

## **AN OVERVIEW OF CIM ENTERPRISE MODELING METHODOLOGIES**

Ganesh Kateel  
Manjunath Kamath  
David Pratt

Center for Computer Integrated Manufacturing  
School of Industrial Engineering and Management  
Oklahoma State University  
Stillwater, Oklahoma 74078, U.S.A.

### **ABSTRACT**

Computer integrated manufacturing (CIM) systems are increasingly being used as weapons by manufacturing enterprises in competitive business environments. The complicated nature of these systems and the high initial investment requirements have necessitated their accurate modeling. A number of models, modeling methodologies, and modeling tools have been developed and used for this purpose. We first present a brief overview of several CIM models as well as modeling tools and methods. Many of the models are said to emphasize only a part of the system. A concern in the research community is that these models must be integrated. We conclude the paper by examining the rationale and feasibility of integrating the different models and/or creating integrated models.

### **1 INTRODUCTION**

Many manufacturing firms, zealous to maintain their competitiveness in the world class manufacturing environment, attempt to use state of the art technology in their operations. Advancements in computer and information technology have led to the development of computerized applications in many functional areas of manufacturing enterprises. As a result, we have a plethora of three letter acronyms starting with C (for Computerized) such as CAD, CAM, and CAE along with others such as MRP, MRP-II, and SFC which deal with different aspects of manufacturing. The design of such systems is generally developed in a fashion that emphasizes 'local' solutions mostly overlooking the overall organizational effectiveness. This together with the use of heterogeneous databases and incompatible computer operating systems, has led to 'islands of automation' which suffer from data inconsistencies and lack of control of the functional interactions between manufacturing application systems.

Traditionally, different functions of enterprises have been modeled with a focus on the performance evaluation of the individual departments (referred to in this paper as local optimization) rather than the effectiveness of the entire organization. But, a modern enterprise requires the departmental goals to be in tune with the organizational goals for its betterment and survival. Designers need tools that will allow them to evaluate the impact of local decisions on the performance of the total enterprise. However, many CIM models and modeling approaches tend to focus only on particular functional parts of the enterprise. Hence, one of the issues that we examine in this paper is the rationale and feasibility of integrating the different models and/or creating integrated models.

The rest of the paper is organized as follows. Several different definitions of CIM are provided in Section 2 as a basis for further discussion. Section 3 provides a general perspective on modeling with an emphasis on CIM modeling. CIM enterprise architectures, modeling tools and methods are briefly discussed in sections 4, 5, and 6 respectively. Section 7 discusses the issues pertaining to integration of models.

### **2 CIM ENTERPRISES-DIVERSE PERSPECTIVES**

Jorysz and Vernadat (1990) state the objective of CIM as the appropriate integration of enterprise operations by means of efficient information exchanges within the enterprise with the help of information technology. This indicates that CIM includes all of the design and manufacturing functions of CAD/CAM, as well as all the business functions such as marketing and accounting. A comprehensive definition of CIM has been given by Doumeingts et al. (1995): "CIM refers to a global approach in an industrial environment which aims at improving industrial performances. This approach is applied in an integrated way to all activities, from designing to delivery and after sale, and uses

various methods, means and techniques (computer and automatic) in order to simultaneously improve productivity, decrease costs, meet due dates, increase product quality, secure flexibility at local or global level in a manufacturing system, and involve every actor. In such an approach, economic, social and human aspects are at least as important as technical aspects.”

Nagata et al. (1993) describe CIM as “an integrated system which combines the areas of production, marketing and R&D, to manage and operate them under a single management strategy with the support of computers so that the production operation can be efficient and flexible.”

A few of the other definitions of CIM are: “The automation and integration of information, processes, and functions in a manufacturing environment, including customers and vendors, with the result being a closed-loop, functionally integrated manufacturing planning and control system” (Weston 1994). “CIM is the integration of computer based manufacturing process, drawing on a common database and communicating via some form of computer networks” (Bessant 1991).

The foregoing discussion indicates that most authors encompass a broad range of enterprise activities in the definition of CIM. So it is difficult and may be inappropriate to view CIM in isolation. It is not surprising to see that many enterprise modeling tools, methods and architectures are being used to model or are being structured to model CIM. Further evidence for this can be found in Ngwenyama and Grant (1994).

Generally the implementation of CIM involves high investment and complex system design issues. Although much research is being carried out regarding the development and implementation of CIM, many problems are yet to be addressed satisfactorily. Ngwenyama and Grant (1994) provide a list of these problems which includes lack of integration, islands of automation, sub-optimization of resources, ad-hoc development, lack of well-defined architecture, confusing definition of CIM and so on. *Many researchers have tried to address these problems by developing good models of CIM because models form a rational basis not only for designing new systems but also for learning about existing systems. Evaluation using models instead of the actual systems allows fast acquisition of knowledge and avoids the risk of costly disruptions to the real system. Models can be used for system optimization, performance analysis and prediction, gathering insight into the system or facilitating learning processes.*

### 3 CIM MODELING

Modeling is the process of creating an abstraction of a real world system which reflects the system properties to the desired degree of detail (Kochikar and Narendran 1994). Models filter out irrelevant details and represent only information essential to the task. The traditional approach to modeling begins with the identification of the purpose of the model. Then the modeler identifies all system features germane to the purpose, constructs and validates an abstract structure with sufficient scope and complexity so that it can aid in integration of system components and their interaction under suitable assumptions.

The purpose driven, tool dependent modeling described above has the inherent disadvantage that one has to construct different models for different purposes even though the system being represented is the same. Hence, Duse et al. (1992) suggest building robust, tool independent models to represent manufacturing enterprises from which one can extract the information relevant for a particular purpose. They refer to this model as a *Base Model*. This base model is a rich, robust, representation of the system consisting of physical, informational and control elements as well as their relationships.

The above base model concept is a new paradigm in the field of manufacturing systems modeling, and additional research is necessary before researchers fully implement this concept. Meanwhile, constraints such as cost, quality, flexibility, time, size, nature, and complexity make the designing and modeling of CIM a very complicated task and as such requires good modeling methodology and tools. The comprehensive definition of CIM provided by Doumeingts et al. (1995) indicates that CIM systems are remarkably complicated with a multitude of activities carried out by human beings and machines using information flows and control signals. Generalized models and well defined architectures are required to understand the system complexity at a manageable level. Since humans have “bounded rationality” they like to break (such) complicated systems into better manageable pieces for analysis, design and management (Little 1991). Humans prefer to abstract only that part of reality which may be relevant to them. Due to this tendency, most of the modeling approaches have separate models for different aspects of CIM. The above argument follows from systems theory. It is well known that any system, with a very few exceptions, consists of subsystems. These subsystems can be considered to be systems in their own right. Hence, they can be modeled independently. This may not allow us to have a ‘big-

picture' of the system and it is difficult to move from one model to the other (Savolainen et al. 1995).

The means that we use to model systems are referred to as *modeling tools*. An Entity Relationship (ER) diagram is an example of a tool used for representing the relationship between entities. The procedure to construct models may be termed as *methodology*. The result of using the modeling tools in a manner prescribed by the methodology is a representation or a *model*. In the case of CIM modeling this may be an architecture.

As a preparation for the further discussion about CIM enterprise modeling issues, a brief overview of existing architectures, modeling tools, and modeling methods is presented in the following sections.

#### 4 CIM ENTERPRISE ARCHITECTURES

Architectures of CIM systems contain conceptual models as well as rules that help to translate the models into a working reality (O'Sullivan 1994). O'Sullivan defines architecture as "a body of rules that define those system features which directly affect the manufacturing environment into which the system is placed. These features include system configuration, component locations, interfaces between the system and its environment, and modes of operation." Most authors refer to two types of architecture. The first is a detailed collection of generic information management and automatic control tasks and their necessary functional requirements referred to as *reference architecture*. A *particular architecture* is the instantiation of a reference architecture.

A CIM reference model committee of Purdue University (1989) describes a reference model or architecture as a previously agreed upon or standard definitive document or conceptual representation of a system. The reference model defines requirements common to all implementations but is independent of the specific requirements of any particular implementation. Several reference architectures have been put forth by collaborative research projects, computer manufacturers and individuals. Some of the important architectures frequently referred in the literature are briefly discussed below.

**CIM-OSA:** CIM-OSA is an Open-System Architecture and has three levels of model derivation viz., requirements definition, design specification and implementation description, four views viz., functional, informational, resource, and organizational views, and three levels of instantiation viz., generic, partial and particular models (Jorysz and Vernadat 1990). This

architecture guides the system designer in deciding what is to be implemented to achieve what is required.

**ICAM Architecture:** ICAM (Integrated Computer Aided Manufacturing) architecture uses tools such as IDEF0 (ICAM DEFinition - Zero) and IDEF1 (ICAM DEFinition - One). A Hierarchical decomposition method is used to define this architecture. The architectures generated in ICAM are not generally available publicly (O'Sullivan 1994).

**CAM-I Architecture:** The Computer Aided Manufacturing - International (CAM-I) architecture is a general representation of manufacturing enterprises. The functional decomposition method used to create this architecture is prescribed to indicate the details such as company policies and procedures, organizational structure and standards (Doumeings et al. 1995).

**NBS Architecture:** The NBS (National Bureau of Standards - now, National Institute of Standards and Technology) architecture uses a hierarchical control approach with five levels of hierarchy viz., factory, shop, cell, workstation and machine. Each level can be broken into more activities. The decomposition is based on procedures, functions, or rules. NBS (NITS) developed this architecture so that manufacturing system vendors could develop products compatible for CIM (O'Sullivan 1994).

**IMPACS Architecture:** The IMPACS architecture uses IDEF0, Data Flow Diagrams (DFD), Graphe a Resultatis er Activities Intercies (GRAI) Grids and nets, IDEF1x and group technology. IMPACS outlines a cellular architecture. The production cells are controlled by software modules such as dispatcher, scheduler, mover, producer and monitor. These software modules are designed to be compatible with each other even when they are developed by different vendors. O'Sullivan (1994) states that the IMPACS architecture has been widely accepted among manufacturing software vendors as a useful and practical interpretation of the production management system.

#### 5 MODELING TOOLS

Models are built using the modeling tools in the manner prescribed by modeling methodologies. *System modeling tools* refer to techniques used for diagrammatically representing functions or activities (O'Sullivan 1994).

**IDEF Modeling Tools:** IDEF0, IDEF1, IDEF1x, and IDEF2 are modeling tools developed by the ICAM (Integrated Computer Aided Manufacturing) project of the US Air Force at Soft Tech, Inc. These four main modeling tools, which are complementary to each other, provide functional, informational and dynamic models of the system. IDEF models are used primarily for requirements definition.

**IDEF0 Models:** IDEF0 is a comprehensive and expressive functional modeling language capable of graphically representing a wide variety of business, manufacturing and other types of enterprise operations to any level of detail. Three important features of IDEF0 technique are activity modeling graphics, gradual exposition of details and disciplined team work. The basic construction used in an IDEF0 model is a function block linked to other function blocks by inputs, outputs, mechanisms and controls. Links between the blocks may be either physical objects, such as material flow, or information flow. IDEF0 models have three important features viz., 'context' which indicates the position that the subject model takes up in the systems hierarchy, 'viewpoint' which refers to the perspective which the model adopts, and the 'purpose' which indicates the reason for existence for the model (Pandya 1995). IDEF0 has been widely used in industry because of its user-friendliness, computer support, rigor and conciseness, and well documented rules and procedures. The static nature of the models produced using IDEF0 has been cited as the major drawback of this tool. A comprehensive treatise on the state of the art of IDEF0 can be found in Colquhoun et al. (1993).

**IDEF1 and IDEF1x Models:** IDEF1 is a technique for modeling information requirements of a function, in terms of the structure of information. (An overly simplified example: What information should an invoice have?). It is based on the entity-relationship approach developed by Chen (1976). IDEF1x is an extension of IDEF1 and deals with the flow of information. IDEF1x is not widely used due to its drawbacks such as its inability to support composite entity types and the strict procedural requirements such as complete enumeration of attributes of entities before instantiation (Pandya 1995).

**IDEF2 Models:** IDEF2 models have the capability to describe as well as analyze a system (Banerjee and Al-Maliki 1988). However, IDEF2 is an

unsupported simulation language (Young and Vesterager 1991). Hence, other simulation languages such as ARENA (Drevna and Kasales 1994) and PROMODEL (Baird and Leavy 1994) are more commonly used.

**Structured System Analysis (SSA):** (Gane and Sarson 1979). SSA is considered as a data flow approach to systems design. It is also quite effective in representing the flow of physical entities. This technique prompts the user to think in terms of what to accomplish than how. SSA uses hierarchical decomposition similar to IDEF0. However, it is more detailed and software oriented than IDEF0 and is the most preferred modeling tool for data flows (O'Sullivan 1994). This is also referred to in the literature as Data Flow Diagrams (DFD).

**GRAI Grids and GRAI Nets:** GRAI grids and GRAI nets are the tools developed by the GRAI Laboratories of France to model decision-flow processes in manufacturing environments. In the GRAI grid, various activities are modeled with respect to the decisions and information flows between them while the GRAI net delineates the decision making process itself.

## 6 MODELING METHODS

*Modeling methods* are guidelines to combine the modeling tools described above to model a particular system. Some of the methods used for modeling CIM enterprise systems include IDEF, SSADM, SADT, GIM, and CIM-OSA cube.

**IDEF Method:** This methodology prescribes the integration of IDEF0, IDEF1 and IDEF2 models which describe functional, informational and dynamic model of an enterprise, respectively.

**Structured Systems Analysis and Design Methodology (SSADM):** SSADM is a procedural framework which was developed specifically for use in system development projects. It uses three modeling tools: data flow diagrams (DFDs), logical data structures (LDSs), and entity life histories (ELHs) to provide function, data, and event views of the systems, respectively (Pandya 1995).

**Structured Analysis and Design Technique (SADT):** SADT (Ross 1977) is a system analysis and design methodology to represent the structure of the system diagrammatically. The representation begins with a diagram depicting the general description of a system. Modelling then progresses in a top down fashion. This

method makes use of a number of graphical and textual tools such as activity diagrams, data diagrams, node lists, and data dictionaries to represent the structure of the system being addressed. The analysts are required to consider activity or function and data views of the system being modeled, thereby encouraging the creation of integrated enterprises. The methodology requires a functional model before considering the physical design. This methodology has computer support in the form of software packages viz., AUTOIDEF0 and SPECIFX.

**GRAI Integrated Method (GIM):** The ESPRIT IMPACS project uses an approach which employs tools such as GRAI, IDEF0, IDEF1x, and Group Technology. This combination is called the GRAI Integrated Method (GIM).

**CIM-OSA Method:** CIM-OSA defines an integrated methodology to design, implement, operate, and maintain an enterprise. The CIM-OSA method deploys many well-accepted ideas and principles to design a CIM enterprise. These include functional decomposition in SADT; function/activity and information modeling used in IDEF, the entity-relationship model and the three schema approach from ANSI/SPARC development for data communications (Jorysz and Vernadat 1991).

**Object-Oriented Approach:** Object oriented concepts provide a new promising ontology for enterprise and CIM modeling. Ngwenyama and Grant (1994) and Kim et al. (1993) suggest ways to model CIM information systems using an object-oriented approach. The methodology proposed by Kim et al. (1993) consists of an analysis phase and a design phase. In the analysis phase, functional decomposition is employed to define the information flow among the manufacturing functions and their infrastructures. Decomposed functions are represented by functional diagrams. The functional diagrams are then transformed into an object-oriented information model consisting of a class dictionary and class relationship diagrams. Using these, a class dictionary consisting of function classes and entity classes is formed. These class dictionaries can then be translated to a specific data dictionary of an object oriented database management system.

## 7 MODELS, ARCHITECTURES, AND INTEGRATION

Consideration of the architectures and modeling methods described above reveals that almost all architectures do rely upon multiple views of manufacturing enterprises. The functional and

informational views appear in most of the architectures. But most of these architectures have differing additional views. IDEF has a dynamic view. The CIM-OSA architecture has a resource view and an organization view. CAM-I depicts three more perspectives viz., management perspective, computer systems perspective, and physical structure perspective in addition to functional and informational perspectives.

Savolainen et al. (1995) note that a single model will not be able to capture all aspects of a CIM enterprise. Therefore, a complete model is composed of different views, aspects, or perspectives.

Weston (1994) suggests that CIM may be viewed from three different perspectives, viz., from an engineering perspective, from the viewpoint of information system and networking and from the operations view point. He encourages manufacturing organizations to examine CIM in the context of a total organizational perspective. He argues that a single dimensional CIM perspective most likely will produce results that are less than desired, particularly in a global competitive environment.

Brandimarte and Cantamessa (1995), however, note that while the aim of the CIM community is to develop an integrated manufacturing enterprise, the integration between the different domains and cultures which cooperate in this field is difficult to achieve. But it would be advantageous to the CIM community to integrate the different disciplines of CIM modeling, mainly the physical, communication (information) and operational dimensions.

Little (1991) observes that "we indulge in modeling myopia if we believe as system analysts that we can (or should) be building complete models of our system and setting all the control variables. Doing so misses the major opportunities for system improvement that are possible by finding new ways to improve the people on the front lines of the organization by giving them information, training and tools with which to improve our performance." He further comments "the building of more and more complicated models of systems using the same methodologies is likely to yield diminishing returns. Managers face dozens of different problems each day; not just late schedules and excess inventories but also issues such as key people being hired away, roofs that leak, customer dissatisfaction with products, and employee absenteeism. Thus a hundred different models are often needed, not one big model." This seems to be a very strong argument/opinion against building of a single integrated model.

At this point it may be worthwhile to discuss the differences, if any, between development of an integrated model and model integration. The first is the development of one 'big-picture' (global view) from

which the detailed local models can be built by 'zooming-in'. Conventionally this may be akin to the top-down hierarchical approach. Most of the models discussed above fall in this category. The second is the pooling of the different models and tying them with some common thread. The argument in favor of the first approach (integrated model) is that the local models built with the big-picture in mind will not be striving towards local optimization, rather they would contribute towards the system effectiveness. But the biggest drawback here is that it would be a Herculean task to use this concept in existing systems which are already in operation. It may be easier to adopt this while establishing new systems. A serious challenge for the designers of such models is that those models should be flexible and must be robust to adapt to changing environments without losing focus.

The argument in favor of the second approach (model integration) is that this is a plausible integration methodology in the case of existing organizations. But the problem here is that each of the local models developed separately will be myopic and naturally strive towards local optimization. Another practical problem is to make these models communicate with each other, as these might have been developed on different platforms and may operate in different languages.

An attempt at *integrating distributed models* called the ENVISION architecture was presented by Heim (1994). He describes model integration as an alternative to aggregate refinement and decomposition. He provides a conceptual framework to integrate individually and independently functioning models to communicate with each other by acting as message passing objects in client-server relationships. This allows data sharing and coordination of the activities. This also reduces the programming effort and simplifies model validation. The author presents a prototype implementation of the model integration architecture.

Wang et al. (1993) challenge the approach of integrating distributed models built using different tools to represent the various viewpoints. They cite some of the latest developments in this area reported by Gay, et al. (1991). Gay et al. (1991) are said to have described the functional, informational, and dynamic perspectives of a flexible manufacturing system by applying IDEF0, IDEF1 and SLAM II, respectively. Wang et al. (1993) challenge these approaches using the following arguments: first, since the dynamic view point is being modeled using a language different from that used to represent functional and informational view, some actions have to be re-represented in building the dynamic version of the model. Second, the different modeling tools use different syntax. This may result in loss of information especially if different people have

modeled the various views. Third, any change in the system will require a significant remodeling effort since the change in one is not automatically reflected in the other. This makes the modeling and remodeling process highly inefficient.

Wang et al. (1993) present a comprehensive methodology called IDEM (*Integrated system Description Model*) to overcome this difficulty. IDEM uses an extended IDEF0 methodology to build a functional view of the system. An information view of the model is described by adding an object oriented information model on to the previously defined functional model. A dynamic view of the model is created by associating time attributes to the functional model. The dynamic behavior of the model is governed by defining rule sets. IDEM is developed using a LISP based language called LOOPS.

The first International Conference on Enterprise Modeling Technology (ICIEMT) identified three types of approaches to the problem of syntactic and semantic model integration (Petrie 1992). They are described below.

*Master Models* are a single reference model from which all other models and instantiations are derived. This is similar to what has been referred to in this paper as an integrated model. *Unified models* are metamodels which translate between models. Generally these models may have similar structures. For example, all the models may be databases which can communicate with each other. *Federated models* are loosely coupled models. Here the individual models may not have same structure. In such a case the interface between the models has to be manually constructed. Hence these models are also referred to as interfaced models.

## 8 CONCLUSION

The literature reviewed suggests that there are various tools and methods to model CIM enterprises. There appears to be some consensus that there is a need to develop a generic approach to build CIM models which can represent the entire manufacturing organization while simultaneously allowing each function of the enterprise to maintain its uniqueness. Though the present day models do not appear to have completely achieved this character, some of the approaches appear to be quite promising.

## ACKNOWLEDGMENT

We wish to acknowledge the support provided by the National Science Foundation for this research through Grant No. EEC-9527493.

## REFERENCES

- Baird, S. P., and J. J. Leavy. 1994. Simulation modeling using PROMODEL for Windows. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, S. Manivannan, D. A. Sadowski, A. F. Seila. 527-532. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.
- Banerjee, S. K., and I. Al-Maliki. 1988. A structured approach to FMS modelling. *International Journal of Computer Integrated Manufacturing*. 1(2): 77-88.
- Bessant, J. 1991. *Managing Advanced Manufacturing Technology*. Manchester: NCC Blackwell.
- Brandimarte, P., and M. Cantamessa. 1995. Methodologies for designing CIM systems: A critique. *Computers in Industry* 25: 281-293.
- Chen, P.P. 1976. The entity-relationship model: Toward a unified view of data. *ACM Transactions on Data Base Systems*, 1(1), 1976.
- CIM reference model committee. 1989. A reference model for computer integrated manufacturing from the viewpoint of industrial automation. *International Journal of Computer Integrated Manufacturing*. 2(2): 114-127.
- Colquhoun, G. J., R. W. Baines, and R. Crossley. 1993. A state of the art of IDEF0. *International Journal of Computer Integrated Manufacturing*. 6(4): 252-264.
- Doumeings, G., B. Vallespir, and D. Chen. 1995. Methodologies for designing CIM systems: A survey. *Computers in Industry*. 25:263-280.
- Drevna, M. J. and C. J. Kasales. 1994. Introduction to ARENA. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, S. Manivannan, D. A. Sadowski, A. F. Seila. 77-80. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.
- Duse, M., J. Gharpure, H. Bhuskute, M. Kamath, D. B. Pratt, and J. H. Mize. 1992. Tool independent model representation. In *Proceedings of the 2nd Industrial Engineering Research Conference*, 700-704. IIE, Norcross, GA.
- Gane, C., and T. Sarson. 1979. *Structured Systems Analysis-Tools and Techniques*. Englewood Cliffs, NJ: Prentice Hall International.
- Gay, R. K. L., R. Lim, and L. W. Shing. 1991. Using IDEF methodology for functional, information and dynamic modelling of a FMS. In *Proceedings of International conference on Computer integrated Manufacturing*. ed. B.S. Lim. 97-100. World Scientific, Singapore.
- Heim, J. A. 1994. Integrating distributed models: the architecture of ENVISION. *International Journal of Computer Integrated Manufacturing*. 7(1):47-64.
- Jorysz, R. H. and F. B. Vernadat. 1990. CIM-OSA part 1: total enterprise modeling and function view. *International Journal of Computer Integrated Manufacturing*, 3(3) & (4):144-156.
- Kim C., K. Kim, and I. Choi. 1993. An object oriented information modeling methodology for manufacturing information systems. *Computers and Industrial Engineering*. 24(3):337-353.
- Kochikar, V. P., and T. T. Narendran. 1994. On using abstract models for analysis of flexible manufacturing systems. *International Journal of Production Research* 32(10):2303-2322.
- Little, J. D. C. 1991. Are there 'laws' of manufacturing? *Manufacturing Systems*. Washington D. C.: National Academy Press.
- Nagata, T., Y. Nagata, and H. Koshimitsu. 1993. Building a CIM system for compound plant: Utilization of distributed processing system. *Computers & Industrial Engineering*. 24(4): 561-569.
- Ngwenyama, O. K., and D. A. Grant. 1994. Enterprise modeling for CIM information systems architectures: An object oriented approach. *Computers & Industrial Engineering*. 26(2): 279-293.
- O'Sullivan, D. 1994. *Manufacturing System Redesign: Creating the Integrated Manufacturing Environment*, New Jersey: Prentice Hall.
- Pandya, K. V. 1995. Review of modeling techniques and tools for decision making in manufacturing management. In *IEE Proceedings on Science, Measurement, and Technology* 142(5): 371-377.
- Petrie, C. J. 1992. Introduction In *Proceedings of the First International Conference on Enterprise Modeling* Ed. Petrie, C. J., 1-17. Massachusetts: MIT Press.
- Ross, D. 1985. Applications and extensions of SADT. *Computer April*: 25-34.
- Savolainen, T., D. Beeckmann, P. Groumpos, and H. Jagdev. 1995. Positioning of modeling approaches, methods and tools. *Computers in Industry*. 25: 255-262.
- Wang W., K. Popplewell, and R. Bell. 1993. An integrated multi-view system description approach to approximate factory modeling. *International Journal of Computer Integrated Manufacturing*. 6(3): 165-174.
- Weston, Jr. "TED". 1994. Three dimensions of CIM, *Production and Inventory Management Journal*. First Quarter: 59-61.
- Young, R.E. and J. Vesterager. 1991. An approach to CIM system development whereby manufacturing people can design and build their own CIM system. *International Journal of Computer Integrated Manufacturing*. 4(5): 288-299.

**AUTHOR BIOGRAPHIES**

**GANESH PAI KATEEL** is a Research associate in the Center for Computer Integrated Manufacturing in the School of Industrial Engineering and Management at Oklahoma State University, Stillwater, OK. He received the B.Tech degree in Mechanical Engineering from the Karnataka Regional Engineering College, India and Post Graduate Diploma in Industrial Engineering from the National Institute for Training in Industrial Engineering, Bombay, India. He has two years experience in the CNC machine building industry. He has taught both at undergraduate and graduate levels at Karnataka Regional Engineering College, India for four years. He was the Project Officer for the State of Karnataka, National Technical Manpower Information System, Govt. of India from 1990 to 1994. Currently he is completing his M.S. degree in Industrial Engineering and Management at Oklahoma State University. During 1995 - 96, he taught Production Management at the College of Business, Oklahoma State University.

**MANJUNATH KAMATH** is an Associate Professor in the School of Industrial Engineering and Management and Director of Center for Computer Integrated Manufacturing at Oklahoma State University, Stillwater, OK. He received the B.Tech. degree in Mechanical Engineering from the Indian Institute of Technology, Madras, India, in 1982, the M.E. degree in Automation from the Indian Institute of Science, Bangalore, India, in 1984, and the Ph.D. degree in Industrial Engineering from the University of Wisconsin-Madison, in 1989. His primary areas of interest are stochastic modeling and queueing theory, analytical performance modeling of manufacturing systems, object-oriented modeling and simulation of discrete-event systems and Petri nets. He is a member IEEE, IIE, and INFORMS.

**DAVID B. PRATT** is an Assistant Professor in the School of Industrial Engineering and Management at Oklahoma State University, Stillwater, OK. He holds B.S., M.S., and Ph.D. degrees in Industrial Engineering from Oklahoma State University. He has twelve years experience in the petroleum, aerospace and pulp and paper industries. His research and teaching interests include manufacturing systems modeling, enterprise integration, and the strategic implications of CIM. He is a registered Professional Engineer in Oklahoma, a certified Fellow in Production and Inventory Management, and an ASQC Certified Quality Engineer. He is a member of IIE, NSPE, APICS, INFORMS, and ASQC.