems engineering at Virginia Tech. He received a BS degree in electrical engineering from Washington State University in 1982, and a MS degree in electrical engineering from Clemson University in 1984. From 1984 to 1987 he was a Member of the Technical Staff at GTE Laboratories. He received a Ph.D. degree in Industrial and Systems Engineering at Virginia Tech in 1993. His research interests include system performance modeling and analysis. He is a member of INFORMS, IIE, IEEE, Sigma Xi, and Alpha Pi Mu.

C. Patrick Koelling is an associate professor of industrial and systems engineering at Virginia Tech. He earned BSIE, MSIE, and MBA degrees from the University of Missouri, and a Ph.D. from Arizona State University. He is a senior member of the Institute of Industrial Engineers, and a member of the ASEE, ASQ and INFORMS. He is also a member of Sigma Xi, Tau Beta Pi, Alpha Pi Mu, and Beta Gamma Sigma honor societies. Dr. Koelling's areas of expertise are in operations research and computer applications.