Proceedings of the 1995 Winter Simulation Conference
ed. C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman

INTRODUCTION TO ARENA®

John E. Hammann
Nancy A. Markovitch

Systems Modeling Corporation
504 Beaver Street
Sewickley, Pennsylvania 15143, U.S.A.

ABSTRACT

This paper highlights the features and functions of Systems Modeling Corporation's general purpose simulation system, Arena. Arena represents an advancement in simulation technology by enabling enterprise-wide simulation. It is a comprehensive system that addresses all phases of a simulation project from input data analysis to the analysis of simulation output data.

Building on the capabilities of Systems Modeling's earlier products, SIMAN and Cinema, Arena uses a hierarchical approach to provide the user with the power of a simulation language and the flexibility of a simulator. The object-oriented approach that was central to Arena's development, coupled with the hierarchical architecture, enables the professional user to define the personality of the system for the end-user. The professional user can actually build his or her own simulation system by combining SIMAN and Arena constructs into modules for the end-user.

Arena is focused on bringing the use of simulation to broad new classes of users. Its application focus addresses the needs of manufacturing as well as decision support for many other areas including, business process reengineering, medical systems, transportation, logistics, and data communications.

1 INTRODUCTION

The many recent advances in simulation technology have created a greater awareness and use of simulation in industry. Many managers are now more aware of the potential benefits of simulation. However, even with the many important advances that have been made over the past several years, there are still many cases where complex systems are being designed and implemented without the benefit of simulation. A very small percentage of systems that could benefit from simulation are actually simulated; the primary reason for this is the high level of effort required to employ simulation technology successfully. The key to making simulation technology more widely used is to make the tools significantly easier to learn and use without sacrificing modeling power and flexibility.

This paper describes the Arena simulation system developed by Systems Modeling Corporation. Arena is a one-step, graphical modeling and animation system that is based on concepts from object-oriented programming and hierarchical modeling (see Pegden 1992).

The key idea behind Arena is the concept of tailoring it to a specific application area. The Arena system is not restricted to a specific set of predefined modeling primitives, but can be tailored easily to a specific application by means of a template. For example, an Arena user who is modeling network communication systems, could employ a template that would contain modeling constructs focused on networking systems such as node, hub, bridge, router, concentrator, etc.

An extension of this is the reusability at this level. That is, unlike conventional simulation systems that have their modeling primitives "hard-coded" into the software by the vendor, Arena provides the end user the freedom to use the template again and again in many different situations and under different constraints by utilizing only "soft-coded" information.

The power of Arena is in providing the user with the capability to model quickly and easily with a tool that is customized to his or her application, but not limiting the user to only the constructs provided by a single domain-restricted construct set. The user can combine constructs from one or more application-focused templates with the full modeling power of the SIMAN simulation language and thereby avoid the modeling "brick wall" encountered with traditional hard-coded, domain-restricted packages. The modeling power of SIMAN is made available to the user as simply one additional template, so that at *any* level (even the lowest, most detailed level), the modeling environment and approach are the same. That is, it

remains an object-oriented, drag-and-drop, fill-in-the-blanks style with no programming involved.

The template concept is fundamental to the flexibility and ease-of-use provided by Arena. This mechanism makes it possible to provide the end user with a tool that closely matches the real system being modeled; hence the user is presented with concepts and terminology that are familiar and focused to his or her application. This dramatically reduces the level of modeling abstraction and, therefore, helps to bring simulation to a new class of users who may not necessarily be technically oriented, but who know their system well. Expanding the user base in this way helps make simulation more visible within industry, thereby broadening the overall use of simulation as a decision support and planning tool.

2 THE ARENA SYSTEM

Arena provides an integrated framework for building simulation models in a wide variety of applications. An entire simulation project may be completed within the Arena system, whereby integrated support is provided for all of the functions necessary to complete a successful simulation (including input data analysis, model building, interactive execution, animation, execution tracing, model verification, and output analysis).

2.1 Model Window

In the model window, new models are created, existing models are modified, animations are developed and models are executed. Arena takes full advantage of the easy-to-use Windows® operating system interface by including such components as 3-D controls, toolbars, status bars, tool tips, a context-sensitive help system, True-Type fonts, document windows, and common dialogs. Like all windows in Arena, the model window can be moved, resized, enlarged to full screen, or minimized to an icon. You also can open multiple model windows in the work area at the same time. When a model window is first open, all of the toolbars are automatically displayed. The modeler can essentially design his or her own workspace by moving the toolbars around or turning them off based on his or her own preference.

Models are developed in the model window by attaching a template panel to the template toolbar, selecting modules from the template panel and placing them in the model window; modules are then interconnected to form a logical flow, and the modules' dialogs are edited to reflect the system parameters and characteristics.

Each template contains a number of modules used for model development. Each module represents one or more modeling constructs used to depict a component of a system. For example, a model of a simple service system might be built by interconnecting modules from a general purpose modeling template representing an arrival process, service process, and departure process. In describing a system, modelers can mix modules from various different templates. Additional animation symbols, graphs and plots may be added into the window by using objects found on the Animate toolbar. Additionally, background symbols or text can be added by selecting objects from the Draw toolbar.

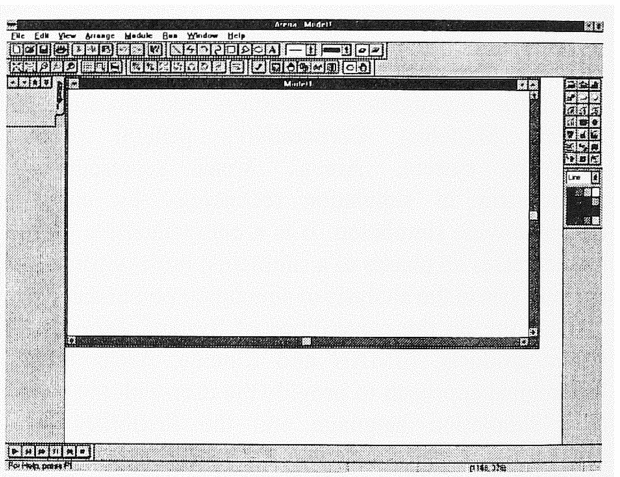

Figure 1: New Model Window

2.2 Template Window

In the template window, new modeling constructs (modules) are created and existing modules are modified. A template panel contains modules collected in a file and intended to be presented as a self-contained group. A template consists of a panel or set of panels that encompass modeling constructs for a particular application.

A module's definition is created by working with five windows: operand, logic, user view, switch, and panel icon. The template window provides a base from which these five module definition windows are opened.

Three aspects of the module definition are visible to the user of the module: the panel icon, the user view, and the module's dialog and operands. When a template is attached to the template toolbar for use with a model window, the panel icons are displayed. The panel icon window is the mechanism used within the module's definition to define the icon for that module.

After a module has been selected and placed in the model window by the user, an instance is formed and the module's user view is displayed. The user view consists of a module handle (the name of the module, displayed as a text object that opens the module's main dialog when the modeler double-clicks on it), and may contain connection points, static drawing graphics, and/or animation objects. This user view is designed by the template developer from within the user view window.

After a module is placed in a model window, its operands may be edited by the user. The template designer decides which operands are to be presented to the modeler, the default values, and their organization into one or more dialogs. A module's dialogs (and the operands contained in them) are defined using the operand window.

The two final aspects of a module are hidden from the modeler: the module logic and the definitions of module switches. The logic underlying an Arena module is created by building an Arena "submodel" from within the Logic Window.

Finally, in an Arena module definition, individual objects in the user view, operand, and logic windows may be selected to be included in an instance only if a particular condition is true. To define this behavior, objects called switches are created in the module definition. The definition of a switch is based on conditions involving the values of operands. The switch is "attached" to an object in one of the three other definition windows (user view, operand, or logic).

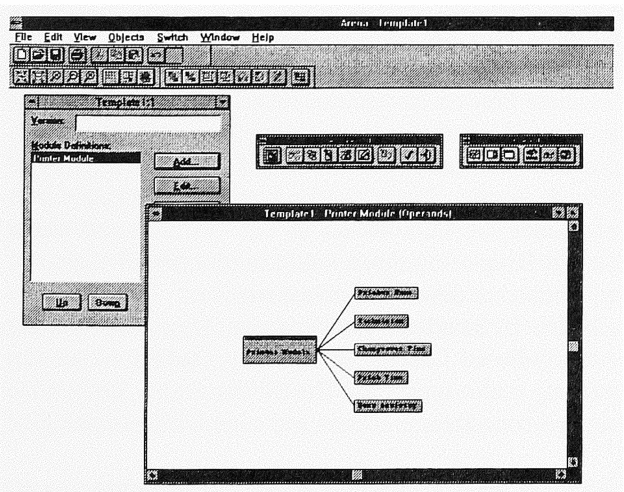

Figure 2: Template Operand Window

2.3 Input Analyzer

The graphical, menu-driven Input Analyzer provides users with the capability of fitting process or performance data to statistical distributions. Users can evaluate data downloaded from a file and may display the input data in the form of a histogram. The data may then be fit to one of 14 commonly referenced distributions. Users unfamiliar with the data characteristics may choose to do a best fit analysis by selecting the Fit All option. This will perform statistical tests and choose the distribution that has the minimum mean square error. Additionally, the expression required as input to the Arena model will be provided, and can be copied directly from the Input Analyzer and pasted into your Arena model.

Other options provided allow the user to modify parameters related to the input distribution to identify alternative input scenarios.

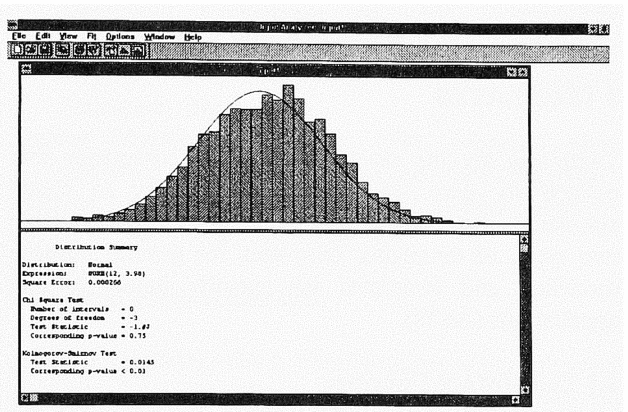

Figure 3: Curve Fitting with the Input Analyzer

2.4 Output Analyzer

The graphical, menu-driven Output Analyzer provides users the capability to evaluate the statistical reliability of the results generated from their simulation model. Commonly used output data testing procedures such as correlograms, t-tests, data filters, moving averages, analysis of variance, and confidence intervals are included. These tests allow users to define and evaluate system performance or to compare the results from two or more simulation runs that evaluate different scenarios. The Output Analyzer is integrated within Arena such that a user can execute a simulation run, then select output analysis options that will display system performance characteristics.

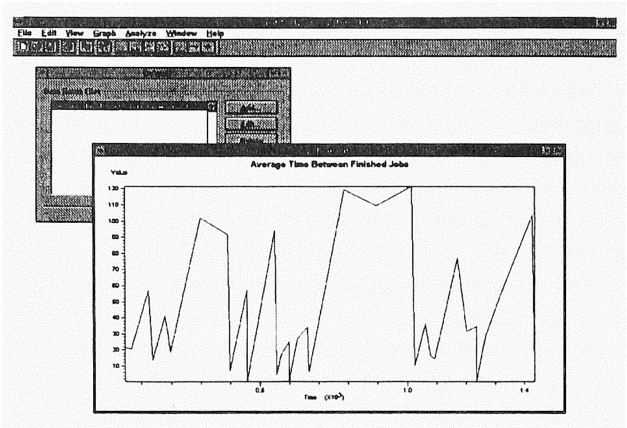

Figure 4: Performance Plot with Output Analyzer

3 TEMPLATES

Templates are the set of modules or commands that collectively give the modeler the ability to create simulation models. Systems Modeling makes available two types of templates: templates for general purpose modeling and templates for specific industries and applications (ASTs).

3.1 The SIMAN Template

The SIMAN template contains the base modules that correspond directly to the SIMAN blocks and elements. The SIMAN template contains two panels, the Blocks panel which contains logic modules, and the Elements panel which contains data modules. Users developing a model with the SIMAN template have full access to the power and flexibility of the SIMAN simulation language.

3.2 The Arena Template

The Arena template is a collection of modules that are useful for modeling a broad range of systems. The modules in this template capture the logic and system elements that are commonly encountered in simulation modeling, such as arrivals, servers, inspection processes, and material transport systems. By providing high level tools for representing these processes, the Arena template enables the user to build complete, graphically animated models quickly with no programming.

Three panels compose the Arena template: the Common panel, containing modules representing fundamental simulation processes such as arrivals, service, and departures; the Support panel, containing supplemental modules for modeling specific actions and decision logic; and the Transfer panel, whose modules are used to model the transfer of entities through the system and the representation of material transfer devices. Modules from these panels may be combined in the same model or may be used with modules from other templates, such as the SIMAN template or an Application Solution Template.

A hierarchical relationship exists among modules in the Arena template. For example, the Advanced Server module in the Common panel incorporates three steps (entering the server, processing at the server, and leaving the server) that are defined individually by the Enter, Process, and Leave modules, also in the Common panel. Each of these modules, in turn, is defined by a sequence of activities that are captured in modules contained in the Support and Transfer panels.

3.3 Application Solution Templates

The Arena Professional Edition is used to create an Application Solution Template (AST), a custom set of modules that contain a user interface and logic specific to a given application or industry. ASTs rapidly decrease the learning curve normally associated with the use of simulation and are a key facilitator of enterprise-wide simulation.

The following ASTs are currently available for Arena: the BPSimulator, the Advanced Manufacturing AST, and the Wafer Fabrication AST.

3.3.1 The BPSimulator

The BPSimulator is designed to make it easy to simulate business processes and their interrelations. It is intended to be an easy-to-use simulator in support of a company's Business Process Reengineering efforts. Model building is focused on the activities that comprise a business process. To build the model, Activity modules are placed in the Arena workspace. The modeler then completes the information in the module's associated dialog. By placing those modules, the model logic and animation are developed in one step.

The BPSimulator contains the logic for tracking Activity-Based Costing through any process. Fixed cost tracking is also supported. The BPSimulator is designed to support concurrent processes and complex decision logic.

3.3.2 The Advanced Manufacturing Template

The Advanced Manufacturing template provides a simplified approach for modeling complex manufacturing systems. The template was designed to support the majority of discrete manufacturing environments—from job shops to flow shops to assembly lines.

The template supports various types of process flow including unconstrained push, constrained push, pull or

a combination of those styles. The template also facilitates modeling Kanban and JIT environments.

The Advanced Manufacturing template makes complex decision making easy when determining part flow from one work center to another. Model logic can be based on probabilities, part types, process plans, selection rules, or user defined conditions. The template generates extensive statistics including queue times and lengths, workcenter utilization, resource state, and part flowtimes.

3.3.3 The Wafer Fabrication Template

The Wafer Fabrication template is designed specifically to be used in modeling the operation and performance of semiconductor wafer fabrication facilities. Models that are built using this template utilize two types of user input, model data and model rules. Model data definitions include products, technologies, process plans, recipes, workcenters, resources, operators, and production schedules. Model rules define the specific logic by which a work order lot of wafers are processed. This architecture gives the Wafer Fabrication template great flexibility in modeling semiconductor fabs.

4 SUMMARY

Arena is an object-based, hierarchical modeling tool that addresses a wide range of applications. It gives the simulation analyst a decision support tool that combines the capabilities and power of a simulation language with an easy user interface for model building. Arena is a comprehensive toolset that spans the scope of a simulation project from input data analysis to the analysis and presentation of model results.

Through the use of Application Solution Templates, the analyst works in a simulation environment that defines the personality of Arena for him or her. Templates provide an environment that can be customized to represent the industry or application of the analyst. The use of these templates also facilitates enterprise-wide simulation by making the power of simulation available to a wider audience of potential users.

REFERENCES

Drevna, M., and C. Kasales. 1994. Introduction to Arena. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J.D. Tew, M.S. Manivannan, D.A. Sadowski, and A.F. Seila, 431-436. Institute of Electrical and Electronic Engineers, Piscataway, NJ.

Hammann, J.E., and T. Matwijec. 1994. Enterprise-Wide Simulation-The Next Generation. In *Proceedings of Manufacturing in Southern Africa 1994*, 149-153. South African Institue of Industrial Engineers, Capetown, South Africa.

Pegden, C.D., D. Davis. 1992. Arena: A SIMAN/Cinema-Based Hierarchical Modeling System. In *Proceedings of the 1992 Winter Simulation Conference*, ed. J.J. Swain, D. Goldsman, R.C. Crain, J.R. Wilson, 390-399. Institute of Electrical and Electronic Engineers, Piscataway, NJ.

Pegden, C.D., R.E. Shannon, and R.P. Sadowski. 1995. *Introduction to Simulation Using SIMAN.* 2nd ed. New York: McGraw-Hill

Profozich, D., and D. Sturrock. 1994. Introduction to SIMAN/Cinema. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J.D. Tew, M.S. Manivannan, D.A. Sadowski, and A.F. Seila, 427-430. Institute of Electrical and Electronic Engineers, Piscataway, NJ.

Systems Modeling Corporation. 1994. *Advanced Manufacturing Template Reference Guide.* Sewickley, PA.

Systems Modeling Corporation. 1995. *Arena Template Reference Guide.* Sewickley, PA.

AUTHOR BIOGRAPHIES

JOHN E. HAMMANN is Vice President of the Sales and Marketing Division of Systems Modeling Corporation. He directs the company's sales and marketing functions and coordinates all international business relations. Before joining Systems Modeling in 1988, he held positions in systems engineering, sales, and management with IBM, Hewlett-Packard, and ASK Computer Systems. John received his bachelor of arts in Economics from Lock Haven State University and his MBA from The Pennsylvania State University. He is certified (CPIM) in production management from the American Production and Inventory Control Society. He is also active in the Pittsburgh High Technology Council.

NANCY A. MARKOVITCH is an Applications Engineer in the Product Applications Division of Systems Modeling Corporation. Her responsibilities include technical phone support, in-house product support and training, technical sales support, client relations, marketing support, contributions to quarterly publications, software and template testing, custom template design and development, and simulation modeling. Nancy also currently serves as Systems Modeling's Release Coordinator. In this role, she oversees all details of the Arena 2.0 release. Nancy holds a bachelor's degree in Industrial and Systems Engineering from Youngstown State University. She is a member of the Institute of Industrial Engineers and the Society of Women Engineers.