To Be Published in the Proceedings of the Summer Computer Simulation Conf., San Jose, CA, 2004, Methodology Track.

Design of Scalable Simulation Models for Semiconductor Manufacturing Processes

Ranjit K. Singh
Hessam S. Sarjoughian
Gary W. Godding
Arizona Center for Integrative Modeling & Simulation
Department of Computer Science & Engineering
Arizona State University
Tempe, Arizona, 85281-8809
{Ranjit.Singh | Sarjoughian | Gary.Godding}@asu.edu

Keywords: Configuration, Scalability, Semiconductor, Modeling, DEVSJAVA, Software Design.

Abstract

Modeling of manufacturing processes plays a key role in analysis and design of semiconductor supply chain networks. As the scale and subsequently complexity of a supply chain network grows, the characterization of its process model which consists of physical elements (e.g., assembly test and semi-finished goods) becomes highly important. In particular, it is essential to employ a modeling approach that can handle specification of scalable physical models as size and intricacies of data and control increases. In this article, we present a set of primitive and composite discrete event model components targeted for manufacturing and logistic aspects of semiconductor supply chain networks. We describe the design of these models in terms of their capability to support scalable structural model configuration using well-defined uniform network data and control communication schemes. In addition, we highlight the software design of these models within the DEVSJAVA environment. The paper concludes by examining the importance of this approach and its application to support modeling physical and decision-policy layers of semiconductor supply networks.

INTRODUCTION

Semiconductor Supply Networks (SSNs) impact economies worldwide [1, 2]. Fundamental knowledge of how such global enterprises behave is central to their operation and their application to new technical practices. However, it is impractical to undertake what-if experiments with many increasingly complex systems such as semiconductor supply networks because of the significant risks involved [3]. Appropriate simulation modeling, therefore, becomes a necessity. Without it, engineers and decision-makers are unable to predict the impact of new manufacturing plant designs.

A semiconductor supply network is multifaceted – it is an amalgamation of different entities spread across physical and decision layers [4]. Supply networks are a composition of many different and complex aspects and are therefore viewed and modeled at different levels of abstractions. The *physical layer* represents components of manufacturing and logistics facilities stretching across multiple geographical boundaries. At this layer, the behaviors of the different facilities are modeled in terms of how each element processes inputs and produces outputs. The *decision layer* complements the physical layer. It characterizes relatively high-level command and control decisions that span one or more manufacturing or logistic facility. Given the importance of separation between physical and decision layers, it is imperative to understand not only the behavior of each layer individually, but also their interactions. A model encompassing these layers must have well-defined components and corresponding compositions such that it provides a rigorous basis for characterizing control and command both at the physical and decision layers. For example, the decision layer provides high level control and decision-making based on observations of the material processing that occurs at the physical layer.

With modeling and simulation, it is possible to undertake such synthetic experimentations on a SSN. However, the existing state-of-the-art modeling and simulation methodologies, tools, and practices cannot support the scalable characterization of semiconductor supply chain networks. Without modeling constructs, it is difficult to handle inherent complexities that result from many interacting physical elements.

In the remainder of this paper, we will describe the physical elements of the semiconductor supply networks which are used as a basis to develop domain-specific model components and their interactions. We briefly discuss the semiconductor supply network physical layer model and its previous model development in DEVSJAVA. The semiconductor supply chain network structural constructs (e.g., primitive and composite model components) are developed based on the general DEVS modeling constructs. We also highlight the software design of the semiconductor supply chain model library. The paper concludes with an assessment of the scalability trait of the semiconductor supply network model abstractions and their corresponding modeling constructs.

SCALABILITY

An important attribute of manufacturing systems is scale which directly relates to structural and behavioral aspects of systems. Scalability can be considered from different points of views – e.g., modeling and simulation and software engineering. From an M&S point of view, an adopted modeling and simulation approach needs to support scalability. Modeling constructs and simulation protocols need to naturally support increasing the scale of a model – e.g., increase number of model components and their interactions. For example, without undue difficulty and complications, the size of model with a few tens of components should be increased to many thousands. This implies *(i)* the foundation of a modeling framework needs to support component-based modeling, *(ii)* the simulation of these models are amenable to parallel and/or distributed execution, and *(iii)* the modeling and simulation capabilities lend themselves naturally to component-based computation.

An approach that can satisfy these three traits is the Discrete-Event System Specification (DEVS) [5-7]. A realization of this framework is DEVSJAVA. However, since DEVS is a generic, it must remain neutral to particular domains such as semiconductor supply network systems. The consequence is that the generic scalability of DEVS needs to be extended to take advantage of scalability traits that are inherent to network-based systems. Therefore, of the above three scalability traits, it is evident that component-based modeling and software design are the primary enablers for large-scale model development. In the remainder of this paper, we describe the concept of domain-specific scalability for semiconductor supply networks, look at modeling constructs to support scalability, and highlight design extensions to the DEVSJAVA software.

SEMICONDUCTOR SUPPLY NETWORK MODELING

A semiconductor supply network requires models of material processing through a network of components such as assembly and finish lines. The network also includes warehouses for semi-finished goods, finished goods and shipping (see Figure 1). The physical layer depicted in Figure 1 can be developed in a variety of modeling and simulation environments (e.g., ARENA [8] and DEVSJAVA [9]) and standards (e.g., Workflow Management Coalition [10] and Supply Chain Operations Reference-model (SCOR) [11]). For example, a semiconductor supply network is modeled using DEVS hierarchical composition of components [4-7]**.** The network also includes warehouses for semi-finished and finished goods as well as shipping (see Figure 1). There are fabrication plants (*FAB*), end of factory stores (*EOF*), processed wafer inventories (*PWI*), assembly/test lines (*AssmTest*), semi-finished inventory (*SFGI*), finish lines (*Finish*), finished goods (*FG*), geographical warehouses (*Geog Whrs*), shipping (*SHPG*), and customers (*Cust*).

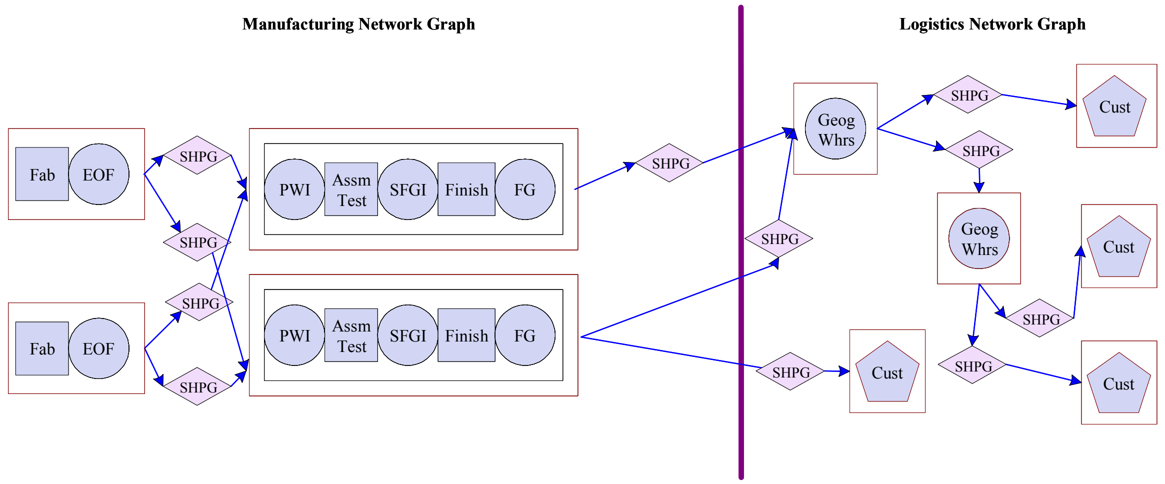

Figure 1: Semiconductor Supply Network Domain-Specific Scalability

Taken together these elements constitute the physical layer which, as alluded to in the previous section, is a conscience separation from the decision layer for characterizing fundamentally different dynamics within an SSN. At the end of each simulation cycle, the manufacturing model components send their current state information (e.g., inventory level and orders filled) to the decision layer. Similarly, the decision layer sends control signals (e.g., how much to be released from an inventory) to the elements of the physical layer at the end of each iteration cycle. State information from the physical layer to the decision layer and vice versa provides a feedback-control loop that supports control of manufacturing processes based on higher-level decisions. Therefore, the separation lends itself to model *scalability* and *reuse* – e.g., models developed for the manufacturing line can be scaled and reused with the decision layer model. We note that the design specification for communications between these layers plays a central role toward supporting quality attributes such as performance and scalability.

Domain Specific Scalability

Scalability for the semiconductor supply network model can be viewed in terms of generic as well as domain-specific modeling constructs. A generic component-based modeling framework can support scalability of components, their composition, and encapsulation. Domain-specific capability, however, can offer higher levels of modeling by identifying common behavior of components and patterns of interactions among them. Figure 2 illustrates the conceptual separation of the supply network domain-specific modeling from domain-neutral modeling.

With this separation, domain specific scalability can be realized. The key advantage of domain-specific scalability is the support for model reusability at the domain level as opposed to relying on low-level modeling. Designing domain-specific models are particularly suited to handling scalability and complexity traits for configuring a semiconductor supply network. Combining generic (*domain-neutral*) and *domain-specific* scalability offers a disciplined approach for developing large-scale simulation models instead of depending on ad-hoc approaches. Furthermore, it enables managing complexity and scalability aspects from a model composability point of view [12, 13]. The detailed software design of the interactions shown in Figure 2 is outside the scope of this paper (for related software design specification techniques see [6]).

APPROACH TO MODELING SUPPLY NETWORK SYSTEMS

A semiconductor supply chain network can be seen as a network of nodes and connectors. To address data and control flow through a manufacturing line in a generic setting, it is key to specify suppliers and consumers. For example, *what to send* and *where to send it* are not static traits determined by topology alone. Rather, material flow *paths* are determined by the decision layer and are routed accordingly at the material processing layer dynamically during a simulation run. This implies that the decision layer (and those components with multi-path releases) is *topology aware*. That is, behavioral logic within some of the models is dependent upon the current topological layout of the material processing layer model. Thus, any change to the topology (such as magnitude scaling) requires subsequent logic changes to occur. The solution is to find a way to facilitate such scaling through configuration alone while avoiding changes to the logic inside of the models. Our resulting approach is to configure topologies of the models where low-level interactions among model components are handled automatically.

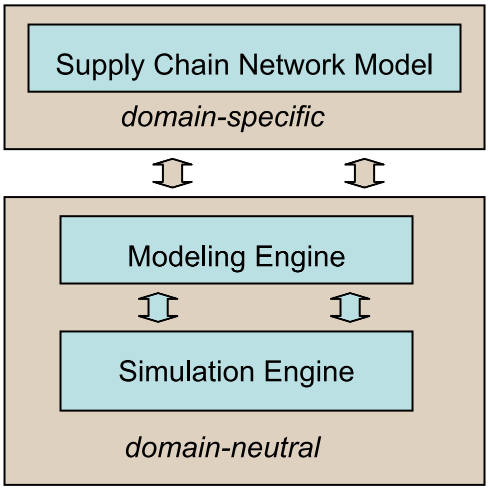

Figure 2: Model Scalability at the Domain Level

Design of Semiconductor Supply Network Model Components

To help identify patterns within the Semiconductor Supply Network Model, we focus on its *characterization as a network* and related concepts (routing and composition) from graph theory and software engineering. The core building blocks of a SSN model can be categorically separated into a *Manufacturing Network Graph* and a *Logistics Network Graph* (see Figure 1) where a graph represents a set of nodes and edges. The characterization of the manufacturing network and logistic network in terms of their structural data and control roles and responsibilities is as shown in Figure 3.

Each model component depicts product flows and control flow in and out of it. The separation offers modularity and the ability to reuse Logistical topologies with different Manufacturing layouts. More importantly, as noted above, we can develop new types of scalability and reusability using common traits of graphs as applied to semiconductor network systems.

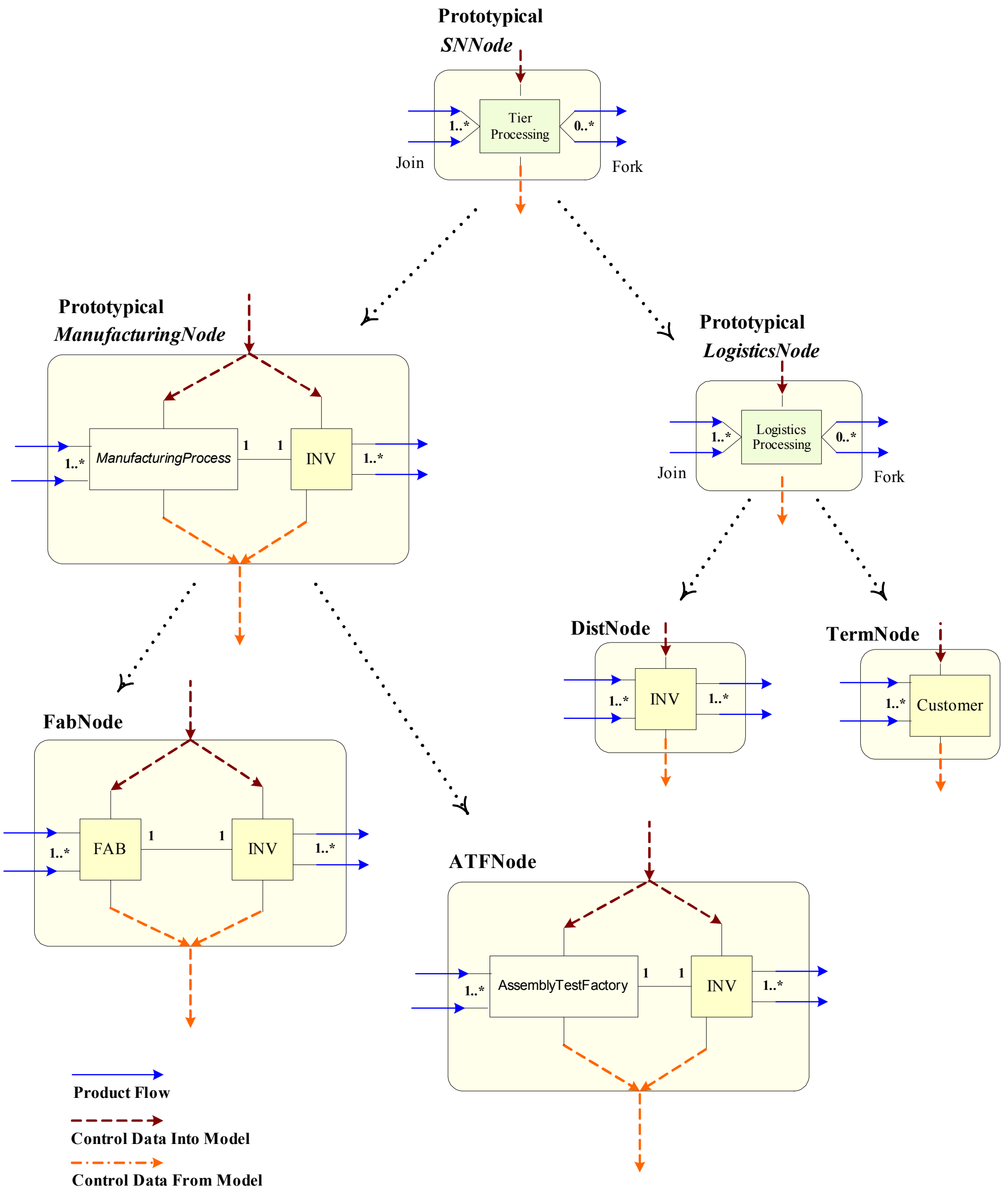

Figure 3: Primitive Physical Model Components for Semiconductor Supply Chain Networks

The SNNode is the most generic model and represents the basic pattern shared among the elements of a supply network as shown in Figure 1. It signifies that an entity can accept and join one or more connections (dynamically configurable) and can send data to one or more outgoing ports (also dynamically configurable). Other model elements are the ManufacturingNode and LogisticsNode where distinct aspects of manufacturing and logistics are accounted for. The other model elements are the FabNode and ATFNode which are specialized to handle FAB and Assembly Test Factory (ATF), respectively.

The elements of the logistic network are DistNode and TermMode where the former represents a *distribution* capability and the latter represents a *termination* point where material flow ends (e.g., a customer).

To configure models as shown in Figure 4, a set of adjacency matrices can be specified for automatic connectivity among components of the manufacturing network graph, logistics network graph, and their combination. The key value of these matrixes is inherent scalability since they automate control interaction based on the concept of connecting every component to all other components.

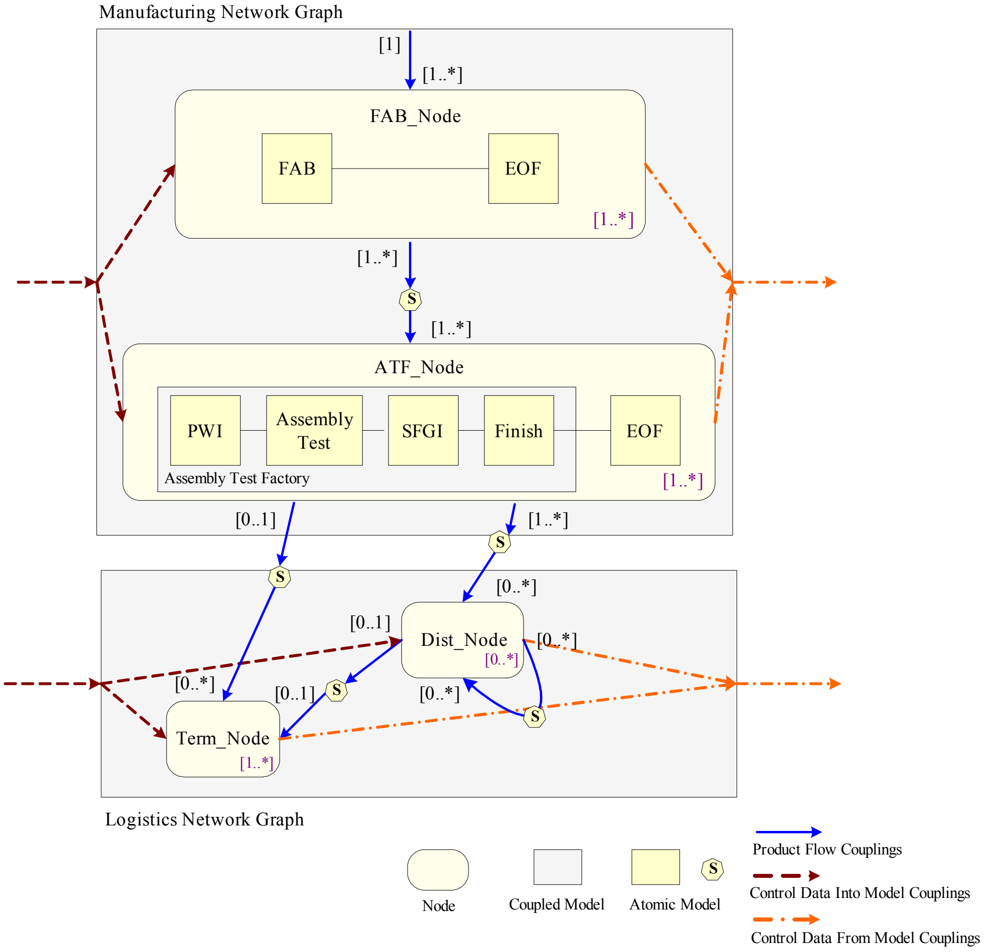

Figure 4: Reconfiguring model components using common data and control interconnections

Note that the connectivity among all components is only for control data into and from model as shown in Figures 3 and 4. The drawback of this approach is the necessity for the receiving components to determine whether messages it receives are intended for it or not – the number of message interactions can increase exponentially. This increase coupled with message-inspection paradigms degrades performance. However, performance can be restored by automatically mapping a configuration to an equivalent model where unnecessary interactions are removed.

SOFTWARE DESIGN OF SEMICONDUCTOR SUPPLY NETWORK MODEL COMPONENTS

This section discusses the software design of the semiconductor supply network modeling elements. These models constitute a library which extends the DEVSJAVA environment. The software specification extends the ViewableAtomic and ViewableDigraph base classes of the DEVSJAVA environment using two configurable interfaces – ManufacturingProcess and Configurable.

The interfaces provide the basis to introduce domain-specific classes that can support scalable control and data ports. The Configurable provides a consistent input/output port scheme to support configurability among all models that implement this interface. The ManufacturingProcess identifies each model that implements this interface as an aspect of the SSN manufacturing process. The two interfaces allow automatic interchanging of domain-specific atomic or coupled models such as SNNode, ManufacturingNode, LogisticsNode, FabNode, ATFNode, DistNode and TermNode which have a variety of parent/child and realization relationships. These relationships are critical for uniformly supporting scalable configurability instead of relying on customized, low-level modeling. This uniformity is achieved via data (dataIn and dataOut) and control (ctrlIn and ctrlOut) ports for control and data messages, respectively. At a higher level of abstraction, our ManufacturingNetworkGraph and LogisticsNetworkGraph classes logically divide the SSN into its two recognizable manufacturing and logistics parts.

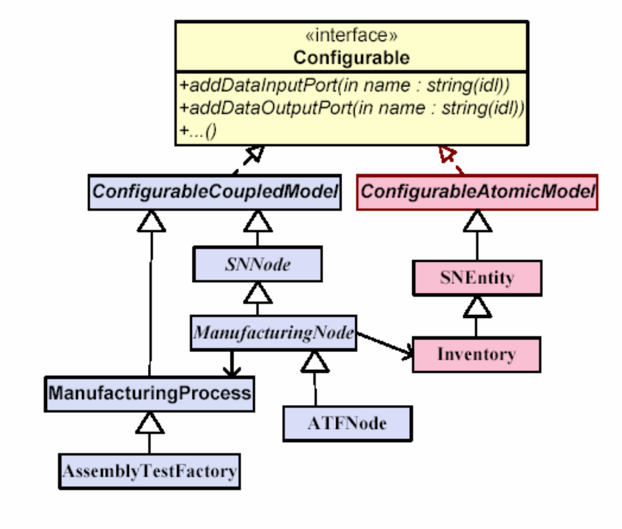

Figure 5: Partial Specification of the DEVSJAVA Classes for the SNM Domain Model Components

CONCLUSIONS

Handling scalability from the domain-specific perspective offers an additional means to overcome known modeling challenges when faced with simulating large-scale semiconductor supply chain network systems. In particular, with this approach, the specification of data and control interactions (couplings) between manufacturing and decision models is simple and scaleable – e.g., as the number of model components in the manufacturing process level increases, there is no need to create couplings manually. Furthermore, specification of composite models such as ATFNode is scalable. The result of this work is a well-defined separation between domain-specific and general component-based semiconductor model development. This approach is the culmination of the principles of networks as applied to manufacturing and logistic processes. The modeling environment offers basic and composite model components for the semiconductor supply network. These models were developed as extensions to the generic DEVSJAVA modeling engine. The correctness of the hierarchical general data and control specification was verified using the supply chain model developed in [4].

Acknowledgement

This research has been supported by a research grant from the Intel Corporation Research Council and partially by NSF grant DMI-0122227. The authors are grateful to Karl G. Kempf of the Intel Corporation for his contributions and stimulating discussions on the approach and development of this work. The authors also extend their thanks to Toni Farley of the Computer Science and Engineering Dept. at ASU for her contributions during the early phase of this research project.

REFERENCES

[1] Jain, S., et al, 1999, *Criticality of Detailed Modeling in Semiconductor Supply Chain Simulation*, *Proceedings of 1999 Winter Simulation Conference*, p. 888-896, Phoenix, AZ.

[2] NRC, 2002, *Modeling and Simulation in Manufacturing and Defense Acquisition: Pathways to Success*, Washington DC: National Research Council, National Academy Press.

[3] Kempf, K.G., 2003, *Optimization of a Semiconductor Supply Chain*, *Proceedings INFORMS Conference on OR/MS Practice*, Phoenix, AZ.

[4] Godding, G.W., H.S. Sarjoughian, and K.G. Kempf, 2003, *Semiconductor Supply Network Simulation*, p. 1593-1601, *Proceedings of Winter Simulation Conference*, New Orleans, LA, USA.

[5] Zeigler, B.P., H. Praehofer, and T.G. Kim, 2000, *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*, Second Edition, Academic Press.

[6] Singh, R., 2004, *A Software Architecture Design for Discrete Event Simulation Environments*, *Computer Science & Engineering Dept.*, Master Thesis, Arizona State University, Tempe, Arizona.

[7] Sarjoughian, H.S. and R. Singh, 2004, *Building Simulation Modeling Environments Using Systems Theory and Software Architecture Principles*. *Advanced Simulation Technology Symposium*, p. 99-105, April, Wash. DC, SCS.

[8] System-Modeling, *ARENA*, 2004, http://www.arenasimulation.com.

[9] ACIMS, 2004, *DEVSJAVA software*, http://www.acims.arizona.edu/SOFTWARE/software.shtml.

[10] WFMC, 2004, *Workflow Management Coalition*, http://www.wfmc.org.

[11] Supply Chain Council, 2003, http://www.supply-chain.org.

[12] Sarjoughian, H.S. and F.E. Cellier, 2001, *Toward a Unified Framework for Simulation-Based Acquisition*, in *Discrete Event Modeling and Simulation Technologies: A Tapestry of Systems and AI-Based Theories and Methodologies*, H.S. Sarjoughian and F.E. Cellier, Editors, p. 1-14, Springer-Verlag.

[13] Godding, G.W., H.S. Sarjoughian, and K.G. Kempf, in-preparation, *A Multi-Formalism Modeling Approach for Semiconductor Supply Network: Composing Discrete-event and Linear Program Models*.