Proceedings of the 1997 Winter Simulation Conference
ed. S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson

STRUCTURED MODELS AND DYNAMIC SYSTEMS ANALYSIS: THE INTEGRATION OF THE IDEF0/IDEF3 MODELING METHODS AND DISCRETE EVENT SIMULATION

Larry Whitman
Brian Huff

Automation & Robotics Research Institute
The University of Texas at Arlington
7300 Jack Newell Boulevard South
Fort Worth, Texas 76118, U.S.A.

Adrien Presley

Division of Business and Accountancy
Truman State University
100 East Normal
Kirksville, Missouri 63501, U.S.A.

ABSTRACT

The role of modeling and simulation is receiving much press of late. However, the lack of practice in employing a link between the two is alarming. A static model is used to understand an enterprise or a system, and simulation is used for dynamic analysis. Generally, most models are considered static, whereas simulation is really a dynamic model. Static models are useful in achieving understanding of the enterprise. Simulations are useful in analyzing the behavior of the enterprise.

Most enterprises develop and even maintain multiple types of models for different purposes. If a single model can be used to drive other modeling purposes, then model maintenance and development could be reduced. This paper describes the procedure necessary to use a static representation as the primary input for an animated simulation. It presents the additional steps necessary to annotate a static model for input to a dynamic model. Two commercial suites, WorkFlow Modeler™ to ServiceModel™ and ProSim™ to WITNESS™, are compared and contrasted based on the respective ease of conversion from the static model to the dynamic model. Any user who purchases these products can follow the steps described in this paper for either of these product suites to generate a simulation from a static model. Finally, some general observations of using an existing IDEF (0 or 3) model to create a working simulation are presented along with conclusions.

1 INTRODUCTION

The commonly accepted definition is that a model is a representation of reality. Generally, details that are unnecessary are not included. The typical uses (Nathan and Wood 1991) (Snodgrass 1993) (Reimann and Sarkis 1996) of modeling are:

- To analyze and design the enterprise and its processes prior to implementation
- To help reduce complexity
- To communicate a common understanding of the system
- To gain stakeholder buy-in
- To act as a documentation tool for ISO 9000, TQM, Concurrent Engineering, and other efforts.

A primary thrust of this research is to determine the feasibility of using a single master static model of the enterprise for multiple purposes. Previous research has presented a single suite perspective (Lingineni, Caraway et al. 1996) and uses custom developed software (Harrell and Field 1996). This research uses commercial products exclusively. For the static model, the research uses two specific methods for static representation, IDEF0 and IDEF3. Two concerns were: (1) the amount of required change of the IDEF methodology, and (2) the amount of additional annotation to the IDEF model required by the IDEF tool. Further explanation of these concerns follows.

IDEF is a rigorous methodology. The reason for the rigor is to ensure a robust and complete representation. As part of this rigor, a thorough review process is used. The review cycle is enhanced by the rigid IDEF syntax. The syntax for IDEF is very explicit. A concern in utilizing an IDEF model to create a simulation is that the IDEF method would have to be compromised to enable the creation of a simulation directly from the IDEF model. However, it should also be noted that there are certain characteristics of IDEF modeling that have become considered standard practice, yet they are not a strict IDEF syntactic rule. This research found that the only changes to the methodology required were in these time-honored traditions, whereas no actual IDEF rule of syntax was broken.

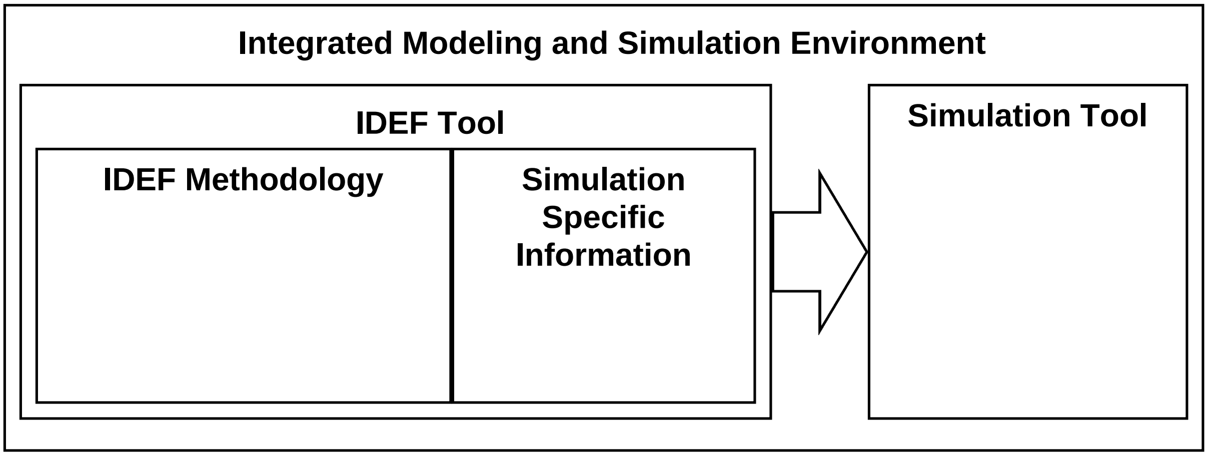

Figure 1: Integrated Modeling and Simulation Environment

Another concern was the amount of additional annotation required by the tool to drive the simulation. Figure 1 shows how the IDEF methodology, the IDEF tool, and the simulation data interact. Note the distinction between the IDEF methodology and the IDEF tool. The tool is an implementation of the methodology itself. Tools sometimes have limitations which the methodology does not require. The opposite is also true in that the methodology sometimes restricts a certain characteristic, yet the tool does not restrict this feature. Therefore, it is important to note the differences between the limitations of methodologies and tools. A primary research task was to validate the need for additional population of data inside the simulation itself. The goal is to eliminate the need for the user to add *any* information inside the simulation tool. This will facilitate the master model concept discussed later.

The paper will first discuss the types of models and explain the differences between static and dynamic models. Next, we will provide an overview of the IDEF methodology that includes an explanation of the tools used. Then, the static model creation process is discussed, including a description of the sample process used and the process of creating the static models. A description is then given of the method used to convert the static model to a dynamic model. Some general observations are made and future directions are then presented.

2 TYPES OF MODELS

In this section, we discuss the two types of models: static and dynamic, and explain the five different views of a model.

2.1 Static

Static models attempt to provide a static representation of dynamic systems. Static models generally portray the possible flow paths of objects through a system. This information is very helpful in determining what items participate in the process and the functions performed by the system. Although static representations can indicate the allowable system behaviors, they cannot depict the range of time-variant behavior generated as a result of resource availability or the number of items flowing through the process. To adequately predict the performance characteristics of dynamic systems, the time-variant behaviors of the system must be able to be defined and represented.

2.2 Dynamic

Dynamic representations of systems attempt to capture and describe the behavior of the system over time under different operating conditions. For the purposes of this paper, we are referring to discrete-event simulation as the dynamic system model. Although the static system representations are capable of providing the vast majority of the information needed to construct a dynamic systems model, they do not possess the mechanisms needed to enact the process behavior constraints defined in their representations. Discrete-event simulation tools, in contrast, are capable of executing sets of system behavior roles and tracking the system’s transition through a series of states. In this manner, a dynamic model can provide information about the state of the system at a given instance in time or can generate performance measures of the system over a given period of time. Dynamic models can be used iteratively to study system behavior under different operating conditions. Subtle changes in resource availability or system loading can have dramatic effects on the performance of the system. This range of potential behaviors is very difficult to represent with a static system model.

2.3 Views

Previous work in the development of architectures by the Automation & Robotics Research Institute (Presley, Huff, Liles 1993) describes a five-view approach. The Business Rule (or Information) View defines the entities managed by the enterprise and the rules governing their relationships and interactions. The Activity View defines the functions performed by the enterprise (*what* is done) while the Business Process View defines a time-sequenced set of processes (*how* it is done). The resources and capabilities managed by the enterprise are defined in a Resource View. Finally, the Organization View is used to define how the enterprise organizes itself and the set of constraints and rules governing how it manages itself and its processes.

This does not, however, mean that all these views must be present in all models. A model is an abstract representation of reality which should exclude details of the world which are not of interest to the modeler or the ultimate users of the model. Models are developed to answer specific questions about the enterprise. This research focuses specifically on the need for analysis of resource constraints and process flows.

3 OVERVIEW OF IDEF MODELS

IDEF (Integration DEFinition) was developed by the U.S. Air Force's Integrated Computer Aided Manufacturing (ICAM) project in the late 1980's. There are many different IDEF methods. Each method is useful for describing a particular perspective of an enterprise. The major IDEF methods in use are functional or activity modeling (IDEF0), information modeling (IDEF1), data modeling (IDEF1x), process description capture (IDEF3), object oriented design (IDEF4), and ontology capture (IDEF5) (Mayer, Painter, deWitte 1992). Although IDEF2 was intended to be used as a dynamic modeling method for simulation, the numerous simulation tools commercially available have supplanted this method. The modeling methodologies used in this paper are IDEF0 and IDEF3. These two methods best facilitate a structured approach to system model development and review and the creation of a corresponding discrete-event simulation of the system. Both of these methods utilize a subordinate principle of abstraction called decomposition (Rumbaugh et al. 1991), which is the breaking down of each box (activity) into more detail in a continuous manner until the greatest level of detail is achieved. (Marca and McGowan 1988)

3.1 IDEF0

There are five elements in the IDEF0 functional model as shown in Figure 2. The activity (or function) is represented by the boxes; inputs are represented by the arrows flowing into the left hand side of an activity box; outputs are represented by arrows flowing out the right hand side of an activity box; the arrows flowing into the top portion of the box represent constraints or controls on the activities; and the final element represented by arrows flowing into the bottom of the activity box are the mechanisms that carry out the activity (Marca and McGowan 1988, Mayer 1992).

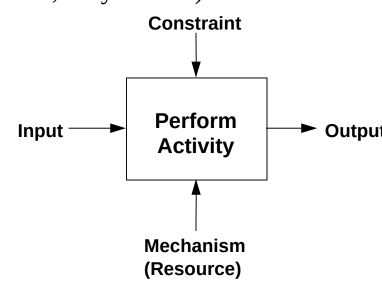

Figure 2: IDEF0 Nomenclature

3.2 IDEF3

The IDEF3 Process Description Capture Method (Mayer, Painter, deWitte 1992) consists of process flow diagrams and elaboration diagrams. Only the process flow diagrams were used to feed the simulation model. IDEF3 uses a rigid syntax that eliminates model ambiguity. The basic elements of IDEF3 process descriptions used in this research are Unit of Behaviors (UOBs), Junctions, and Links. A Unit of Behavior describes the actual process detailed in the box. Links connect the boxes and describe the relationship between the various UOBs. Junctions explicitly describe the logic of multiple links either coming together or spreading apart. Two examples of junctions are decision points or entities branching into parallel flows of process steps. An example of an IDEF3 model with UOBs, links and various types of junctions is shown in Figure 3.

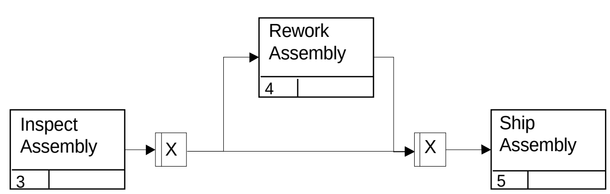

Figure 3: IDEF3 Example Model

3.3 Modeling Tools

The IDEF0 tool used was WorkFlow Modeler™ version 3.7MR2 by Meta Software Corporation (MetaSoftware 1997). This tool was formerly named Design/IDEF and integrates with ProModel Corporation's ServiceModel™ Simulation Software version 3.01 (ProModel Corporation 1996).

The IDEF3 tool used was ProSim™ version 2.2 by Knowledge Based Systems Incorporated (Knowledge Based Systems 1996). This tool integrates with WITNESS™ Simulation Software version 7.40. WITNESS™ is a registered trademark of the Lanner Group Incorporated.

4 STATIC MODEL CREATION PROCESS

In this section, we describe the actual sample process used, the task of creating the static model, and the additional information necessary for generating a simulation input model.

4.1 Process Used

The process used in this research was that of "Performing Production Control Activities" which was developed as part of a consensus model for Operating a Manufacturing Enterprise (Automation & Robotics Research Institute 1991). The consensus model describes the generic processes performed by most manufacturers. Process experts at several companies specified the model in a pre-released version of IDEF3 through a series of reviews. The Performing Production Control Activities diagram was the lowest decomposition available that was chosen for simplicity. This process contains only eleven activities, two decision points, and one input with three outputs. Two resources were used and the constraints were ignored.

4.2 Static Model Creation

A generic sketch was created of the process previously described. The sketch simply outlined the flow required to accomplish the overall task of production control activities. This sketch was used as a guideline to create the two IDEF models using their respective tools. For IDEF0, there were some arbitrary modifications necessary to make the model comply with the IDEF0 method primarily related to converting the model from IDEF3 to IDEF0. For IDEF3, the data easily complied with the method. The authors reviewed the diagrams for syntax and completeness.

4.3 Additional Model Annotation

A primary focus of this research was to determine the additional requirements in annotating a static model as an initial input to simulation. This section addresses these issues.

IDEF0 (WorkFlow Modeler™ – ServiceModel™) – There are six steps in annotating an IDEF0 model for simulation. The first step is to define any entity subtypes. This is required for any branching of the entities during the process. The next step is to place these subtypes on the appropriate leaf. Next, the actual process times are placed in the model using the Glossary:Cost Info option. However, the distribution times had to be keyed in using the exact distribution syntax times (T(15,18,23,1)). This is planned to be remedied in a later version. The process times could also be assigned transmission times and priorities. The resources were then added in a straightforward manner. The general simulation attributes are then established. A check can then be made to ensure the capability to output to simulation. This check had a useful tool to place the user in the model directly where the problem existed. Finally, the model is output to a file to be read by the simulation package. Also, a graphical representation of the appearance of the simulation is shown in WorkFlow Modeler™. The WorkFlow Modeler™ syntax check that ensures there are no problems when converting to simulation is very rigid. As an example, any apostrophes are not allowed for activities such as mat'l.

IDEF3 (ProSim™ – WITNESS™) – The ProSim™ tool has a handy option to create the defaults for resource, location, queue, entity, and process information as well as setting the version of WITNESS™ to transfer to and the method of displaying the entities and the icon styles (IDEF3 notation or not). The user must then add the required entities, resources, and locations using the IDEF3 pools as objects. The entities can have several description types, which allows the user to have entities being affected, created, or destroyed during the simulation. Unfortunately, this appears to be a global characteristic. Typically, an entity is created at one point in the simulation and can then be an agent, affected or other description types. Resources can be agents or participants and the number of the particular resource can be modified. Locations can be set for buffer information, set-up and breakdown details. To add the simulation information to a particular UOB, the user right clicks on the UOB and selects "Edit Simulation Info". Then the user adds the process time by entity and can build the required distribution. For example, when a triangle function is chosen, a sample distribution is entered and the user edits the parameters. The resource and location information is added in the same manner.

The check provided by ProSim™ allows the user to designate which characteristics to verify. The syntax check function doesn't appear to do well when assigning multiple tasks to the same location.

5 CONVERSION TO DYNAMIC MODELS

Another key concern of this research was the perceived difficulty in the actual conversion process. If the average user found the process of converting the IDEF model to simulation difficult, the use of this method would be minimal. In fact, the process necessary in converting from one tool to the other was quite simple.

In WorkFlow Modeler™, the user simply selects the option to output the ServiceModel™, indicates the file name and the export process occurs. Then in ServiceModel™, the user opens this file and the conversion process is complete. There are no additional tasks in ServiceModel™; therefore, the static IDEF0 model may be used as the master.

In ProSim™, the user selects Build Simulation Model, indicates if the entire model with all levels of decomposition are to be translated to the simulation or only the current diagram. Then the user indicates the file name and the export process occurs. Then in WITNESS™, the file is read as a control file and the conversion process is complete. There are no additional tasks in WITNESS™; therefore, the static IDEF3 model may be used as the master.

Both suites make the actual conversion process simple and straightforward. The task of converting the static model to the simulation tool in the view of the researchers is a non-issue.

6 OBSERVATIONS

Several interesting items were observed during this research. The primary items of note were: model review, the concept of a master model, the case for bottom-up modeling, advancement of technology tools, user instructions and interface, and decomposition.

6.1 Model Review

A major advantage of using a static model is the ease of review. The IDEF methodology is a rigid methodology that uses a thorough review process. This process is to ensure completeness and accuracy of models. Most enterprise personnel can review and comment in a structured model review cycle as in IDEF0 or IDEF3. Several experts are used in the reader review cycle in an iterative process until a correct and complete model is achieved. Our past experience has shown a poor track record of factory personnel judging the accuracy of even a snapshot of a simulation model. A static model is simpler to review than an animated simulation.

6.2 Master Models

As previously stated, a goal of this research is to arrive at a "master enterprise model" that serves multiple purposes. Frequently, separate efforts create different views of the enterprise by different modeling techniques. Both static and dynamic models serve a valid purpose. However, if a static model can be used to drive a dynamic model, then the static model can serve as the master model and version control can be kept on a single model. Therefore, the cost of modeling an enterprise can be reduced and yet the advantage of the different types of models may still be achieved.

6.3 Bottom-Up Modeling

There are two methods of developing models: top-down and bottom-up. Most IDEF methods portend a top-down approach. For enterprise-wide analysis this is clearly the optimal approach. However, for a simulation, a top-down approach does not lend itself well to gathering the appropriate information in an efficient manner. The data necessary for simulation is generally at the lowest level of decomposition and the IDEF0 method forces some arbitrariness in the mid-level diagrams. By using a bottom-up approach, an accurate representation can be reached quickly. However, caution should be used in ensuring that a myopic approach is not taken and a suboptimal enterprise-wide solution is implemented. More information on bottom-up modeling can be found in Pratt (Pratt, Mize, Kamath 1993).

6.4 Advancement of Technology Tools

As technology moves forward, the ease of conversion between tools in a given suite should increase. Simply because a tool doesn't perform as expected in the past does not necessarily indicate that it still doesn't. Our research met with difficulties with both modeling tools that were eliminated with the later versions of each tool. In WorkFlow Modeler™, the syntax check did not display a particular error that was preventing the simulation from working. After, installing the latest version, the syntax problem was correctly displayed. In ProSim™, the simulation portion of the software was extremely difficult to use. After installing the latest version, the user interface was very intuitive. Therefore, the acquisition of the latest version of each tool in the suite is highly recommended.

6.5 Interface and User Instructions

A key aspect in using a static modeling tool to drive simulation is that the user interface must be as simple as the driven simulation tool. If it is more difficult to enter the parameters necessary in the static tool than in the simulation, the user will naturally avoid the static tool. In both tools, it was easier to enter the parameters in the static tool than in the dynamic tool. WorkFlow Modeler™ was more difficult to use than was ProSim™. WorkFlow Modeler™ had a fairly good manual regarding the additional annotation of IDEF0 to support simulation, whereas, there was little if any documentation for ProSim™ on these same steps. On the other hand, the ProSim™ requirements for additional information were fairly straightforward.

6.6 Decomposition

One characteristic of the IDEF0 modeling technique that was not directly addressed in this research is that each activity and the arrows can, and in most cases must be, decomposed (or exploded) into more detailed levels of analysis. This characteristic is especially useful in enterprise modeling where details about lower level activities can be captured, but at the same time, be hidden from models of the enterprise at higher, more abstract levels. This can be thought of as equivalent to the development of hierarchical simulation models. The IDEF3 method does not explicitly limit the number of activities which can exist within a given process model. The process modeled for this paper could easily be represented within one level of abstraction, and as a result the IDEF3 model did not contain any decompositions. The IDEF0 method did require multiple decompositions of the system model. Future research will involve the creation of larger system models that will force the study of the ability to support hierarchical decompositions in both the IDEF0 and IDEF3 methods.

7 CONCLUSION AND FUTURE DIRECTIONS

This paper has presented the types of models, an overview of IDEF, the creation of static models, static model creation process, the process for the conversion to dynamic models, and observations from the research. The research did substantiate the premise that a static model can be used to drive a simulation. Beginning with a static representation can reduce the time necessary to create a validated simulation. Adding the necessary information to the static model was accomplished in a simple, straightforward manner. The conversion process was again simple. The entire process, once the steps are understood is quite simple. It is the opinion of these authors that the ease of use of these tools makes the concept of using a static model as the master for review and analysis the preferred method. Most casual users of simulation (not specific simulation tool experts) will find this method simpler than entering and reviewing the information directly in the simulation tool.

A planned extension of this research is to apply this method to more complex, real-world models to validate the extensibility of the approach. When the models become large enough to require several levels of decomposition, is this method still simpler than the use of the simulation tool itself? Also, as more complexity is modeled, do the static tools incorporate all of the required parameters?

An extension of this research would be to include the various views of an enterprise and the links between views and to investigate how these links could be used to enhance the transition to a simulation model.

This research demonstrated that there are multiple methods available to reach a correct and complete model that is useful for both documentation and analysis and can enable the use of a master model.

ACKNOWLEDGMENT

Research for this paper is funded in part by the National Science Foundation sponsored Agile Aerospace Manufacturing Research Center.

REFERENCES

Automation & Robotics Research Institute. 1991. *A consensus process model for small manufacturers, an IDEF3 model*. Automation & Robotics Research Institute: Fort Worth, TX.

Harrell, C.C., and K.C. Field 1996. Integrating process mapping and simulation. In *Proceedings of the 1996 Winter Simulation Conference*, ed. J.M. Charnes, D.J. Morrice, D.T. Brunner, and J.J. Swain, 1292-1296. Association for Computing Machinery: Coronado, CA.

Knowledge Based Systems. 1996. *ProSim user's manual and reference guide*. Knowledge Based Systems, Incorporated: College Station, TX.

Lingineni, M., B. Caraway, P. Benjamin, and R. Mayer. 1996. A tutorial on ProSim: A knowledge-based simulation model design tool. In *Proceedings of the 1996 Winter Simulation Conference*, ed. J.M. Charnes, D.J. Morrice, D.T. Brunner, and J.J. Swain, 476-480. Association for Computing Machinery: Coronado, CA.

Marca, D.A., and C.L. McGowan. 1988. *SADT: structured analysis and design technique*. McGraw-Hill Book Co., Inc.: New York, NY.

Mayer, R.J. 1992. IDEF0 function modeling - A reconstruction of the original Air Force Wright Aeronautical Laboratory Technical Report - AFWAL-TR-81-4023 (the IDEF0 Yellow Book). 1st ed., Knowledge Based Systems, Inc.: College Station, Texas.

Mayer, R.J., M. Painter, and P. deWitte. *IDEF family of methods for concurrent engineering and business re-engineering applications*, 1992. Knowledge Based Systems, Inc.: College Station, TX

MetaSoftware. 1997. *WorkFlow modeler user's manual for MS Windows*. MetaSoftware Corporation: Cambridge, MA.

Nathan, B., and J. Wood. 1991. *The use of IDEF0 to document a methodology - A novice's point of view*. Automation & Robotics Research Institute: Fort Worth, Texas.

Pratt, D.B., J.H. Mize, and M. Kamath. 1993. A case for bottom-up modeling. In *Proceedings of the 2nd Industrial Engineering Research Conference*, 430-434. Los Angeles, CA.

Presley, A., B. Huff, and D. Liles. 1993. A comprehensive enterprise model for small manufacturers. In *Proceedings of the 2nd Industrial Engineering Research Conference*, 664-668. Los Angeles, CA.

ProModel Corporation. 1996. *ServiceModel user's guide*. ProModel Corporation: Orem, UT.

Reimann, M.D., and J. Sarkis. 1996. An integrated functional representation of concurrent engineering. *Production Planning and Control*, 7(5):452-461.

Rumbaugh, J., M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen. 1991. *Object-oriented modeling and design*. Englewood Cliffs, NJ: Prentice-Hall.

Snodgrass, B.N. 1993. *Integrating activity based costing with IDEF modeling*. D. Appleton Company: Dallas, TX.

AUTHOR BIOGRAPHIES

LARRY WHITMAN is a Research Engineer with the Automation & Robotics Research Institute (ARRI) of The University of Texas at Arlington (UTA). He is currently pursuing his Ph.D. degree from the Industrial and Manufacturing Systems Engineering department at UTA. He received his MSIE and BSET degrees from Oklahoma State University. Prior to joining ARRI, he spent ten years in the aerospace industry integrating factory automation and developing and supporting CAD systems. His research interests are in enterprise modeling, simulation, strategic cost justification, and enterprise applications in manufacturing.

BRIAN HUFF is an Assistant Professor in the Industrial and Manufacturing Systems Engineering Department at the University of Texas at Arlington (UTA). He received his Ph.D. and M.S. degrees from UTA. He received his B.S. in Petroleum Engineering from West Virginia University. His research interests are in manufacturing systems design, industrial simulation, industrial automation and robotics, and shop floor production execution and control.

ADRIEN PRESLEY is an Assistant Professor in the Division of Business and Accountancy at Truman State University. Previously he was a Research Industrial Engineer with the Automation & Robotics Research Institute (ARRI) of The University of Texas at Arlington (UTA). He received his Ph.D. degree from the Industrial and Manufacturing Systems Engineering department at UTA. His research interests include enterprise modeling and analysis and management of technology.