PANEL SESSION:
THE MODELING METHODOLOGICAL IMPACTS OF WEB-BASED SIMULATION

Moderator:
Ernest H. Page, The MITRE Corporation

Panelists:
Arnold Buss, Naval Postgraduate School
Paul A. Fishwick, University of Florida
Kevin J. Healy, Thread Technologies
Richard E. Nance, Virginia Tech
Ray J. Paul, Brunel University

KEYWORDS

Methodology, multimodels, environments, verification and validation.

ABSTRACT

We consider the current and future impact of the World Wide Web on the fundamental nature of simulation conceptualization, design and use.

1 THE QUESTION

Simulation modeling methodology deals with creation and manipulation of models over the lifetime of their use. Motivated by the recognition that the manner in which a simulation model is conceived, developed and used can have a significant impact on the ability of the model to achieve its objectives, modeling methodology has been an active research area since the inception of digital computer simulation.

Over the past forty years the practice of simulation model creation has evolved from coding in general-purpose languages, to model development in special-purpose simulation languages, to model design using higher-level simulation model specification languages and formalisms, to comprehensive theories of simulation modeling and holistic environment support for the modeling task. Thematic in much of the modeling methodological work to date has been the recognition of Dijkstra's principle of the "separation of concerns" which argues for the separateness of specification and implementation (Dijkstra 1972). In many cases, this philosophy has been tempered by the pragmatic observations of Swartout and Balzer (1982), who observe that separation is a worthy goal but not achievable in totality since any specification, S, may be viewed as an implementation of some higher-order specification, S'.

Another argument in favor of the intertwining of specification and implementation is that technological advancements may enable new approaches to accomplishing a task – approaches that were not even conceivable prior to the advent of the technology. Consider, for example, the advent of the assembly line in manufacturing. The potential of the WWW assuch a technology push is cited in a recent article that describes the application of the WWW within the manufacturing process (Erkes et al. 1996), "Our initial experiments at putting engineering, design and manufacturing services on the Web are so successful that we believe we should rethink the traditional approaches and tools for coordinating large, distributed teams." With respect to simulation, a similar revolution seems plausible. Web technology has the potential to significantly alter the ways in which simulation models are developed (collaboratively, by composition), documented (dynamically, using multimedia), analyzed (open, widespread investigation) and executed (using *massive* distribution).

Is the web such an elixir, demanding that we radically alter our modeling philosophies and approaches? Or is web-based execution merely another implementation detail that can, and should, be abstracted from the model development process?

2 WEB-BASED SIMULATION MODELING (Arnold Buss)

The explosion of computer networks have created an environment for computer modeling in general, and simulation modeling in particular, that is revolutionary. In order to properly exploit these developments the nature of modeling must change.

Simulation models have been traditionally monolithic in design. The advent of Object-Oriented Programming has resulted in more elegantly designed monoliths. Simulation models for both industrial and military applications have been mostly designed for models running on a single machine. For such models the network offers little. Using the full power of the network offers potentially substantial benefits to modeling and simulation, but only if models are designed differently.

Use of up-to-date data by dynamically interacting with databases across the network. This speeds up the modeling and decision-making cycle by an order of magnitude. The integration of computer models running with systems has great potential for military analysis and training.

What Web-Based Simulation Models Must Do:

- Write applications that expect to receive data across the network from a database that will be dynamically determined.
- Write applications that will expect to receive new classes and data unforeseen at the time the model was started.
- Write applications using component that are loosely bound, rather than tightly coupled.

The Java programming language, together with the related cluster of Java Technologies, have substantially extended the capabilities of program-level tasks. Java classes can open sockets across a network, perform database queries, and encrypt data streams for secure transmission. New classes may be dynamically incorporated as the program is running, thus enabling dynamic extensibility. Objects on one computer may be serialized and sent to another, where thay are immediately incorporated into that computer's model. Objects on another computer may be invoked through Remote Method Invocation.

The capabilities of programming languages have outstripped our knowledge of how best to write programs exploiting these capabilities. Software design principles for procedural and even object-oriented programs are well-known. It is not yet known how software should be designed using these tools. It is also not clear how best to exploit the tremendous possibilities offered by the Network.

3 DISTRIBUTED MODELING USING THE WEB AS AN INFRASTRUCTURE (Paul A. Fishwick)

Ernie Page (Chair of Panel) has written a good introduction to the problems and possibilities with regard to simulation and the World Wide Web. We are in the infancy of the potential of the web to simulation (Fishwick 1996) and we should try to determine the future effects of the web on simulation research. Will it have an impact, and what is the nature of this impact? Like most technologies in computer science, the web will most likely have a profound effect on simulation since the web represents a new way of both publishing and delivering multimedia information to the world. Models are a form of information. I would like to focus on one of the effects: distributed modeling. We are familiar with distributed execution of models using conservative, optimistic and hybrid approaches but we have not sufficiently explored the possibilities with regard to distributing our model specifications over the net as well as our execution of models. There are no standards for generally-defined dynamic system representations, although some domain-specific standards exist, such as VHDL. We do have a specification for the shape and geometry of objects in the form of VRML (Virtual Reality Markup Language), but for actions, there is a fairly large hole representing an unexplored region.

If we are to truly share models and digital objects (virtual geometric and dynamic representations of physical objects), then we need to derive a standard such as VRML, but for dynamics. Many simulation researchers have excellent approaches to modeling that can serve to help us as a technical community to build such a standard. At the University of Florida, we have constructed a language that we call DMML (Dynamic Modeling Markup Language). DMML allows multimodels (Fishwick 1995) to be specified for a system. These multimodels do not guarantee that levels of abstraction are behavior-preserving but they do create a step in that direction by allowing the specification and integration of different model types within the overall model. DMML is part of the MOOSE System (Cupert and Fishwick 1997), which is a software prototype for Windows 95, NT and Solaris Unix.

The idea of having a standard for dynamics is that a manufacturer of a physical product will have, on their home page, a complete digital representation of the product that can be found through web search engines, and executed remotely. Through this distributed modeling approach, the web will become a vehicle for simulation model reuse in a way that has not been previously possible.

4 SIMULATION MODELING METHODOLOGY AND THE WWW (Kevin J. Healy)

The World Wide Web was conceived as a set of simple Internet-based client/server protocols for transferring and rendering documents of a primarily textual nature. What distinguished the Web's mode of communicating information from other Internet-based tools that preceded it (e.g. electronic mail, electronic file transfer via ftp, and network newsgroups) was the provision for embedding hyperlinks that allowed users to easily navigate between related documents. The hyperlinking scheme allowed content providers to organize and present information in a natural hierarchical fashion. It also served to insulate users from the tedious details involved in identifying and retrieving a particular document. Since the development and rapid widespread adoption of these conventions, they have been extended and integrated with other new related technologies that provide for the delivery of content that is much more dynamic in nature. The most important of these related developments has been the introduction and rapid widespread adoption of the Java programming language as a standard for Internet-based computation.

The integration of the Web and Java represents a technological advancement that enables a fundamentally new approach to simulation modeling, one that makes possible the development of environments with coherent Web-based support for collaborative model development, dynamic multimedia-based documentation, as well as open widespread execution and investigative analysis of models. A key aspect to the approach is the role the Java language plays in both the specification and implementation of the model.

The evolution to high-level model specification languages and formalisms has been motivated by the desire to make simulation more accessible by eliminating the programming burden. However, such systems are often difficult to modify or extend because of an imposed separation between the specification system and its implementation. This can lead to models that poorly mirror system behavior and have no potential for distribution and reuse within an enterprise. The Java language is ideally suited to implementing an advanced simulation architecture whose features are readily accessible at the programming language level, special purpose simulation language level, and high-level model specifications.

Specifically, key features like the well-designed object-oriented nature of Java and native support for multithreaded execution allow special purpose simulation modeling features to be incorporated directly into the Java language in a natural way so that the underlying modeling and programming languages are the same. These relatively low-level but powerful modeling capabilities can in-turn be used to implement higher-level model specification systems via the JavaBeans component development model. The simple programming conventions that constitute JavaBeans allow Java-based software components to be assembled visually into applications using any of a growing number of sophisticated graphical programming environments including Symantec's Visual Café, Microsoft's J++, IBM's Visual Age, Sun's Java Workshop, Borland's Jbuilder, and Lotus's BeanMachine. When visually assembling predefined simulation modeling components, no programming is required; however, when necessary, the user has access to the underlying code and full power and flexibility of the Java programming language. What's more, any Java environment can be used for model building and debugging. The modeling language capabilities and predefined component assembly capabilities can also be used in isolation or in combination to produce high-level standalone simulation applications that users interact with in predefined ways.

The hardware and operating system independent design of Java facilitates collaboration by allowing modelers to share language level or component level models independent of where they were developed. The documentation and deployment of modeling tools and end-user applications via the Web also serves to make open and widespread both the development and investigative analysis of models.

This vision of Web-based simulation is the motivation behind Thread Technologies' design of *Silk*™, a general purpose simulation language based on the Java programming language. *Silk* merges familiar process-oriented modeling structures with powerful object-oriented language features in an intelligent design that encourages model simplicity and reusability through the development and the visual assembly of *Silk* modeling components in JavaBeans-based graphical software environments. More generally, *Silk*'s openly extensible, scalable, and platform independent design represents the type of approach that is essential to keeping simulation modeling on track with other revolutionary changes taking place in Internet-based computing.

5 SIMULATION MODELING METHODOLOGY IN THE WONDERFULLY WEBBED WORLD (Richard E. Nance)

While modeling methodology has been with us since the inception of simulation, it remained indistinguishable from programming throughout the first two decades. Neverthelss, a few early researchers abstracted beyond the executable form to search for more significant semantic revelations. Lackner and Kribs (1964) and Kiviat (1967) are prominent examples, but Tocher's (1966) wheel charts to assist in model specification and the IFIP proceedings on simulation programming languages (Buxton 1968) show that interest was widespread. Efforts to derive a theory of simulation (Zeigler 1976) generated interest in model representation in the 1970s. The latter part of the decade ushered in the first specific focus on modeling methodology (model life cycle, model specification languages, the DELTA project) (Nance 1979). With the 1980s came the vision of model development environments (Nance 1983) that are now a commercial reality. Is the subject of this panel session presaging the next major transition in simulation model development?

5.1 Modeling Methodology

Since "methodology" is both over-used and misused, a definitional explanation in this context is appropriate. Methodology, following the view of Arthur et al. (1986, p.4), should:

- organize and structure the tasks comprising the effort to achieve global objectives,
- include methods and techniques for accomplishing individual tasks (within the framework of global objectives), and
- prescribe an in which certain classes of decisions are made and the ways of making those decisions leading to desired objectives.

Key in the attainment of the objectives are the *principles* that form the foundational support of a methodology.

5.2 Influence of the Web

If the world wide web is to effect major changes in modeling methodology, then it must alter or abolish existing principles or introduce new principles. At this juncture the capability of the web to influence the technology of model building, model execution and model sharing is clear, and the degree of change appears significant. However, that the potential for influence extends into the principles – the foundational core – is less apparent.

6 WEB-BASED SIMULATION: WHITHER WE WANDER? (Ray J. Paul)

This panel contribution will discuss a variety of new technologies for software development and ways of working that will have an unpredictable effect on the future of simulation modelling.

6.1 Multi-media/Synthetic Environments

The ability to access multi-media on the web clearly introduces greater potential for the use of videos of problem scenarios, for interaction with stake-holders situated at remote locations (for example, when the running model hits an unknown combination of circumstances, an expert stake-holder might be able to determine the successful rules for advance) and sound. For example, on a recent visit made to a Hong Kong container terminal, I was shown a television control centre, computer-based, which had 100% video coverage of the terminal. Whilst its purpose was clearly for security and safety, it requires little imagination to visualise how a simulation of the terminal operation could call up the appropriate video camera when problem discussants get to the point of a simulation run where clarification is desired.

I think that the rush to join the much-hyped bandwagon of Synthetic Environments, driven by technical extravagance and financial greed, is in great danger of neglecting or even forgetting those major simulation issues of ongoing concern over the years. These are the so far intractable problems of verification and validation. The current enthusiasm for Synthetic Environments is therefore in danger of creating more expensive mistakes to the detriment of the reputation of simulationists, analysts and operations researchers in general.

6.2 Natural Born Webbers

A large proportion of the current generation of students entering higher education in the developed countries are already familiar with the pastime of browsing the Web and playing computer games. Both of these activities might loosely be depicted as approaches based on "suck it and see". Browsing and adventure games encourage the participant to try out alternatives with rapid feed-back, avoiding the need to analyse a problem with a view to deriving the result.

Such web users, in order to use simulation, need and desire development tools that allow for fast model building and quick and easy experimentation. Furthermore, such web users should have a natural affinity to the use of simulation models as a problem understanding approach (Paul and Balmer, 1993 and Paul and Hlupic, 1994). Web-enabled simulation analysts will be opposed to classical software engineering approaches and methodologies. They will be seeking tools that will enable them to assemble rather than build a model. Some feel for the change of "culture" that we can expect from future generations of computer users can be gauged from a recent experience of mine on a visit to Taipei (Taiwan). A class of school children were using the local university's multi-media lab. A ten year old schoolboy was typing in HTML codes faster than I can and dynamically checking it by running a rather impressive text/video/sound demonstration system. The boy could not speak, read or write any English, everything was symbolic to him.

6.3 New Software Technologies

Some have predicted that the software industry will be divided into component factories where powerful and general components will be built and tested, and into component assembly shops where these components will be assembled into flexible business solutions. Such component based development, if it occurs, might give significant gains in productivity and quality as well as known obvious benefits to web-based software development.

6.4 JAVA

JAVA is now so ubiquitous that it might appear unnecessary to comment on it. For completeness the reader is reminded that simulation models in JAVA can be made widely available; an applet can be retrieved and run and does not have to be ported to a different platform or even recompiled or relinked; there is a high degree of dynamism because JAVA applets run on a browser; JAVA built-in threads make it easier to implement simulation following the process interactive paradigms; JAVA has built in supports for providing sophisticated animations and JAVA is smaller, cleaner, safer and easier to learn than C++, allegedly.

6.5 Conclusions

For me, the aforegoing indicates a world of dynamic change, which I welcome, but where it is all going is a matter of conjecture that will be colored more by prejudice and opinion than evidence.

REFERENCES

Arthur, J.D., R.E. Nance and S.M. Henry. 1986. "A Procedural Approach to Evaluating Software Development Methodologies and Associated Products." SRC-87-007, Systems Research Center, Virginia Tech, Blacksburg, VA.

Buxton, J.(ed). 1968. *Proceedings of the IFIP Working Conference on Simulation Programming Languages,* North-Holland.

Cubert, R. and Fishwick, P.A. 1997. "MOOSE: An Object-Oriented Multimodeling and Simulation Application Framework" accepted for *Simulation*, July 1997. Further information on MOOSE can be found in `http://www.cise.ufl.edu/~fishwick/moose.html`

Dijkstra, E.W. 1976. *A Discipline of Programming.* Prentice-Hall, Englewood Cliffs, NJ.

Erkes, J.W.; Kenny, K.B.; Lewis, J.W.; Sarachan, B.D.; Sobolewski, M.W; and R.N. Sum, Jr. 1996. "Implementing Shared Manufacturing Services on the World-Wide Web." *Communications of the ACM,* 39, no. 2 (Feb.):34-45.

Fishwick, P.A. 1995. *Simulation Model Design and Execution*, Prentice-Hall, Englewood Cliffs, NJ..

Fishwick, P.A. 1996. "Web-Based Simulation: Some Personal Observations" In *Proceedings of the 1996 Winter Simulation Conference*, (San Diego, CA, Dec. 8-11). Association for Computing Machinery, New York, 772-779.

Kiviat, P.J. 1967. "Digital Computer Simulation: Modeling Concepts." RAND Memo RM-5883-PR, Santa Monica, CA.

Lackner, M.R. and P. Kribs. 1964. "Introduction to a Calculus of Change." System Development Corp., TM-1750/000/01.

Nance, R.E. 1979. "Model Representation in Discrete Event Simulation: Prospects for Developing Documentation Standards." In *Current Issues in Computer Simulation,* N. Adam and A. Dogramaci (eds), Academic Press, 83-97.

Nance, R.E. 1983. "A Tutorial View of Simulation Model Development." In: *Proceedings of the 1983 Winter Simulation Conference,* (Arlington,VA, Dec 12-14). IEEE, Piscataway, NJ, 325-331.

Paul R.J and D.W. Balmer 1993. *Simulation Modelling.* Lund, Sweden: Chartwell-Bratt Student-Text Series.

Paul R.J. and V. Hlupic 1994. "The CASM Environment Revisited." In: *Proceedings of the 1994 Win-*

ter Simulation Conference, (J.D. Tew, S. Manivannan, D.A. Sadowski and A.F. Seila, Eds.) (11-14 December 1994, Orlando). Association for Computing Machinery, New York, 641-648.

Swartout W.; and R. Balzer. 1982. "On the Inevitable Intertwining of Specification and Implementation." *Communications of the ACM,* 25, no. 7(July):438-440.

Tocher, K.D.T. 1966. "Some Techniques of Model Building." In: *Proceedings, IBM Scientific Symposium on Simulation Models and Gaming,* pp. 119-155, White Plains, NY.

Zeigler, B.P. 1976. *Theory of Modelling and Simulation,* Wiley, New York.

AUTHOR BIOGRAPHIES

ERNEST H. PAGE is a Lead Scientist in modeling and simulation at The MITRE Corporation where he is currently working on the Aggregate Level Simulation Protocol (ALSP) program and is the Principal Investigator for a MITRE Sponsored Research project on web-based simulation. He received the Ph.D., M.S. and B.S. degrees in Computer Science from Virginia Tech in 1994, 1990 and 1988 respectively. He is active within the U.S. DoD modeling and simulation community, serving on the Defense Modeling and Simulation Office (DMSO) Verification, Validation and Accreditation Technology Working Group (1996-) and on the Testing (1995, 1996), Interface Specification (1995) and Time Management (1996) Working Groups within the DMSO High Level Architecture (HLA) initiative. He has also served on the Planning and Review Panel for the Simulation Interoperability Workshops (1997). Dr. Page has served as the Secretary/Treasurer (1995-1997) for the Association for Computing Machinery (ACM) Special Interest Group on Simulation (SIGSIM) and currently holds the position of Vice Chair. He is on the Program Committees for the 1998 SCS International Conference on Web-Based Modeling and Simulation and the 1998 SPIE AeroSense Symposium Technical Conference on Web-Based Simulation. His research interests include discrete event simulation, parallel and distributed systems, and software engineering. He is a member of ACM, SIGSIM, IEEE CS and Upsilon Pi Epsilon. (*Address:* The MITRE Corporation, 1820 Dolley Madison Boulevard, McLean, VA, 22102-3481, URL: `http://ms.ie.org/page`, E-mail: `epage@mitre.org`)

ARNOLD BUSS is an Adjunct Professor of Operations Research at the Naval Postgraduate School. He received a BA in Psychology from Rutgers University, and MS in Systems and Industrial Engineering from the University of Arizona, and a Ph.D. in Operations Research from Cornell University. His research interests include simulation modeling and object-oriented software design. He is a member of INFORMS, MORS, and POMS. (*Address:* Operations Research Department, Naval Postgraduate School, Monterey, CA, URL: `http://dubhe.cc.nps.navy.mil/%7Eahbuss/`, E-mail: `bussa@or.nps.navy.mil`)

PAUL A. FISHWICK is an Associate Professor in the Department of Computer and Information Science and Engineering at the University of Florida. He received the PhD in Computer and Information Science from the University of Pennsylvania in 1986. He also has six years of industrial/government production and research experience working at Newport News Shipbuilding and Dry Dock Co. (doing CAD/CAM parts definition research) and at NASA Langley Research Center (studying engineering data base models for structural engineering). His research interests are in computer simulation modeling and analysis methods for complex systems. He is a senior member of the IEEE and the Society for Computer Simulation. He is also a member of the IEEE Society for Systems, Man and Cybernetics, ACM and AAAI. Dr. Fishwick founded the `comp. simulation` Internet news group (Simulation Digest) in 1987. He has chaired workshops and conferences in the area of computer simulation, and will serve as General Chair of the 2000 Winter Simulation Conference. He was chairman of the IEEE Computer Society technical committee on simulation (TCSIM) for two years (1988-1990) and he is on the editorial boards of several journals including the *ACM Transactions on Modeling and Computer Simulation*, *IEEE Transactions on Systems, Man and Cybernetics*, *The Transactions of the Society for Computer Simulation*, *International Journal of Computer Simulation*, and the *Journal of Systems Engineering.* (*Address:* Department of Computer and Information Science Engineering, University of Florida, Gainesville, FL 32611-6120, URL: `http://www.cise.ufl.edu/~fishwick`, E-mail: `fishwick@cise.ufl.edu`)

KEVIN J. HEALY is the author of the Java-based *Silk* simulation language and a partner in Thread Technologies, a company involved in the development of Internet-based simulation capabilities. He received his Ph.D. in Operations Research from Cornell University. His former positions include Vice-President of Development for Systems Modeling Corp. and Assistant Professor in Purdue Uni-

versity's School of Industrial Engineering (E-mail: `healy@threadtec.com`)

RICHARD E. NANCE is the RADM John Adolphus Dahlgren Professor of Computer Science and the Director of the Systems Research Center at Virginia Tech (VPI&SU). Dr. Nance is also Chairman of the Board of Orca Computer, Inc. He received B.S. and M.S. degrees from N.C. State University in 1962 and 1966, and the Ph.D. degree from Purdue University in 1968. He has served on the faculties of Southern Methodist University and Virginia Tech, where he was department head of Computer Science, 1973-1979. He held a distinguished visiting honors professorship at the University of Central Florida for the spring semester, 1997. Dr. Nance has held research appointments at the Naval Surface Weapons Center and at the Imperial College of Science and Technology (UK). Within ACM, he has chaired two special interest groups: Information Retrieval (SIGIR), 1970-71 and Simulation (SIGSIM), 1983-85. He has served as Chair of the External Activities Board and several ACM committees. He is the US representative to IFIP TC7. The author of over 100 papers on discrete event simulation, performance modeling and evaluation, computer networks, and software engineering, Dr. Nance has served on the Editorial Panel of *Communications of the ACM* for research contributions in simulation and statistical computing, 1985-89, as Area Editor for Computational Structures and Techniques of *Operations Research,* 1978-82, and as Department Editor for Simulation, Automation, and Information Systems of *IIE Transactions,* 1976-81. He served as Area Editor for Simulation, 1987-89 and as a member of the Advisory Board, 1989-92, for the *ORSA Journal on Computing.* Dr. Nance was the founding Editor-in-Chief of the ACM *Transactions on Modeling and Computer Simulation,* 1990-1995. He served as Program Chair for the 1990 Winter Simulation Conference. Dr. Nance received a Distinguished Service Award from the TIMS College on Simulation in 1987. In 1995 he was honored by an award for "Distingusished Service to SIGSIM and the Simulation Community" by the ACM Special Interest Group on Simulation. He was named an ACM Fellow in 1996. He is a member of Sigma Xi, Alpha Pi Mu, Upsilon Pi Epsilon, ACM, IIE, and INFORMS. (*Address:* Systems Research Center and Department of Computer Science, Virginia Polytechnic Institute & State University, Blacksburg, Virginia 24061, E-mail: `nance@vt.edu`)

RAY J. PAUL holds the first U.K. Chair in Simulation Modelling at Brunel University. He previously taught information systems and operational research for 21 years at the London School of Economics. He received a B.Sc. in Mathematics, and a M.Sc. and a Ph.D. in Operational Research from Hull University. He has published widely in book and paper form (two books, over 100 papers in journals, books and conference proceedings), mainly in the areas of simulation modelling process and in software environments for simulation modelling. He has acted as a consultant for a variety of U.K. government departments, software companies, and commercial companies in the tobacco and oil industries. His research interests are in methods of automating the process of discrete event simulation modelling, and the general applicability of such methods and their extensions to the wider arena of information systems. Recent research results have been in automatic code generation, colour graphics modelling interfaces, dynamically driven icon representations of simulation models, machine learning applied to model specification and to output analysis, object oriented approaches, and information systems paradigms. He is currently running a simulation research group, the Centre for Applied Simulation Modelling, made up of six faculty members and eight research students, and the Centre for Living Information Systems, made up of five faculty and six research students. He is currently the Head of the Computer Science and Information Systems Department at Brunel University. (*Address:* Centre for Applied Simulation Modelling, Department of Information Systems and Computing, Brunel University, Uxbridge, Middlesex, UB8 3PH, UK.
E-mail: `Ray.Paul@brunel.ac.uk`)