Proceedings of the 1998 Winter Simulation Conference
D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds.

THE RISE OF WEB-BASED SIMULATION: IMPLICATIONS FOR THE HIGH LEVEL ARCHITECTURE

Ernest H. Page

The MITRE Corporation
1820 Dolley Madison Blvd.
McLean, VA 22102, U.S.A.

ABSTRACT

The relationship between the High Level Architecture and web-based simulation is discussed. The notion of *interoperability* is suggested as a binding force between these initiatives. The question is posed whether the HLA could serve as an interoperability technology for the commercial and academic sectors in the age of web-based simulation. Using the development and adoption of parallel discrete event simulation technologies as a case study, some of the potential barriers to DoD technology transfer are illustrated, and mechanisms through which these barriers may be overcome are suggested.

1 INTRODUCTION

Motivated by the desire to reduce costs and improve quality through interoperability and reuse, the U.S. Department of Defense (DoD) has made significant investments in the area of distributed simulation over the past 15 years. Originating with SIMNET, and evolving through the Distributed Interactive Simulation (DIS) protocols, the Aggregate Level Simulation Protocol (ALSP) and now the High Level Architecture (HLA), DoD has fostered the evolution of standards to support the interoperability of simulations, and the interoperability of simulations and "real world" – e.g. C4I – systems.

The emergence of the world-wide web (WWW) has produced an environment within which many disciplines are re-evaluating their inherent approaches, techniques and philosophies. The disciplines concerned with computer simulation are no exception to this phenomenon; the concept of "web-based simulation" has been introduced and is currently the subject of much interest to both simulation researchers and simulation practitioners within the academic and industrial sectors. Web-based simulation as an area of scholarly pursuit debuted as a 3-paper session at the 1996 Winter Simulation Conference (WSC) and was, by far, the most well-attended session within the modeling methodology track of that conference. This success was repeated at WSC '97, and in January 1998 the first conference dedicated to the topic of web-based simulation (WEBSIM '98) was held as part of the annual Society of Computer Simulation (SCS) Western Multiconference (Hill, Fishwick and Smith 1988). A second web-based simulation conference is planned for the 1999 SCS Western Multiconference.

In this paper we consider the relationship between web-based simulation and the High Level Architecture. We assume the reader has a general familiarity with DoD modeling and simulation (M&S). The High Level Architecture and web-based simulation are briefly reviewed in Sections 2 and 3 respectively. In Section 4, we argue that the notion of *interoperability* provides the key element in the relationship between web-based simulation and the HLA, and that the "culture of the web" may clamor for interoperability in such a manner as to possibly overcome some of the recent barriers to DoD technology transfer in this area. Section 5 examines a few similarities and differences between the widespread adoption of distributed simulation technologies and the adoption of parallel simulation technologies. Conclusions are given in Section 6.

2 THE HIGH LEVEL ARCHITECTURE

The High Level Architecture (HLA) has been proposed and developed to support reuse and interoperation of simulations across the U.S. Department of Defense (Dahmann, Fujimoto and Weatherly 1997). The HLA represents both a generalization and extension of the DIS protocols (Voss 1993) and ALSP (Page, Canova and Tufarolo 1997; Weatherly, Wilson and Griffin 1993; Weatherly et al. 1996). The HLA is defined by three components:

- a common model definition and specification formalism (U.S. Department of Defense 1988a);

- a collection of services describing the HLA runtime environment (U.S. Department of Defense 1988c); and
- a set of rules governing compliance with the architecture (U.S. Department of Defense 1988b).

The HLA is intended to have wide applicability across the full range of defense simulation applications, including those used to support training, analysis and acquisition. The HLA is designed with a high degree of flexibility, permitting arbitrary mixtures of fidelity and resolution.

At the heart of the HLA is the notion of a *federation.* A federation is a collection of *federates* – simulations and other systems – that interoperate using the protocols described by the architecture. A Federation Object Model (FOM), constructed in accordance with the formalism identified in (U.S. Department of Defense 1988a), provides the model specification and establishes a contract between the federates regarding the nature of the activity taking place during federation runtime. Federation execution is accomplished through an HLA Runtime Infrastructure (RTI) which is an implementation of the infrastructure services defined in (U.S. Department of Defense 1988c). In addition to defining services for the RTI, the HLA Interface Specification defines services that must be implemented by federates.

In a typical federation execution, a federate joins the federation, indicates its operating parameters (e.g. information the federate will provide *to* the federation and information it will accept *from* the federation) and then participates in the evolution of federation state until the federate departs the federation, or the simulation terminates. FOM data is provided to the RTI at runtime, enabling the infrastructure to provide a level of enforcement with respect to the "information contract" that the FOM represents.

A flexible time flow mechanism has been defined for the architecture, permitting a wide range of implementations from tightly synchronous, causality preserving, fully reliable interprocess communication to completely asynchronous, nontimestamped, unreliable interprocess communication, and many points in between (Fujimoto 1997). In many respects, the HLA is more than simply a framework for distributed simulation. The HLA service suite is potentially useful to many types of distributed enterprises.

For complete descriptions of the HLA, it's motivations and components, and for access to prototype software, consult the HLA web site, `http://hla.dmso.mil.`

3 WEB-BASED SIMULATION

Web-based simulation is a diffuse topic. Certainly there is *ample* hype surrounding the internet, and everything old is new again thanks to, as Nance describes it, the *wonderfully webbed world* (Page, et al. 1988). Numerous relationships between the web and simulation are evident; one need only examine the proceedings from WEBSIM '98 (Fishwick, Hill, and Smith 1998) to verify that research and development efforts being put forward under the auspices of web-based simulation run a wide gamut.

Fishwick (1996) offers his perspective on the issue of web-based simulation, and identifies many potential impacts of web technologies on simulation, with particular attention given to three areas: (1) education and training, (2) publication, and (3) simulation programs. Extending Fishwick's categories, a review of the current literature base suggests five areas of focus:

- *Simulation as hypermedia.* Text, images, audio, video ... simulation – the nature of the WWW design enables the production, storage and retrieval of "documents" containing any or all of these (and other kinds of) elements. The availability of simulation as a desktop, browser-based commodity has the potential to significantly alter current teaching and training methodologies, both for simulation as a technique, and for disciplines that apply simulation, like engineering, physics, and biology. Paradigms that focus on distance learning and interactive, simulation-based education and training are emerging.
- *Simulation research methodology.* The ability to rapidly disseminate models, results and publications on the web permits new approaches to the conduct of simulation research, and scientific research in general. The practical, economic and legal issues associated with the electronic publication of documents, for example, are numerous (e.g. see (Samuelson 1996)). The electronic publication of simulation models raises additional considerations.
- *Web-based access to simulation programs.* Most commonly associated with the term web-based simulation, this area includes both the remote execution of existing (so-called "legacy") simulations from a web browser through HTML forms and CGI scripts, and the development of mobile-code simulations (e.g. applets) that run on the client side.
- *Distributed modeling and simulation.* This area includes activities that deal with the use of the WWW and web-oriented technologies (e.g. CORBA, Java RMI) as infrastructure to support distributed simulation execution (Klein, Straßburger and Beikrich 1998; Page, Moose and Griffin 1997; Sajoughhian and Ziegler 1998; Shen 1998). Internet gaming issues are included here, as is research in tools, environments and frameworks that support the distributed (collaborative) design and development of simulation models (Cubert and Fishwick 1997; Fishwick 1998).

- *Simulation of the WWW.* Modeling and analysis of the WWW for performance characterization and optimization.

Clearly, web-based simulation is a diffuse topic. But it is a new area of investigation and perhaps it is a characteristic of any new area that a few years must pass before a core set of researchers and practitioners emerge and with them, a core focus. In a panel session held at WEBSIM '98, the issue of a core focus, and core relevance, for web-based simulation was examined in terms of the *fundamental* nature of simulation modeling. One of the points made in that session serves as the basis for the next section.

4 THE COMMON GROUND OF INTEROPERABILITY

The HLA initiative grew out of cost and quality concerns that illustrated a need within the DoD to reuse existing M&S applications. The notion of reuse in this context differs slightly from the standard notion of reuse in the software engineering community in that it reflects a broader notion of *interoperation.* Reuse in the HLA context does not imply simply usurping code or linking libraries from one application into a new application, but rather the interoperation, at runtime, of existing applications to support their effective use in new contexts. In the commercial sector, support for interoperability has not traditionally been viewed as an effective business model. DoD attempts to interest the entertainment industry in previous generations of DoD M&S technology (i.e. DIS) have met with resistance. In a recent report on the subject, one author sardonically quips ,"DoD can help the entertainment industry by having more movie theaters on military bases."(National Research Council, p. 154). But perhaps times are changing.

Ray Paul describes an evolving culture of "natural born webbers" in (Page, et al. 1988). He argues that the current generation of students entering higher education are well-versed in the practice of web browsing and computer gaming. Paul notes that browsing and gaming foster an approach to problem solving that is rapid, evolutionary and often *ad hoc.* This approach is contrasted with traditional approaches that involve meticulousness, analysis and rigor. During his panel presentation, Paul observed that when given a problem to research, most students began their search on the web. Many students, he said, ended their searches there – often either finding complete solutions or collections of partial solutions from which a complete solution was readily craftable. The immediacy of the web, despite it's lackluster performance characteristics, is almost irresistible.

Although Paul does not state it, it seems the culture he observes isn't simply a product of the web and computer gaming. More likely, it is a product of the proliferation of computers in general, or even more generally, this phenomenon may be attendant with the evolution of any "labor-saving" device. Taking a personal example, when this author entered college in 1983, the Computer Science courses were taught using punched cards. Programs were developed by fully writing them out on paper and then (after waiting in a sufficiently long line for an available machine) entering the code into the card punching machine. In the following quarters, interactive time-sharing systems became available to the lower-level students, but computer time – in terms of connection time and CPU cycles – was tightly budgeted. If your account ran out of money, appropriate acts of contrition before the Department Head were required before the account could be replenished. And in some classes, professors established a policy that accounts would not be replenished. I recall at least two of my early classmates whose majors were changed as a result of an infinite loop.

In later years, the accounting practice became transparent to the users, and around 1986, or so, entering Freshmen were required to come equipped with their own personal computer. I don't remember if teaching approaches changed significantly during this period, but the programming practices among the student populace certainly did. It was common, if not predominant, by 1987, or so, for students to begin program development from an empty *vi* window. So it appears that a certain level of rigor was lost as the personal computer emerged. How much was lost with the advent of assembly languages, high-level languages, virtual memory?

Is the edit-compile-run-edit-compile-run approach better or less suited to problem solving using computers? Probably it is a matter of scale. Certainly for small programs the interactive approach is quite reasonable and efficient. For very large systems, though, the scope of the task at hand would seem to mandate a certain level of analysis and planning prior to code development. But here, too, a rapid build-test-build approach is easily accommodated with modern CASE tools. (During his panel presentation, Paul took a somewhat controversial position, implying that highly structured approaches to software development are not suited to building *any* large-scale system!)

Regardless of suitability or effectiveness, the rising generations of computer users – and simulationists – are imbued with a desire and propensity for rapid, interactive approaches to problem solving. In a world of web-publishable digital objects (Fishwick 1998), they will gravitate toward solutions reflecting the integration of existing components. They will demand interoperability. How the marketplace will respond is an open question.

5 A (SEMI) CAUTIONARY TALE

The framers of the HLA have been driven by legitimate, pressing needs and admirable goals. But the definition of the architecture exhibits signs of a failure to appreciate "mainstream" simulation practices and priorities. A similar tack was taken by the parallel discrete event simulation (PDES) community during its formative years.

Parallel discrete event simulation techniques were defined and developed largely by parallel computing researchers who saw simulation as an interesting and difficult application area. Generally speaking, they were not simulationists. They understood how simulation programs worked – they had to in order to define methods for executing them in parallel. But they had lesser understanding of how simulation was *used* as a problem solving and decision support tool. They had little feeling for the life cycle factors outside of the programming task, such as input data modeling, and model validation. These issues were perceived as outside the scope of their interest – PDES research was focused on the issue of increased model performance, *speedup.*

PDES researchers demonstrated little concern for the traditional conceptual frameworks adopted in the discrete event simulation community. The techniques defined for PDES required a conceptual and physical model organization that was dissimilar to the traditional approaches. There isn't anything inherently wrong with requiring a different perspective, of course, but over two decades of tool support had emerged for event scheduling, activity scanning and process interaction approaches. Automated support for the logical process view required by PDES approaches has been slow to emerge.

PDES researchers, like most engineers and scientists, were hopeful that their techniques would find widespread utility. They assumed that if users were interested in improved performance of their simulations, they would adopt PDES approaches despite the difficulties involved in the transition. It became evident that this assumption was incorrect (Page and Nance 1994) and that failure to receive mainstream adoption was due, at least in part, to a failure of the PDES community to provide solutions in forms sympathetic to those already in use by the DES mainstream.

So, PDES remains a somewhat niche area. In typical settings, if increased performance is desired, parallel replications are performed or faster machines are acquired. The traditional approaches to model development, analysis and execution have not been supplanted by the PDES approaches. This is not to say that PDES techniques are not products of tremendous intellectual effort and ingenuity. They unquestionably are. And these techniques have been of great utility to much of the distributed simulation work that undergirds the HLA. The point to be made here is simply that if widespread adoption of PDES techniques by the "mainstream" simulation community were ever a goal, it has not yet been achieved.

In several key aspects, the definition and development of the High Level Architecture reflects a similar unfamiliarity with "mainstream" simulation that PDES exhibited. Three areas are briefly considered below.

- *Simulation model specification.* The Object Model Template (OMT) which defines the basis for model definition and specification in the HLA is derived of object-oriented analysis and design (OOA/OOD) techniques for general-purpose software. The OOA/OOD techniques are quite sufficient to capture the static aspects of model definition, but it is widely viewed that the specification of simulation model *dynamics* is the critical element of model specification, and the OMT provides little support for this. There is no indication in the body of literature that surrounds the HLA that simulation-based model definition and specification formalisms – e.g. DEVS (Zeigler 1976), event graphs (Schruben 1992) – or other techniques that allow the specification and analysis of timing properties – e.g. Petri nets (Törn 1991), temporal logics (Henzinger, Manna and Pnueli 1990) – have been considered for roles within the OMT.

- *Centrality of model objectives.* Since *every* model is an abstraction, its quality must be judged in terms of its intended use. The intended use for the model determines the degree of faithfulness to the real system that the model must exhibit. This is the first principle of model validation as identified in (U.S. Department of Defense 1996). It goes to the very *foundation* of the cost-effective use of simulation models. One of the basic tenets of every simulation modeling methodology is the enunciation of the modeling objective and the assumptions underlying the model. Unless captured as part of the unstructured *object model metadata*, these specifications are unsupported in the OMT.

- *Conceptual frameworks.* The HLA provides client applications with a low-level, operating system-like conceptual framework. The conceptual framework support available in modern simulation programming languages (SPLs) and simulation support environments (SSEs) is absent in in the HLA. Perhaps such support can (should) only come from the language and not the architecture. At the moment this is unclear. As efforts such as (Klein, Straßburger and Beikrich 1998; Page, Griffin and Rother 1988) mature the proper support roles between language and architecture will emerge.

6 CONCLUSIONS

The HLA has been mandated as a DoD standard. All DoD modeling and simulation (M&S) efforts must comply with the HLA, receive a waiver, or be retired by 2001. The DoD has made similar proscriptions in the past – with mixed success. The recently rescinded Ada mandate is an example. Could HLA realize the same fate as a DoD standard that befell Ada? Will DoD see a similar impedance to mainstream acceptance of their distributed simulation technologies that met those offered by the parallel discrete event simulation community? It is possible. On the other hand, HLA is potentially quite suitable to provide the framework for distributed simulation systems interoperability in the academic and commercial sectors.

The cultural demand for interoperability will likely drive commercial and academic communities to define solutions to systems interoperation. Their need for interoperability may enable to adoption of HLA – even if it does not completely meet their needs. But it might not. PDES techniques have yet to receive widespread adoption. If the DoD is interested in transferring the technology of HLA into the mainstream simulation community, DoD would be well advised to understand and support the approaches, techniques and philosophies of the mainstream.

REFERENCES

Cubert, R.M., and P.A. Fishwick. 1997. A framework for distributed object-oriented multimodeling and simulation. In *Proceedings of the 1997 Winter Simulation Conference,* ed. S. Andradóttir, K.J. Healy, D.H. Withers, and B.L. Nelson, 1315-1322, Association for Computing Machinery, New York, NY.

Dahmann, J.S., R.M. Fujimoto, and R.M. Weatherly. 1997. The Department of Defense High Level Architecture. In *Proceedings of the 1997 Winter Simulation Conference,* ed. S. Andradóttir, K.J. Healy, D.H. Withers, and B.L. Nelson, 142-149, Association of Computing Machinery, New York, NY.

Fishwick, P.A. 1996. Web-based simulation: some personal observations. In *Proceedings of the 1996 Winter Simulation Conference,* ed. J.M. Charnes, D.M. Morrice, D.T. Brunner, J.J Swain, 772-779, Association for Computing Machinery, New York, NY.

Fishwick, P.A. 1998. Issues with web-publishable digital objects. To appear in *Proceedings of SPIE: Enabling Technologies for Simulation Science II,* ed. Alex Sisti, SPIE Press, Bellingham, WA.

Fishwick, P.A., D.R.C. Hill, and R. Smith, Eds. 1998. *Proceedings of the 1998 International Conference on Web-Based Modeling and Simulation,* SCS Simulation Series **30**(1), San Diego, CA, 10-14 January.

Fujimoto, R.M. 1997. Zero lookahead and repeatability in the High Level Architecture. In *Proceedings of the 1997 Spring Simulation Interoperability Workshop,* Orlando, FL, 3-7 March.

Henzinger, T. A., Z. Manna, and A. Pnueli. 1990. An interleaving model for real time, In *Proceedings of the 5th Jerusalem Conference on Information Technology,* IEEE Computer Society Press, 717-730.

Klein, U. S. Straßburger, and J. Beikrich. 1998. Distributed simulation with JavaGPSS based on the High Level Architecture. In *Proceedings of the 1998 SCS International Conference on Web-Based Modeling and Simulation,* ed. P.A. Fishwick, D.R.C. Hill, and R. Smith, 85-90, Society for Computer Simulation International, San Diego, CA.

National Research Council. 1997. *Modeling and Simulation: Linking Entertainment and Defense.* Washington, DC: National Academy Press.

Page, E.H., and R.E. Nance. 1994. Parallel discrete event simulation: a modeling methodological perspective. In *Proceedings of the 8th Workshop on Parallel and Distributed Simulation,* ed. D.K. Arvind, R. Bagrodia, and J. Y.-B. Lin, 88-93, Association for Computing Machinery, Baltimore, MD.

Page, E.H., B.S. Canova, and J.A. Tufarolo. 1997. A case study of verification, validation and accreditation for advanced distributed simulation. *ACM Transactions on Modeling and Computer Simulation,* 7(3):393-424.

Page, E.H., S.P. Griffin, and S.L. Rother. 1988. Providing conceptual framework support for distributed web-based simulation within the High Level Architecture. To appear in *Proceedings of SPIE: Enabling Technologies for Simulation Science II,* ed. Alex Sisti, SPIE Press, Bellingham, WA.

Page, E.H., R.L. Moose, and S.P. Griffin. 1997. Web-based simulation in Simjava using remote method invocation. In *Proceedings of the 1997 Winter Simulation Conference,* ed. S. Andradóttir, K.J. Healy, D.H. Withers, and B.L. Nelson, 468-474, Association for Computing Machinery, New York, NY.

Page, E.H., A. Buss, P.A. Fishwick, K.J. Healy, R.E. Nance, R.J. and Paul. 1998. The modeling methodological impacts of web-based simulation. In *Proceedings of the 1998 SCS International Conference on Web-Based Modeling and Simulation,* ed. P.A. Fishwick, D.R.C. Hill, and R. Smith, 123-128, Society for Computer Simulation International, San Diego, CA.

Samuelson, P. 1996. Regulation of technologies to protect copyrighted works. *Communications of the ACM*, 39(7):17-22.

Sarjoughhian, H.S., and B.P. Zeigler. 1998. DEVSJAVA: basis for a DEVS-based collaborative M&S environment. In *Proceedings of the 1998 SCS International Conference on Web-Based Modeling and Simulation*, ed. P.A. Fishwick, D.R.C. Hill, and R. Smith, 29-36, Society for Computer Simulation International, San Diego, CA.

Shen, C.-C. 1998. Discrete-event simulation on the Internet and the web. In *Proceedings of the 1998 SCS International Conference on Web-Based Modeling and Simulation,* ed. P.A. Fishwick, D.R.C. Hill, and R. Smith, 57-62, Society for Computer Simulation International, San Diego, CA.

Shruben, L. 1992. *SIGMA: Graphical Simulation Modeling,* San Francisco: The Scientific Press.

Törn, A.A. 1991. The simulation net approach to modelling and simulation. *Simulation,* 57(3):196-198.

U.S. Department of Defense. 1996. Verification, validation and accreditation (VV&A) recommended practices guide, ed. O. Balci, P. Glasow, P. Muessig, E. H. Page, J. Sikora, S. Solick and S. Youngblood. Defense Modeling and Simulation Office, Alexandria, VA, November.

U.S. Department of Defense. 1998a. High Level Architecture Object Model Template, Version 1.3, February.

U.S. Department of Defense. 1998b. High Level Architecture Rules, Version 1.3 (Draft 2) February.

U.S. Department of Defense. 1998c. High Level Architecture Interface Specification, Version 1.3 (Draft 9) February.

Voss, L.D. 1993. *A Revolution in Simulation: Distributed Interaction in the '90s and Beyond,* Arlington, VA: Pasha Publications, Inc.

Weatherly, R.M., A.L. Wilson, and S.P. Griffin. 1993. ALSP – theory, experience, and future directions. In *Proccedings of the 1993 Winter Simulation Conference*, ed. G.W. Evans, M. Mollaghasemi, E.C.Russell, and W.E. Biles, 1068-1072, Association for Computing Machinery, New York, NY.

Weatherly, R.M., A.L. Wilson, B.S. Canova, E.H. Page, A.A. Zabek, and M.C. Fischer. 1996. Advanced distributed simulation through the aggregate level simulation protocol. In *Proceedings of the 29th Hawaii International Conference on Systems Sciences,* Vol. 1, 407-415.

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

AUTHOR BIOGRAPHY

ERNEST H. PAGE is a Lead Scientist in modeling and simulation at The MITRE Corporation where he is currently serving on the OneSAF program, 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. 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 the Program Chairman for the 1999 SCS International Conference on Web-Based Modeling and 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`)