Proceedings of the 1991 Winter Simulation Conference
Barry L. Nelson, W. David Kelton, Gordon M. Clark (eds.)

CONCEPTS FOR PRODUCTION MODELING SYSTEMS BASED ON MULTIPLE USER TYPES

Charles R. Standridge
Martha A. Centeno

Department of Industrial Engineering
FAMU/FSU College of Engineering
Post Office Box 2175
Tallahassee, Florida 32316

ABSTRACT

New approaches to the application of computer-based models in the production system design process are needed due to the intrinsic complexity of designing such sophisticated and integrated systems. A Production Modeling System (PMS) is a computer-based modeling environment for developing and applying computer models in the production system design process. A PMS has several key characteristics including a single model representation of the system which enables multiple analysis types, adaptability of PMS functionality to the system under study, transparent information transfer between analyses, and gateways to external information sources. In addition, a PMS supports multiple user types including those that make decisions, those that use models to evaluate alternatives, those that construct models, those that tailor the PMS to the system under study, and those that build the software that provides the functionality of the PMS. A PMS is implemented through the integration of multiple analysis techniques such as simulation and optimization, databases, and knowledge bases.

1 INTRODUCTION

Manufacturing competitiveness is a critical issue in the United States today. Production systems that transform raw materials into high quality and highly reliable products are necessary for the success of our country in the world market place. More sophisticated production systems are being developed to address this need. For example, computer integrated manufacturing (CIM) technology seeks to improve the quality, efficiency, speed, and cost of production through the integration of production related information and sophisticated management strategies under the control of a computer system. The importance of these emerging technologies and their derived systems has been recognized by various leading institutions in the U.S. For instance, in Florida, CIM has been identified as one of seven key technology areas for research emphasis for economic development by the State University System Board of Regents in co-operation with the Florida High Technology and Industry Council.

The intrinsic complexity of designing sophisticated and flexible production systems, such as CIM systems, requires the support of powerful design tools. Initial designs for new systems and revised designs for evolving systems must be produced quickly and in a methodologically correct manner. Over the past decade, computer based models have proven their value by generating necessary information, available from no other source, for evaluating and improving designs. The construction, analysis, and application of these computer models are supported by computer-based modeling environments.

The effective use of models in the production systems design process can be increased by embodying more comprehensive methods and tools in future modeling environments. Production systems designers will be doing the same things they do now, except that, both individually and in teams, they will be able to complete more activities more effectively within the same cost and time constraints. Stated another way, future computer-based modeling environments 1) will help relieve the designers of low-level, repetitive tasks, 2) will incorporate a variety of analysis tools, 3) will support multiple types of analysis to address a wide variety of design questions, and 4) will ensure that the tasks in the modeling and analysis process are performed in a methodologically correct manner.

In this paper, we present the concepts and functional requirements necessary to create modeling environments that support the design of sophisticated, flexible production systems. Section 2 discusses the need for such modeling tools. The functional requirements for new modeling environments focused on production systems are presented in section 3. A general architecture for these environments is given in section 4. Section 5 presents an example application for such a computer-based modeling environment.

2 MOTIVATION

In a wide variety of design problems and cases, operations research (OR) tools have proven successful in supporting the resolution of design issues through analysis. Many different operations research models may be necessary during the design of a single production system. Usually, the computer-based versions of the models are built independently of each other which may be undesirable for the following reasons:

~ Redundant Models. The description of the production system is entered for each model. As models evolve, the consistency of system representation between models may become less and less. This may lead to inaccurate results and errors being propagated among many analyses.

~ Manual Information Transfer. Information shared between models must be transferred manually. As models are analyzed in an iterative fashion, the clock time to evaluate alternatives may be too long and the process prone to manual transfer errors.

Current general purpose modeling languages are not always well suited to production systems due the difficulty of model design. The modeler must perform the cognitive task of mapping between the objects in the production system and the constructs provided by the general purpose modeling language. This may require "guru" level expertise as the modeling constructs may not allow a straightforward representation of the complex interactions in a production system. Compounded with the redundant models problem, "guru" level expertise may be required in multiple modeling areas, for example simulation and optimization.

Alternatively, some production specific modeling tools have been developed. However, these tools have not been widely used because of their lack of modeling power. From the modeler's view point, the tool provides a non-extendible (or difficult to extend) set of modeling constructs making the accurate representation of production systems difficult. Furthermore, the set of modeling constructs is based on a generic view of production systems and may cause the same difficulty of model design as a general purpose tool.

We propose an approach for production systems modeling that shows promise in relieving current difficulties and enhancing the production systems modeling process. The proposed approach centers around the idea of a Production Modeling System (PMS) that integrates multiple operations research techniques, databases, and knowledge bases to jointly and concurrently help resolve multiple production system design issues. Thus, a PMS should have the following characteristics:

~ A single model for production systems representation. All information needed for all analyses is integrated and represented in one model. Different views or subsets of the model may be selected (transparently) to support different types of analyses.

~ Adaptable functionality. Modeling and analysis capabilities, including the set of model building constructs, data and knowledge editors, analysis controllers and result processors, must be tailorable and adaptable to the particular production system or class of production systems of interest.

~ Transparent information transfer. Information generated by one type of analysis is automatically and transparently captured and stored for later use by other types of analyses.

~ Gateways to external databases. Needed information residing outside of the PMS is accessible to support model construction and analysis specification as well as for other purposes such as project management and classical corporate information processing activities.

~ Support for multiple user types. Users who build models, users who execute models for evaluating alternatives, and decision makers who need the information generated by models must all be supported by the system. A system builder tailors a PMS to a particular production system or class of production systems. A tool builder constructs generic or ad hoc components that provide all of the functional capabilities needed by a PMS. Consequently, the functional requirements of a typical PMS are defined by the needs of its user types.

3 PMS USER TYPES AND FUNCTIONAL REQUIREMENTS

Multiple types of users, each with distinctive skills and objectives, will employ a PMS. These user types represent typical roles in a modeling and analysis project. For any particular project, the same individual may take on multiple roles or many individuals may participate in the same role. We have identified five type of users: 1) decision makers, 2) model users, 3) model builders, 4) system builders, and 5) tool builders. Their distinctive roles and needs are discussed in the following sections.

3.1 Decision Makers

A primary goal of a modeling and analysis project is to provide critical information, recommendations, and unique perspectives, unavailable from any other source, for decision making in a timely fashion. Often meeting this goal requires the development and assessment of multiple alternatives.

A decision maker will choose among multiple alternatives of a given production system, based on the analysis results of each one. Therefore, for the decision maker, a PMS should support the transfer of information generated in the modeling and analysis phases into forms suitable for management evaluation, and report and presentation generation. In an advanced PMS, computer support for the development of problem statements and analysis goals could also be provided.

3.2 Model Users

The primary focus of a model user is supporting the decision maker through the assessment of system alternatives using an existing model. This user describes alternate scenarios through model input parameter values and examines results to assess and compare alternatives. Thus, this user type does not need model construction skills. Instead, the model user is an expert with regard to the system being studied and technical issues related to the study. For the model user, a PMS should help control the analysis of alternatives, help examine and draw conclusions from results, and assure that model validity is not compromised. In providing this kind of support, a PMS must support several facets of analysis control, including *analysis definition, analysis specification, alternative selection,* and *results evaluation.*

Analysis definition has to do with what information must be produced by the analysis, what analysis technique (simulation, optimization, etc.) to use to obtain the information, and what objects and which of their attributes in the overall system model are needed for the analysis. Analysis specification, on the other hand, is the detailed information needed by the selected analysis technique to perform the analysis. For example in a simulation, the run length and initial condition values are typical analysis specification data.

Alternative selection is the process by which each possible alternative is identified and prioritized. For example, traditional statistical analysis identifies all alternatives, using a standard design of experiments procedure, before any are analyzed. Once the alternatives have been established, they are evaluated using classical statistical methods. Another possibility is the incremental identification of alternatives. Results from one analysis are used to determine model parameter values of a new alternative. This new alternative may be generated and analyzed by using a response surface search procedure or a knowledge-based expert system.

Results evaluation includes statistical and other kinds of analyses based on values output from the models, the display of these values, and drawing inferences from these values. Result analysis includes estimating confidence intervals, associating distributions with output values, and relating model input values to model output values using meta-models. Other kinds of analysis, such as post-simulation performance measure definition and computation as discussed by Standridge and Tsai (1991), are possible. Methods for displaying results using animations, graphs, and reports are well known (Standridge, Hoffman, and Walker 1984, Healy 1985, Grant and Weiner 1986). The use of expert systems to draw inferences from analysis results has been widely discussed Klahr (1984), O'Keefe (1986), Reddy, Fox, and Hussain (1985), Shannon (1984, 1986), Shannon, Mayer, and Adelsberger (1985). Types of inferences include:

~ Identification of system conditions, for example:

IF work-in-process inventory > maximum allowed
THEN work-in-process inventory is too large

~ Diagnosis of problems, for example:

IF station N is idle AND
the queue of station N is empty and
station (N-1) is down
THEN station N is starved for work

~ Suggested problem resolution, for example:

IF station i is down AND
station k is up AND
station k can do the work of station i
THEN route all incoming jobs from station i to station k

~ Monitoring of proposed problem solutions, for example:

IF work-in-process inventory for alternative 2 <
work-in-process inventory for alternative 1
THEN work-in-process inventory has improved.

Validation ensures that the model user applies the model in such a way that valid results are obtained. The specification and computer implementation of a complete PMS validation system is a research activity. Input value boundary checking can be used to determine if individual values and combinations of values are within ranges for which the model is valid. Input values must be consistent with each other. Results may be statistically compared to known system values. Statistically significant differences would indicate an invalid application of the model.

3.3 Model Builders

The model builder enables the tasks of the model user and information transfer to the decision maker. The model builder constructs the singular model of the system of interest. An information-based approach for doing this is describe in Centeno (1990), Shannon and Centeno (1990), and in Centeno and Standridge (1991). The model consists of a set of objects that represent the components of system of interest. The attributes of an object tell how it may be used in different types of analysis, tell how it relates to other objects, and give distinguishing characteristics of that object.

In addition to constructing a model, the model builder must validate it using proven methods such as those discussed by Sargent (1988). For example, the model builder may compare the actual system with the model via an animation or compare model results with existing system data. The model builder supports the model user and the decision maker by providing defaults, bounds, and expert knowledge to support analysis control, result processing, validation, and information transfer tasks.

3.4 System Builders

A system builder specifies the scope of a PMS and enables the tasks of the model builder. The system builder defines the object and attribute classes used by the model builder to construct a model of a system of interest. The system builder provides expert system knowledge generic to all systems within the scope of the PMS to support analysis control, result processing, validation, and information transfer activities of the other user types. In an advanced PMS, the system builder provides knowledge concerning the issues and opportunities that the particular PMS can address to support the development of problem statements and analysis goals.

3.5 Tool Builders

Tool builders provide generic, "off-the-shelve" software tools and/or develop tools on an as needed basis. These tools must perform all of the computations and other operations needed in a PMS and help the system builders develop the PMS.

4 A GENERIC PMS ARCHITECTURE

Implementation architectures may vary widely among individual PMS systems according to the particular goals of the PMS. However, the general PMS characteristics discussed in section 2.0 and the general requirements of each of the five user types discussed in section 3.0 give rise to a generic modular architecture. Furthermore, each of the user types has a unique view of the PMS architecture as shown in Figures 1-4 and discussed below. In each figure, the knowledge/database (K/DB) contains information entered by other user types to enable the tasks of that particular user type. The information in the K/DB, that is entered by all user types, is processed by analysis tools that provide for statistical analysis, optimization, simulation, and so forth. Translation of the information from the K/DB organization to that required by the analysis tool is accomplished by translation software. Other software, such as inferential engines, may process the K/DB information as well.

4.1 Tool Builder's View

The tool builder sees a PMS as a set of functional requirements to be met with existing software if possible, or with newly developed software as needed.

4.2 System Builder's View

Figure 1 gives the system builder's view of the PMS architecture. The system builder populates a K/DB with information related to the classes of production systems the PMS will serve. Guidance for this task is provided by the problem definition knowledge in the K/DB as entered by the decision maker. Thus, the scope of the PMS is defined. This information includes definitions of the object classes used for model building as well as data and knowledge used throughout the analysis process.

4.3 Model Builder's View

Figure 2 gives the model builder's view of the architecture. The model builder populates the K/DB with information related to a particular production system, or perhaps more specifically current issues related to the design and/or operation of the system. Thus, one application of the PMS is defined. The model builder describes each relevant object in the system of interest based on the object class definitions provided by the system builder. In addition, the model builder embellishes the data and knowledge entered by the system builder to provide a production system specific context for the model user. Assisted by the validation processor, the model builder seeks to determine whether the object definitions and the analysis results are valid system representations.

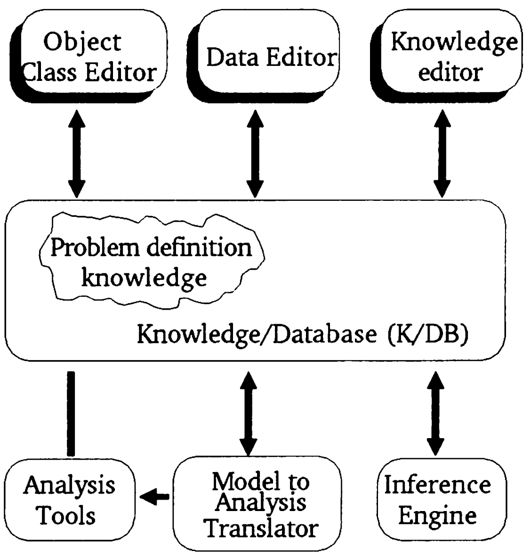

Figure 1: PMS - System Builder's View

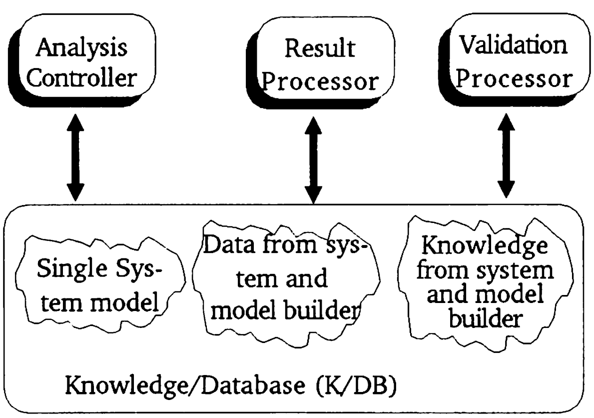

Figure 2: PMS - Model Builder's View

4.4 Model User's View

Figure 3 gives the model user's view of the architecture. The model user performs analysis based on the model of the system of interest, composed of object definitions provided by the model builder, as well as the data and knowledge entered by the system and model builders. The result processor helps the model user examine, interpret, and support decision making with the analysis results. The validation processor helps the model user apply the system model, data, and knowledge correctly.

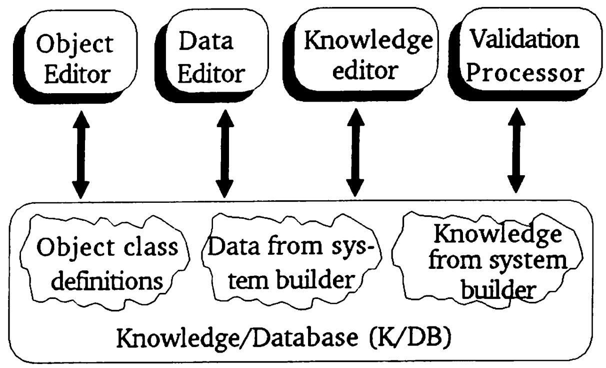

Figure 3: PMS - Model User's View

4.5 Decision Maker's View

Figure 4 gives the decision maker's view of the PMS architecture. The decision maker draws on a K/DB containing the system model and all analysis results as well as previously entered data and knowledge to generate reports and management presentations in support of decision making. An advanced PMS may support the problem definition tasks of the decision maker.

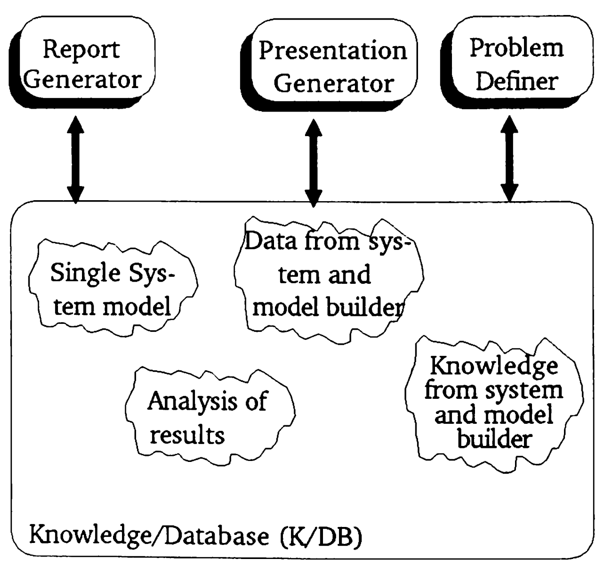

Figure 4: PMS - Decision Maker's View

5 A PMS FOR AN ADVANCED PRODUCTION AUTOMATION LABORATORY

Design is a critical aspect of engineering education. However, providing students with sufficient opportunities for realistic design experiences in an engineering discipline is a difficult task. Local industry may provide some student design experience opportunities. However, these opportunities may not be sufficient in number to serve all students, may not be timely with respect to current student courses, and will strictly depend on the current needs of the local industry. Furthermore, some engineering schools are in locations with little or no engineering-related local industry.

While other engineering disciplines have for some time provided a hardware-based laboratory environment for design experiences, Industrial Engineering was faced with unique difficulties in this regard. "A professor of industrial and systems engineering cannot bring a production system into a laboratory. It is too big, too complex. Furthermore, it involves humans who are not only unpredictable and difficult to measure, but who would also be unwilling to be part of a laboratory experiment... Obviously, much development work is urgently needed in better methods for teaching design principles to Industrial & System Engineering students." Turner, Mize, and Case (1987).

A completely automated production system design laboratory to provide near-realistic industrial design experiences is an ideal application of a PMS. One approach to the educational process in such a laboratory is as follows:

1. Students view a production process in the laboratory, examine the process, and ask questions.
2. Students construct a mathematical model of the process. Data needed to estimate model parameters is collected in the laboratory.
3. Students implement the model on a computer in the laboratory and validate it.
4. Students propose improvements to the process and assess the potential improvements using the model.
5. Students implement the improvements in the laboratory and verify that the process has indeed been improved.

Students iteratively perform these steps until improvement goals are achieved. A PMS integrates data collection, model building and execution, and process improvement verification.

To illustrate, consider the following typical case. During production, printed circuit boards exhibit significant queuing for a drilling operation. This operation must drill multiple holes on each board as a part of the board assembly operation. The number and location of the holes depends on the type of board. This operation is implemented in the laboratory. Faculty act as system builders who provide for automatic collection and storage of data concerning the current assembly operations and the object classes that describe the circuit board assembly system. In addition, the system builders set up links to simulation and optimization analysis tools.

Students act as model builders, model users, and decision makers. Model builders define printed circuit board system objects in such a way that optimization models and simulation models may be generated. Optimization models determine the sequence in which holes are drilled on a board. Simulation models assess the assembly operations in terms of queue lengths, equipment utilization, and throughput. Model users initially assess current drilling patterns and, based on modeling results, propose changes. Particular changes are selected for further assessment by the decision makers. These selected changes are tested by 1) telling the PMS to instruct the CIM software controlling the laboratory hardware to modify the corresponding part of the operation, and 2) gathering data on the performance of the modified system. This data is compared to data gathered from the previous drilling and assembly strategies as well as predictive data resulting from the models. Thus, the designed system changes can be proved or disproved.

6 SUMMARY

Designing a sophisticated, integrated production system is intrinsically a complex process. Computer models are needed to support this process. The effective use of these models can be enhanced through a modeling environments for developing and applying these models. A Production Modeling System (PMS) is such a modeling environment focused on the production systems design process. A PMS supports multiple types of users including those who make decisions, those who assess system alternatives using models, those who build models, those who configure, adapt, and tailor the PMS to the system or class of systems of interest, and those who build the software tools comprising the PMS. A PMS allows a singular description of a system from which several types of models can be generated using internal as well as external data. These models may be analyzed using multiple techniques, such as simulation or optimization, to answer a variety of design questions. Information generated by one analysis is transparently made available for use in the others. To accomplish its goals, a PMS integrates multiple analysis tools, databases, and knowledge bases.

REFERENCES

Centeno, M.A. 1990. Design of an integrated simulation modeling environment using a relational framework. Doctoral Dissertation, Department of Industrial Engineering, Texas A&M University, College Station, Texas.

Centeno, M.A. and R.E. Shannon. 1990. Expert simulation system based on a relational database. In *Proceedings of the 1990 Winter Simulation Conference*, eds. O. Balci, R. P. Sadowski, and R. E. Nance, 412-414. Institute of Electrical and Electronic Engineers, Piscataway, New Jersey.

Centeno, M.A. and C.R. Standridge. 1991. Modeling manufacturing systems: an information-based approach. In *Proceedings of the 24th Annual Symposium*, ed. A.H. Rutan, 230-239. Institute of Electrical and Electronic Engineers Press, Los Alamitos, California.

Grant, J.W. and S.A. Weiner. 1986. Factors to consider in choosing a graphically animated simulation system. *Industrial Engineering* **18(8)**: 36-40, 65-68.

Healy, K.J. 1985. CINEMA tutorial. In *Proceedings of the 1985 Winter Simulation Conference*, eds. D.T. Gantz, G.C. Blais, and S.L. Solomon, 94-97. Institute of Electrical and Electronic Engineers, Piscataway, New Jersey.

Klahr, P. 1984. Artificial intelligence approaches to simulation. In *Proceedings of the 1984 UKSC Conference on Computer Simulation*, 87-92.

O'Keefe, R. 1986. Simulation and expert systems - a taxonomy and some examples. *Simulation* **46(1)**: 10-16.

Reddy, Y., M. Fox and N. Hussain. 1985. Automating the analysis of simulation in KBS. In *AI, Graphics and Simulation*, ed. G. Birtwistle, 34-40. La Jolla, California: SCS Publications.

Sargent, R.G. 1988. A tutorial on validation and verification of simulation models. In *Proceedings of the 1988 Winter Simulation Conference*, eds. M.A Abrams, P.L. Haigh, and J.C. Comfort, 33-39. Institute of Electrical and Electronic Engineers, Piscataway, New Jersey.

Shannon, R.E. 1984. Artificial intelligence and simulation (keynote address). In *Proceedings of the 1984 Winter Simulation Conference*, eds. S. Sheppard, U.W. Pooch, and C.D. Pegden, 3-9. Institute of Electrical and Electronic Engineers, Piscataway, New Jersey.

Shannon, R.E., R. Mayer, and H.H. Adelsberger. 1985. Expert systems and simulation. *Simulation* **44(6)**: 275-284.

Shannon, R.E. 1986. Intelligent simulation environments. In *Intelligent Simulation Environments*, eds. P. A. Luke and H. H. Adelsberger, *Simulation Series* 17(1): 150-156. San Diego, California: SCS Publication.

Standridge, C.R., J.R. Hoffman, and S.A. Walker. 1984. Presenting simulation results with TESS graphics. In *Proceedings of the 1984 Winter Simulation Conference*, eds. S. Sheppard, U.W. Pooch, and C.D. Pegden, 51-58. Institute of Electrical and Electronic Engineers, Piscataway, New Jersey.

Standridge, C.R. and J. Tsai. 1991. A Method for Discrete Event Trace Processing. *Simulation*, Submitted.

Turner, W.C., J.H. Mize, and K.E. Case. 1987. *Introduction to industrial and systems engineering*. Englewood Cliffs, New Jersey: Prentice-Hall.

AUTHOR BIOGRAPHIES

CHARLES R. STANDRIDGE is an associate professor in the Department of Industrial Engineering at the Florida A&M / Florida State University College of Engineering. He led the development of the Simulation Data Language (SDL) and of The Extended Simulation Support System (TESS) for Pritsker Corporation. His current research interests are in the development of modeling and analysis environments for manufacturing systems, in the use of animation as a modeling and analysis tool, and in the analysis of health care delivery systems.

MARTHA A. CENTENO is an assistant professor in the Department of Industrial Engineering at the Florida A&M / Florida State University College of Engineering. She received a B.S. degree in Chemical Engineering from ITESO University (Mexico) in 1981, an M.S. degree in Industrial Engineering from Louisiana State University in 1985, and a Ph.D. degree in Industrial Engineering from Texas A&M University in 1990. Her research interests are in the areas of intelligent simulation environments and relational databases for simulation. Dr. Centeno is a member of Alpha Pi Mu, IIE, AAAI, ORSA, TIMS, SCS, and ACM.