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

INTEGRATED MODELING AND ANALYSIS GENERATOR ENVIRONMENT (IMAGE): A DECISION SUPPORT TOOL

Dursun Delen
Perakath C. Benjamin
Madhav Erraguntla

Knowledge Based Systems, Inc.
One KBSI Place
1408 University Drive East
College Station, Texas 77840, U.S.A.

ABSTRACT

A truly integrated modeling and analysis environment, which facilitates multi-use and multi-tool models, is necessary for today's enterprises to meet the challenges of the competitive global marketplace. In this paper, we present an ongoing research and software implementation effort that addresses these challenges by automatically generating simulation and optimization models using the multi-perspective-information captured in an enterprise model set. We start by providing a detailed analysis of the major roadblocks to the broader use of enterprise modeling and analysis methods in industry today. We then define a solution methodology and our technical approach to addressing these roadblocks. A brief discussion of current activities and future research directions concludes the paper.

1 INTRODUCTION

A key element to achieving success in today's extraordinarily competitive global market is skillful management. Organizations must continually monitor and control an increasing number of complex systems and situations, each with an expanding set of interdependent parameters and variables. To achieve this capability, organizations—from traditional manufacturing to software development, medical facilities, government agencies, and universities—require highly sophisticated enterprise modeling and analysis methods and tools.

The basic philosophy behind traditional approaches to enterprise modeling is to develop a model for a specific problem or situation (usually from scratch), apply it in solving the problem, and then discard it. Each problem or situation is viewed as unique, requiring a unique modeling and analysis effort. In general, so conventional thinking goes, a model created for one purpose or one tool cannot be used for *another* purpose or tool. Such an approach, in addition to being labor intensive, requires very specialized knowledge of the problem area and corresponding tool domains, necessitating the use of analysis experts and driving the cost of the effort even higher.

In this paper, we present an approach to enterprise modeling and analysis that has the potential to overcome many of these shortcomings. The basic philosophy of our approach is that modeling should be viewed as an ongoing process and not, as in the traditional approach, as a series of disjointed projects. Our approach is centered on the concept of an "enterprise model set": an integrated set of models that represent the current system from multiple perspectives in an accurate and detailed fashion. An enterprise model set evolves with the organization and is the basis for generating problem-specific and tool-specific execution models over the life cycle of the enterprise.

Reusable and multi-tool modeling is not new. A research team from Oklahoma State University has been conceptualizing, designing, and implementing a framework to enable multi-use, multi-tool models of manufacturing systems (Mize *et al.* 1993, Pratt *et al.* 1994, Kamath *et al.* 1995, and Delen *et al.* 1996b). The main theme of their approach uses the concept of a generic, persistent "base model" of the manufacturing system. This base model is continually maintained to provide a current, accurate, and detailed system representation. The base model is specific to the organization, and supports modeling across several problem domains using multiple analysis tools (Duse *et al.* 1993).

In Section 2 of this paper, we identify the roadblocks commonly found in building and using enterprise modeling and analysis tools to support managerial decision making. In Section 3, we present a solution strategy. In Section 4, we explain the details of our approach through the presentation of our solution framework. We conclude the paper in Section 5 by summarizing the current activities and identifying future research directions.

2 CHALLENGES IN ENTERPRISE ANALYSIS MODELS

Despite their well-recognized benefits, enterprise modeling and analysis methods remain largely unharnessed, and advances in enterprise analysis theories have yet to filter into the mainstream of managerial decision-making. The reason for this limited application is that these methods are generally very elaborate and require acute expertise to be used effectively. They operate on very intricate models of the enterprise being analyzed. Such models require specific formats and use technical jargon hardly comprehensible to non-experts.

In the process of modeling the world around us and accounting for all of its complexity, physicists have created powerful and sophisticated models. These models enable us to predict events, understand their impact, and manage and react more efficiently to the changes they generate. However, most people find it difficult to relate these complicated models to their own perception of the world. A similar situation has developed in the area of enterprise analysis. This field of science and engineering has made tremendous progress in its ability to answer questions, determine optimums, and weigh alternatives. However, it has done so by providing abstractions that are far removed from the systems they model and therefore are inaccessible to the average decision-maker. In physics, the path from our perception of the real world to intricate mathematical models contains intermediate abstractions; in enterprise analysis, there still is a significant gap between a decision-maker's perception of an enterprise and an executable model of that enterprise.

The dichotomy between the executable models created for analysis and the actual enterprises they model has promoted the impression that enterprise analysis is complex, time consuming, and prohibitively expensive. This perception is reinforced by the following characteristics of today's enterprise analysis efforts.

1. *Enterprise analysis efforts are analyst-dependent.* To produce executable models, most enterprise analysis methods rely heavily on an expert. Applying a particular analysis technique requires the abstraction and classification of enterprise concepts and elements into non-intuitive categories. In addition, enterprise analysis methods make use of specialized languages that demand a substantial amount of training to learn. For example, the building of an optimization model necessitates the expression of business rules and constraints with mathematical equations and the classification of the enterprise's elements and concepts into parameters and variables. Hence, most domain experts do not have the training necessary to generate and execute analysis models and instead must rely on experts in the various analysis fields.

2. *Enterprise analysis involves time- and communication-intensive activities.* The domain experts' dependence on experienced analysts to generate an executable model of a complex enterprise has made effective communication imperative. Domain experts possess in-depth knowledge of the enterprise to be analyzed. They understand the concepts underlying its functioning, the rules that constrain and govern its operations, and the interfaces and relationships among its components. Analysts, on the other hand, are experts in their particular analysis methods but typically have no understanding of the intricacies of an enterprise. Hence, the success of enterprise analysis depends on how well the domain expert can transfer his knowledge of the enterprise to the analyst and on how well the analyst can understand that enterprise, extract needed information, and design a valid executable model from that information.

3. *A significant amount of the effort spent is not reusable.* The knowledge transfer between a domain expert and an analyst is mostly an ad-hoc one. The analyst directs the activity that extracts the information and data needed to create a specific type of executable model. The analyst abstracts the domain expert's enterprise knowledge and directly encodes the resulting abstractions into mathematical formalisms and highly technical languages. It is seldom possible to reuse that knowledge in later analysis efforts of a different nature. Because knowledge transfer is one of the most critical and most time-consuming activities of an enterprise analysis effort, this situation is one that can greatly influence its cost.

4. *Decision-makers are not in control of the enterprise analysis effort.* The analysis models used to respond to a particular problem or to improve a certain aspect of an enterprise depend on the nature of the problem or on the desired improvement. The prevailing approach is to develop piecemeal custom models tailored to each specific decision-making situation. Hence, given a series of questions about a particular enterprise, an analyst may develop as many as five different models encoded in five different formalisms. This customization is often necessary because each analysis method is better suited to answer a particular type of question. Nevertheless, since there is no underlying representation of the knowledge from which the various models are obtained, there is no mechanism to help the domain expert interpret the results of the various analyses as a whole and to ascertain their

impact on the overall enterprise. Hence, each question or goal is answered in isolation from the rest of the analysis process, and the burden falls on the decision-maker to relate and integrate these independently obtained results.

These four characteristics are often viewed by decision-makers as significant, if not insurmountable, obstacles that are far too costly to overcome. Therefore, a major challenge to increasing the use of enterprise analysis methods in businesses and organizations is to provide the tools and methods that will address those obstacles and render analysis activities more attractive to all participants.

3 A PROPOSED SOLUTION

A solution methodology to overcome the four shortcomings described in the previous section is the development of a totally integrated modeling environment that supports the following features.

1. *Capturing the entire enterprise model set within a single application.* The term "enterprise model set" is used to refer to a group of conceptual models built to obtain a coherent and comprehensive picture of an enterprise. This set includes models of various types, and each type of model defines a perspective or viewpoint from which the system is considered for a given purpose, concentrating on some aspects and hiding irrelevant ones to reduce complexity. An enterprise model set may include various activity, process, organization, information, and behavioral models.

2. *True integration of various models.* Enterprise model sets have three critical characteristics. First, each type of model in a set is different *in nature* from any other model type. The second critical characteristic is that each model type plays an equally important role in describing an enterprise. Finally, the third major characteristic is that the models constituting the set are not independent from one another. Each model describes some aspect of the enterprise; each aspect is constrained by or related to other aspects of the enterprise described in other models. For example, the information captured in a data model may limit the execution of tasks described in a process model. The dependencies and relationships across models that constitute the enterprise model set can be handled through the integration of all model types. Hence, modeling components in various models that refers to the same real-world concepts will have a dynamic connection such that any changes to one will be propagated to the others.

3. *Having access to multiple analysis tools.* Once the system description is captured in the enterprise model set, the user (decision-maker) can analyze the whole or part of the system through various analysis techniques by configuring and translating one analysis model at a time. System analysis models may include simulation, optimization, queueing, and Petri nets.

4 TECHNICAL APPROACH

Our approach to overcoming the challenges and the shortcomings commonly found in today's enterprise modeling and analysis tools is the development of a truly integrated modeling and analysis generator environment (IMAGE) that supports the managerial decision making process in an accurate and timely fashion. The conceptual architecture of IMAGE is depicted in Figure 1. As it is illustrated in Figure 1, the major components of IMAGE are: (1) system description models, (2) system analysis models, (3) modeling knowledge base, (4) IMAGE database, and (5) graphical user interface.

The scenario of use starts from creating a rich representation of the enterprise in terms of the system description models (that is, the enterprise model set). This set may include models of various types, each of which may define a perspective or viewpoint from which the system may be considered for a given purpose. Such a multi-perspective approach to capturing the system description reduces the inherent complexity of each model by concentrating on some aspects and hiding irrelevant ones. Such an enterprise model set may include various activity, process, information, and ontology models. Once created, the enterprise model set serves as a rich repository of models of that enterprise.

Having such a rich repository of models creates the opportunity for automatically (or semi-automatically) configuring a variety of analysis and optimization models from the system description models. For instance, a simulation model (an analysis technique that is commonly used in studying the stochastic behavior of a system for a given goal) can be created from the information captured in the system description models. Simulation specific information can be added to the model during the configuration phase either automatically (with intelligently assigned default values) or manually (with values entered by the model builder). Based on the analysis results, the user (decision-maker) may repeat the process by changing the parameters (i.e., the goal or scope) or changing the analysis model type. This cyclic process can be repeated until a satisfactory decision result is obtained from the environment.

Such a rich set of system description models of the enterprise encourages decision-makers to use the analysis models in their decision-making processes. Obviously the modeling and analysis paradigm described above saves time and money through reuse, accuracy through the true integration of various descriptions capture models, and independence from analysis experts through the automatic generation of system analysis models. In the following section, the major software components of IMAGE are discussed in detail.

4.1 System Description Models

IDEF [Integrated Computer-Aided Manufacturing (ICAM) DEFinition] methods are used in constructing the system description models of the enterprise. In the following sub-sections, relevant IDEF methods are briefly introduced.

- **IDEFØ Function Modeling Method.** IDEFØ is a method designed to model the decisions, actions, and activities of an organization or system. IDEFØ was derived from a well-established graphical language, the Structured Analysis and Design Technique (SADT). IDEFØ is useful in establishing the scope of an analysis, especially for a functional analysis. As a powerful analysis method, IDEFØ assists the modeler in identifying what functions are performed, what is needed to perform those functions, what the current system does right, and what the current system does wrong. Thus, IDEFØ models are often created as one of the first tasks of a system modeling effort. A detailed explanation of the IDEFØ method can be found in KBSI (1994a).

- **IDEF1 Information Modeling Method.** IDEF1 was designed as a method for both analysis and communication in the establishment of requirements. IDEF1 is generally used to: 1) identify what information is currently managed in the organization, 2) determine which of the problems identified during the needs analysis are caused by lack of management of appropriate information, and 3) specify what information will be managed in the implementation. IDEF1 captures the information that exists about objects within the scope of an enterprise. The IDEF1 perspective of an information system includes not only the automated system components, but also non-automated objects such as people, filing cabinets, telephones, etc. IDEF1 was designed as a method for analyzing and clearly stating information resource management needs and requirements. A detailed explanation of the IDEF1 method can be found in KBSI (1994b).

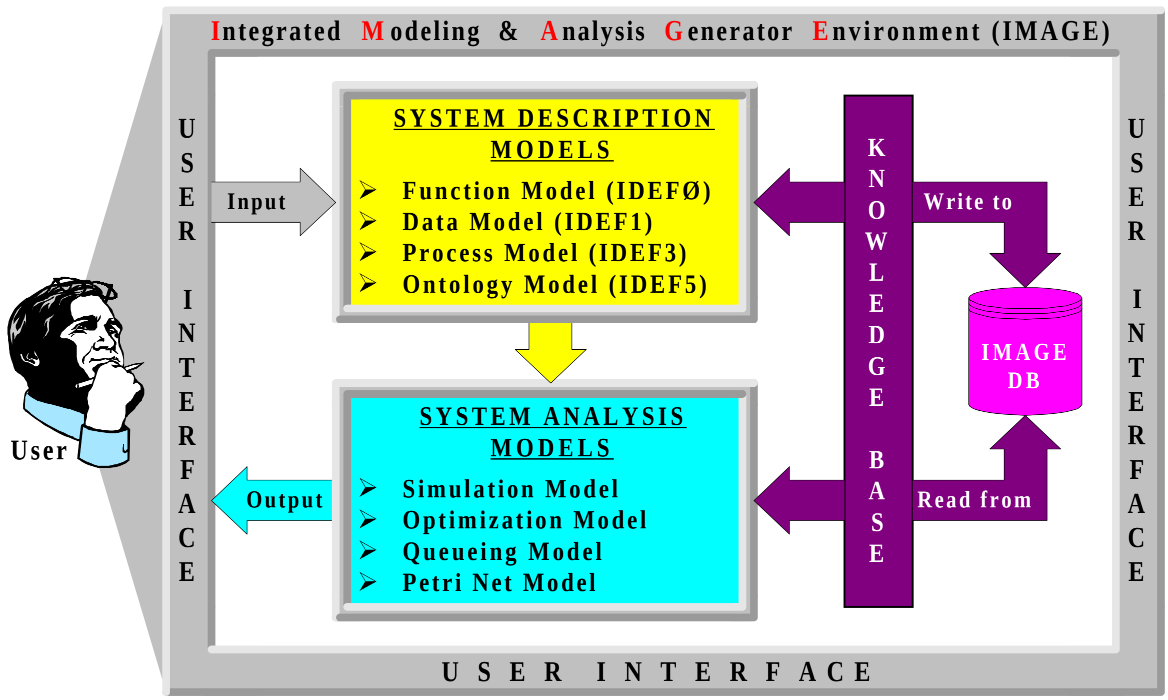

Figure 1: IMAGE Framework

- **IDEF3 Process Description Capture Method.** The IDEF3 Process Description Capture Method provides a mechanism for collecting and documenting processes. IDEF3 captures precedence and causality relations between situations and events in a form natural to domain experts by providing a structured method for expressing knowledge about how a system, process, or organization works. There are two IDEF3 description modes: process flow and object state transition network. A process flow description captures knowledge of “how things work” in an organization; e.g., the description of what happens to a part as it flows through a sequence of manufacturing processes. The object state transition network description summarizes the allowable transitions an object may undergo throughout a particular process. Both the Process Flow Description and Object State Transition Description contain units of information that make up the system description. These model entities form the basic units of an IDEF3 description. The resulting diagrams and text comprise what is termed a “description” as opposed to the focus of what is produced by the other IDEF methods whose product is a “model.” A detailed explanation of the IDEF3 method can be found in KBSI (1995).

- **IDEF5 Ontology Description Capture Method**. Historically, ontologies arose from the branch of philosophy known as metaphysics, which deals with the nature of reality—of what exists. The traditional goal of ontological inquiry, in particular, is to divide the world “at its joints”: to discover those fundamental categories or kinds that define the objects of the world. The IDEF5 method provides a theoretically and empirically well-grounded method specifically designed to assist in creating, modifying, and maintaining ontologies. Standardized procedures, the ability to represent ontology information in an intuitive and natural form, and higher quality results enabled through IDEF5 application also serve to reduce the cost of these activities. A detailed explanation of the IDEF5 method can be found in KBSI (1994c).

4.2 System Analysis Models

In the current implementation, the system analysis model set includes simulation and optimization. Simulation is known to be a modeling method that seeks to “duplicate” the behavior of the system under investigation by studying the interactions among its components. The outcome of a simulation model is normally presented in terms of selected measures that reflect the performance of the system. As we’ve described, our implementation automatically generates a simulation model from the enterprise model set; this process has two major steps (see Figure 2). The first step is to configure a generic simulation model using the system specific information captured in the enterprise model set. The second step is to translate this generic simulation model to a solver-specific execution model. The intermediate representation (referred to as the generic simulation model) enables us to use multiple commercial simulation solvers (e.g., WITNESS, ProModel, ARENA, and SlamSystem) based on the problem being answered. More detailed explanations of this multi-tool approach can be found in Delen *et al.*, (1996a and 1996b).

Optimization is a mathematical modeling technique that seeks the best (optimum) solution under the restriction of limited resources for a given problem. Our implementation automatically generates an optimization model from the enterprise model set; this process has two major steps. The first step is to configure a generic optimization model using the system specific information captured in the enterprise model set along with the predefined optimization related knowledge capturing templates. The second step is to translate this generic optimization model to a solver specific execution model. Again, the intermediate representation (referred to as the generic optimization model) enables us to use multiple target optimization engines based on the capabilities of the tools and the requirements of problem being answered.

Besides simulation and optimization, we intend to add two more analysis tools: queuing network modeling and Petri net modeling. Queueing network models are probably one of the most commonly used mathematical

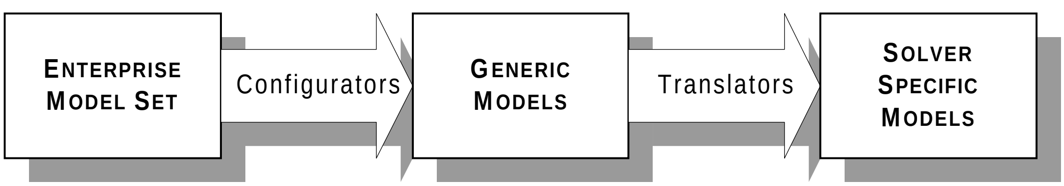

Figure 2: Enabling Multi-Tool Capability Through Configurators and Translators

modeling technique to estimate the performance measures of a stochastic system, especially for the analyses that must be completed in a relatively short period of time (Suri and de Treville 1993). Petri nets area relatively new yet powerful modeling and analysis techniques. They are well suited to modeling concurrency, synchronization and conflict, and hence are a natural choice for modeling complex systems (Viswanadham and Narahari 1992). They are known to be superior in providing qualitative performance measures including absence/presence of deadlock, reinitializability, and buffer overflows along with the classical quantitative performance measures.

4.3 Modeling Knowledge Base

Our current implementation of IMAGE uses a powerful production rule system built upon a modeling knowledge base to provide total integration between the different models of an enterprise. There seem to be several advantages to using a rule-based engine as the foundation for model integration. Rules for maintaining consistency between models and model elements do not need to be coded into the program but instead are explicitly captured and stored separately in the knowledge base. Because the rules are not represented in the source code, they can be changed without rebuilding the application. This means that they can be changed without intervention or support from a developer. This flexibility allows the user to customize the behavior of the model integration mechanisms to better fit their specific domain and needs. In the current implementation, only the basic rules are used to provide the behavior for keeping the various complementary perspectives consistent with each other. The rationale for providing only these basic rules, besides the time constraints, is to provide robust model integration while keeping users from being overwhelmed with unexpected or unnecessarily complex tool behavior.

Note that giving the tool the ability to maintain and manipulate meta information (i.e., the information *types* supported by the various modeling methods) will allows users to work at a higher, more generic level of abstraction. At this meta level, users will be able to make connections and define relationships between model and model element types. This unique feature is expected to greatly increase the flexibility and the power of integrating models to maintain consistency among the various enterprise perspectives, focuses, and levels of detail.

The use of a rule-based engine to maintain consistency and propagate changes across models is made possible by encoding all of the information contained in the enterprise model set (as well as other types of information) in the integrated information base. The set of model integration rules, together with the integrated information base, will form a knowledge base that can be used by IMAGE to provide expert, system-like functionality. When changes are made to some part of the information base, the expert system will be invoked and the rules will be evaluated and applied as appropriate. In this manner, the various aspects of the enterprise will be synchronized and remain consistent without any cumbersome effort on the part of the user.

To maintain consistency across the various aspects of an enterprise, the IMAGE expert system will require users to identify relationships between model elements across models. To this end, and through simple drag-and-drop user gestures, it lets users designate associations between model elements displayed on the screen. Once such associations have been explicitly captured, consistency across models is maintained automatically by the firing of rules in the knowledge base. To simplify the model integration task, IMAGE provides users with a set of predefined relationships that are used to relate model elements in a meaningful way.

4.4 IMAGE Data Base

IMAGE will provide a simple programming interface to be used by all model builders to store information in the integrated relational database. Each model builder will describe the objects that are to be stored in the database through the simple programming interface. The database manager will use this description to create the necessary tables in the database and to store and retrieve data as needed. As new components are added to the environment, their data will be integrated seamlessly into the existing database. The new component's objects will be easily linked to the other model elements through the use of explicitly defined objectified relations that are stored together with the objects in the database. The addition of a new model builder to IMAGE thereby expands the environment to include more information and functionality with little extra effort on the part of the developer. New components added to the framework will not require those components to use their own particular storage mechanism but rather rely on the one provided by IMAGE. When a component needs an object, the component requests the object through the database interface. The database uses its registered description for the object to find it in the relational database. This registration of object descriptions will ensure that the information created using one of the IMAGE components is accessible by the other components or plug-in applications.

This feature is critical to support major advances in enterprise technology. In particular, it will provide the means to extend and customize the environment to support domain-specific information types. Using this feature, each enterprise will be able to customize the environment to support their particular modeling needs.

4.5 Graphical User Interfaces

The graphical user interface of IMAGE is the means to enter system description models as well as making use of what has been captured in the enterprise model set through system analysis models. In doing so, the main objective has been to make the user interface as intuitive and automatic as possible, thus focusing the user on issues of modeling rather than interaction with the system. The user interface should also follow, as closely as possible, the users' mental model of the system and of the data being manipulated to make interfacing with the environment logical and consistent.

5 SUMMARY AND FURTHER RESEARCH DIRECTIONS

In this paper, we have provided an analysis of the challenges facing the broader use of enterprise modeling and analysis techniques and presented an approach that has the potential to address these challenges. Our proposed approach provides an integrated modeling environment that supports: (1) the development of all types of models to capture the various aspects of an enterprise, (2) the seamless integration of these models and the use of inter-model relationships to automate consistency maintenance across models, and (3) multi-tool functionality extensions to the environment.

The current state of our implementation includes (1) the fully developed modules to specify the enterprise model set, (2) the knowledge bases for integration rules, (3) modules to automatically generate optimization and simulation models via the use of the enterprise model set, and (4) a state-of-the-art GUI that facilitates the creation and utilization of IMAGE.

Our future plans for IMAGE includes adding more and more tools for enhanced enterprise analysis. In the short-term, this includes adding Queueing, Petri Net, and Cost Analysis modules. The natural extension to IMAGE's rich enterprise representation and multi-tool capabilities is an intelligent, automated advisor. This automated advisor would aid users in defining a problem from a given set of symptoms and identifying the best tool to use for the defined problem. The use of knowledge-based technology in the current IMAGE framework, along with Dr. Delen's expertise in expert systems (Delen 1997), will greatly facilitate the implementation of the intelligent advisor.

ACKNOWLEDGMENTS

The research described in this paper was partly supported by NASA Kennedy Space Center through the SBIR Optimization Modeling Assistant (OMA) project, Contract No NAS10-97002. This work could not have progressed to its current level had it not been for the significant contributions of Ms. Florence Tissot and Mr. Wes Crump whose efforts we gratefully acknowledged.

REFERENCES

Delen, D., D. Pratt, and M. Kamath. 1996a. A New Paradigm for Modeling of Complex Manufacturing Systems. In *Proceedings of the First Turkish Symposium on Intelligent Manufacturing Systems*, ed. M. Ozbayrak, H. Taskin, E. Oztemel, 27-37. Adapazari, Turkey: Sakarya University Press.

Delen, D., D. Pratt, and M. Kamath 1996b. A New Paradigm for Manufacturing Enterprise Modeling: Reusable, Multi-Tool Modeling. In *Proceedings of the 1996 Winter Simulation Conference*, ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Skinner, 995-992. Picataway, New Jersey: Institute of Electrical and Electronics Engineering.

Delen, D. 1997. *Development of an Expert System Based Experimental Frame for Modeling of Manufacturing Systems*. Ph.D. Dissertation, Oklahoma State University, Stillwater, OK.

Duse, M., Gharpure, J., Bhuskute, H., Kamath, M., Pratt, D. B., and Mize, J. H. 1993. Tool-independent model representation. In *Proceedings of the 2nd Industrial Engineering Research Conference,* Norcross, GA, 700-704. IIE.

Kamath, M., D. B. Pratt, and J. H. Mize. 1995. A Comprehensive Modeling and Analysis Environment for Manufacturing Systems. In *Proceedings of the 4th Industrial Engineering Research Conference*, Nashville, TN, 759-768. Institute of Industrial Engineers.

Knowledge Based Systems, Inc. (KBSI). 1994a. *IDEFØ Function Modeling Manual*. A reconstruction of the original Air Force Wright Aeronautical Laboratory Technical Report AFWAL-TR-81-4023, College Station, TX.

Knowledge Based Systems, Inc. (KBSI). 1994b. *IDEF1 Information Modeling Manual.* A reconstruction of the original Air Force Wright Aeronautical Laboratory Technical Report developed under Air Force contract AF33615-80-C-5155, College Station, TX.

Knowledge Based Systems, Inc. (KBSI). 1994c. *IDEF5 Ontology Description Capture Method Report.* Information Integration for Concurrent Engineering (IICE), College Station, TX.

Knowledge Based Systems, Inc. (KBSI). 1995. *IDEF3 Process Description Capture Method Report.* Information Integration for Concurrent Engineering (IICE), College Station, TX.

Mize, J. H., H. C. Bhuskute, D. B. Pratt, and M. Kamath. 1992. Modeling of Integrated Manufacturing Systems Using an Object-Oriented Approach. In *IIE Transactions*, 24, 3, 14-26.

Pratt, D. B., P. Farrington, C. Basnet, H. Bhuskute, M. Kamath, and J. H. Mize. 1994. The Separation of Physical, Information, and Control Elements for Facilitating Reusability in Simulation Modeling. In *International Journal in Computer Simulation*, 4, 327-342.

Suri, R. and S. de Treville. 1993. Rapid Modeling: The Use of Queueing Models to Support Time-Based Competitive Manufacturing. In *Operations Research in Production Planning and Control*, Springer-Verlag, 21-30.

Viswanadham, N. and Y. Narahari. 1992. *Performance Modeling of Automated Manufacturing Systems.* Prentice-Hall, Inc., Englewood Cliffs, NJ.

BIOGRAPHICAL SKETCHES

DURSUN DELEN is a Research Associate at Knowledge Based Systems, Inc., College Station, Texas. He received his Bachelor of Science degree in Industrial Engineering from the Istanbul Technical University, Istanbul, Turkey, in 1986; his Master's degree in Industrial Engineering from the Yildiz University, Istanbul, Turkey, in 1988; and his Ph.D. in Industrial Engineering and Management from Oklahoma State University, Stillwater, Oklahoma, in 1997. He has more than five years of industrial experience in information systems analysis and design. His research interests include systems modeling, discrete event simulation, object-oriented modeling, knowledge representation and artificial intelligence.

PERAKATH C. BENJAMIN is the Vice President of Research at Knowledge Based Systems, Inc., College Station, Texas. He received his Master’s degree in Industrial Engineering from the National Institute for Training in 1983 and his Ph.D. in Industrial Engineering from Texas A & M University in 1991. He has over 12 years of professional experience in systems analysis, design, development, testing, documentation, deployment and training.

MADHAV ERRAGUNTLA is a Research Scientist at Knowledge Based Systems, Inc., College Station, Texas. He received his Master’s degree in Industrial Engineering from the National Institute for Training in 1989 and his Ph.D. in Industrial Engineering from Texas A & M University in 1996. His research interests are knowledge representation and reasoning, simulation, planing, qualitative reasoning, manufacturing, and operation research.