



ENTERPRISE MODELING FOR CIM INFORMATION SYSTEMS ARCHITECTURES: AN OBJECT-ORIENTED APPROACH

OJELANKI K. NGWENYAMA¹ and DELVIN A. GRANT²

¹School of Business Administration, University of Michigan, Ann Arbor, Michigan and ²College of Business Administration, Rochester Institute of Technology, Rochester, New York, U.S.A

(Received for publication 23 November 1993)

Abstract—Computer Integrated Manufacturing (CIM) information systems have become extremely important to the global competitiveness of most manufacturing firms. And although much has been written about CIM development and implementation many problems still plague practitioners. For example, lack of integration, islands of automation, sub-optimization of resources, and the inability to migrate to future technology are a few types of problems. Further, few methodologies have been developed that systematically address these issues. In this paper, we outline an Object-Oriented approach to modeling manufacturing enterprises. The approach offers a procedure and techniques for defining CIM information systems architectures which could serve as blueprints for CIM development and implementation in manufacturing enterprises.

1 INTRODUCTION

As never before, computer integrated manufacturing information systems (CIMIS) architects and planners are faced with the pressures of plotting a course for CIMIS development and use, to meet the competitive and survival needs of world class manufacturing enterprises in the 1990s. They must understand the goals, objectives, and capabilities of their firms and develop strategies to attain these goals. This is no easy task, as the path to efficient and effective CIMIS is plagued with problems and often ends in failure. For several years, researchers and practitioners have been struggling to identify and document some of the major problems facing CIMIS development and implementation (see Table 1), but few integrated approaches have emerged for solving them. For the CIMIS architect and planner, the problems of charting a course for the manufacturing firm still remain. And although, progress has been made over the years, there are few methodologies to assist with the systematic planning and development of these complex systems (Flatau, 1988; Ciampa, 1988; Nalder and Robinson, 1987; Banerjee, 1986). This lack of methodologies has led to *ad-hoc* development practices that are associated with several of these organizational problems (Yeomans *et al.*, 1986; Ciampa, 1988; Gunn, 1987; Solberg, 1988; Flatau, 1988; Levvilis, 1985).

Computer integrated manufacturing incorporates a wide range of information technologies, such as, electronic data processing (EDP), management information systems (MIS), decision support systems (DSS), expert systems (ES), computer-aided design (CAD), computer-aided manufacturing (CAM), computer-aided process planning (CAPP), flexible manufacturing systems (FMS), etc. Developing, implementing and operating manufacturing systems with these technologies, in the absence of appropriate planning and development methodologies, is an intractable problem. We believe that enterprise modeling methods and techniques developed to help solve organizational information management can provide assistance with many of the CIM problems cited above.

In this paper, we present a case-study of the application of an Object-Oriented Enterprise Modeling (OOEM) approach to CIMIS planning and development. The approach focuses on systematically modeling the global information requirements needed to plan, design, and develop information systems for a CIM environment. The OOEM approach provides methods, technologies, a philosophy and procedure for identifying and representing the CIM processes, information flows, and data entities that are relevant to the goals and objectives of the manufacturing enterprise. We argue that this approach assists in narrowing the gap between planning and implementation,

Table 1 CIM planning and integration problems

- | |
|---|
| (1) Lack of Integration Methods and Approaches |
| (2) CIM as an Emerging Field |
| (3) <i>Ad-Hoc</i> CIM Development |
| (4) Lack of Standards |
| (5) Lack of Understanding of CIM Variables |
| (6) Bottom-Up Approaches—Lack of Pre-Defined Architecture |
| (7) Island of Automation Paradigm |
| (8) Confusing and Narrow Definitions of CIM |

and provides a conceptual view the information requirements, critical interfaces, and functional dependencies across the organization (Seifert and Settles, 1989; Kroenke, 1989). It also offers an alternative to *ad-hoc* CIMIS development, accommodates incremental development of systems, and supports the integration process.

The rest of the paper is organized as follows: Section 2 discusses enterprise modeling (EM) and provides an overview of object-oriented concepts and terms, and discusses the advantages of applying these concepts to EM. Section 3 provides a case describing an application of the OEM approach in a manufacturing firm. And finally, Section 4 concludes with a brief discussion of the lessons learned.

2 THE CIMIS PROBLEM SPACE

The basic objective of CIM is to achieve enterprise-wide cost efficient and effective manufacture and distribution of the firm's products. A fundamental component of CIM is the information systems which support integration the core activities of different management levels, (*strategic, tactical and operational*), and across business functions (purchasing, design, production, distribution) important to manufacturing. The successful manufacturing enterprise is a highly coordinated complex of *Task, Technology, People, Communication, and Structure*. Several studies have pointed out the importance of synergy among these aspects of the manufacturing enterprise (Leavitt, 1965; Galbraith, 1977; Yadav, 1983; Grant, 1991; Grant *et al.*, 1992). Since the development and implementation of CIMIS is a change process that intervenes in the manufacturing enterprise, close attention must be given to each of these aspects. Figure 1 shows the conceptual relationships among the various organizational aspects and the levels of management in the manufacturing enterprise. In order to convey the complexity of the CIMIS problem space, we will briefly outline the organizational aspects that are important for the development and implementation of effective CIMIS.

Task. From one perspective, organizations may be conceptualized as a complex of interrelated sets of tasks. These tasks may be performed by people, computers or machines working independently or in collaboration. *Technology* has become an essential component of modern manufacturing. Intense competitive pressures are forcing more manufacturers to adopt a wider

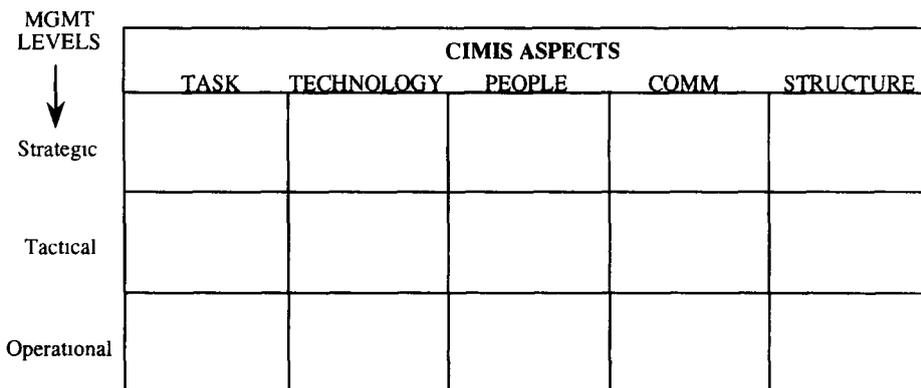


Fig 1 CIMIS framework

variety of manufacturing technologies than ever before. These technologies often impact the organization structure, its task components, and human users. The efficient and effective integration and application of new technologies to the manufacturing process is vital to the survival of the firm.

People. In as much as automation has advanced considerably, people remain vital to manufacturing. They are required to manage the manufacturing process, monitor and control computers and machines. Consequently, the human dimension should be carefully considered during the design of any human activity system. Peoples' skills, competence, and training are important parameters when deciding on the tasks to be performed, the technology to be used, the type of user interface to use, or the type of organizational structure to adopt.

Communication. Two types of communication are important considerations for CIMIS development: technical and human. Technical communication concerns the ability of different types of technology to effectively communicate. Human communication concerns the ability of people to communicate with each other, computers, and machines. Technical communication is very important for CIMIS development because CIMIS attempts to integrate different types of IS, CIM technologies, and people. This integration is achieved by linking these different technologies via computer networks. Failure to address these human concerns has led to implementation problems which contribute to CIMIS failures.

Structure. Structure is concerned with organizing how people, technology, and task complexes are arranged to achieve specific objectives of the enterprise. The type of technology (i.e. tools, techniques, and processes), and the manufacturing and management philosophy of the firm are important parameters in determining organization structure. With regard to CIMIS, there is another important aspect of structure: the configuration and design of technology complexes. To effectively control and monitor an integrated CIMIS, levels of computer and machine control and monitoring are necessary.

2.1. The Enterprise Modeling Approach

Enterprise modeling provides methods, tools, techniques, and a philosophy for describing and analyzing relevant aspects of the business enterprise, and deriving a conceptual architecture upon which the development and implementation of CIMIS could be based. The architecture is a set of conceptual models representing key features of a business enterprise (such as functions, processes, activities, role responsibilities, information flows, data structures, etc.) that are relevant to global planning, analysis, specification, and development of IS. EM provides a top-down view of the enterprise's information requirements and a "road map" for guiding the development of integrated enterprise-wide information systems (Martin, 1983). There are several approaches to EM for information systems architecture development: BSP (IBM, 1984), BIAIT (Carlson, 1979), SASS (DeMarco, 1979), IE (MacDonald, 1986; Martin, 1989), SADT (Ross *et al.*, 1977, McGowan and Marca, 1988), each with its own strengths and weaknesses. However, the approach that we are proposing is based on the Object-Oriented Paradigm which has simplicity as its greatest strength.

Recently, researchers and practitioners have moved toward an object-oriented approach for information systems modeling, analysis, and design (Iivari, 1991a, b; Bailin, 1989; Bulman, 1989; Hoza *et al.*, 1989; Ward, 1989). The primary motivation for this paradigm shift is the recognition that the object-oriented approach offers powerful but simple concepts for representing organizational reality and the system designer's ideas about the system of interest (Agha, 1985; Norman, 1991a, b; Iivari, 1991b). As Banerjee *et al.* (1987) see it, the "objects", the basic concept of this approach, can be used "to represent anything from a simple number . . . to a complex entity, such as an automobile or an insurance agency."

2.2. Object-oriented Modeling

Several approaches to object-oriented modeling of information systems have been suggested (Bailin, 1989; Bulman, 1989; Chen and Nunamaker, 1989; Hoza *et al.*, 1989; Shlaer and Mellor,

1988; Ward, 1989, Coad and Yourdon, 1990), each having individual strengths and weaknesses. Our approach, however, is informed by Iivari (1991a, b), who suggests a framework and a strategy for modeling information systems which fit well within the basic objectives of enterprise modeling and architecture definition. The framework identifies three levels of abstraction in object-orienting modeling which are important to integration in CIM: (1) the *organizational* level defines the context of the information system and its role in the organization; (2) the *conceptual* level defines the functional architecture of the information system independent of technical and implementation considerations; and (3) the *technical* level defines the technological architecture of the information system upon which plans for its implementation are based. Before we discuss the modeling strategy, we will outline the basic concepts to be used.

2.2.1. Object-oriented concepts

The fundamental concept of the object-oriented paradigm is the *object*. Generally, objects may be characterized as active or passive. *Active objects* can operate on and transform *passive objects*. The operations that transform passive objects are specific properties of active objects called *methods*. Methods are sets of procedures or rules for bringing about specified state changes in passive objects. Relevant examples of: (a) *active objects* are business processes and activities, computer hardware, and manufacturing tools; (b) *passive objects* are data, manufacturing material etc.; and (c) *methods* are business and manufacturing procedures, design procedures, etc. Objects which share similar characteristics and enact or respond to similar methods are considered the same *class*. For example, we can represent all activities that have to do with the design of a specific product, say a compressor, as a *class* of compressor design objects; and at a higher level, all activities having to do with design in general as design objects. It should be clear that all compressor design objects share some similarities with the higher level design objects. In object-oriented language, we would say that these similarities are *inherited* from the higher level design objects. Finally, active objects communicate with each other via *messages*. These *messages* could signal events, request operations to be performed, or report on operations completed.

2.2.2 Architecture definition

Enterprise modeling for CIMIS architecture definition is a three phase process: (1) Requirements Definition, sometimes called business modeling; (2) Conceptual Design; and (3) Technical Design. In the requirements definition phase, the focus is on defining the role which the CIMIS must play in supporting the manufacturing enterprise. The information it must provide and its organizational context are defined via the modeling process in which the Global, Business, Information Flow, and Responsibility models for the functional area are developed. The Global Model defines the wider context, within which the organizational function to be supported by the CIMIS is situated. It defines the main interactions and information flows between the function of interest and other organizational functions. The Business Model is a representation of all the relevant aspects (processes, activities, information flows, roles, etc.) of the function of interest, and their relationships. The conceptual design phase consists of a set of activities concerned with defining the functional architecture of the CIMIS independent of technology considerations. The Data, Database, and Application models are developed in this phase. Essentially, the Conceptual design specifies the specific information (reports, screens, etc.) to be provided in support of the various work activities, the information processing procedures (i.e. Application Model), and the database schema and definition of subject processing databases. The Technical design phase is concerned with identifying and analyzing different technological options for implementing the CIMIS. Computer and communications hardware and software options are identified and studied, and an appropriate cost effective, efficient, and reliable technical architecture is defined. In this phase, the designers must take into account existing hardware and interface requirements in order to optimize resource utilization and throughput.

Three other important phases of activities must follow the CIMIS architecture definition if the gains of integration and system optimization are to accrue. They are: planning, systematic development and implementation. These are not the focus of this paper, but we will briefly discuss what was accomplished in these phases during the case study.

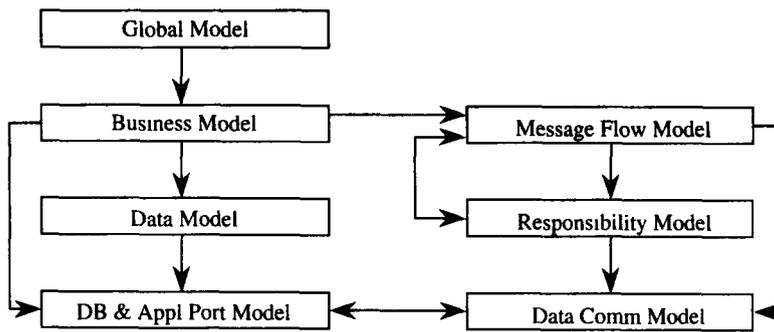


Fig 2 Overview of CIMIS architecture models

2.2.3. The modeling procedure

As stated above, eight conceptual models are used in deriving the CIMIS architecture: Global Model, Business Model, Information Flow Model, Responsibility Model, Data Model, Application Model, and Technology and Data Communication Model. The strategy for deriving the models is iterative and loop-linear (cf. Fig. 2). The iterations are important because the development of some models required information derived from other models. The arrows in the center of each box indicate the primary flow for developing the seven models of the architecture. Because the development of a single model may be influenced by or dependent upon another, the arrows shown on the side of the boxes indicate the secondary flows. Therefore, for some models to be completely specified, both the primary and secondary flows are needed.

The procedure for deriving a model is as follows: (1) identify candidate objects, (2) classify them as active or passive, (3) define their characteristics, (4) define the methods they enact with or which operate on them, (5) define the messages that they send or receive, (6) construct the diagrams. In enterprise modeling, identifying and classifying objects of interest could be a difficult task. Figure 3 is a general framework for identifying and classifying objects in the problem domain. Although this framework is not comprehensive, it provides several examples that will assist inexperienced analysts, who might be modeling complex manufacturing enterprises, to quickly identify and classify relevant objects.

| Levels of Modeling | OBJECT-ORIENTED CONCEPTS | | |
|--------------------|--|-------------------------------------|--|
| | Objects | Messages | Methods |
| Organizational | Business Process Business Activities Organizational Roles Information Views | Information Flows Physical Flows | Business Procedures Decision Making Procedures Business Policy |
| Conceptual | Information Handling Processes Communication Processes Data Subject Databases | Data Flows | Information Processing Procedure Communication Procedures |
| Technical | Information Systems Database Structures Hardware Systems Software Communication System | Data Flows | Application Programs Operating Procedures |

Fig 3. Conceptual framework for identifying objects

3 CASE ILLUSTRATION

The domain of this case study is the Engineering Release Function (ERF) of a major manufacturing firm located in a northeastern state of the U.S.A. The ERF is responsible for pre-manufacturing activities which focus on the processing, management, and distribution of engineering design information. It is a specialized information resource management function which serves as the interface between engineering design and manufacturing. The Engineering Release Function is comprised of six departments (D1–D6), eight managers, and over 100 employees, most of whom are engineering and IS professionals.

Each department is responsible for a distinct set of information processing activities. D1 assembles engineering design data; and develops, generates, and distributes reports to other departments. D2 further processes the design data by adding more manufacturing process information and by instruction. D3 creates process routings and their accompanying data. D4 and D5 collaborate on generating artwork, NC data, and manufacturing materials requirements information. The artwork information is a detailed description of the various features (holes, lands, etc.) that make up the “card”, the image of the printed circuit boards which will be produced in the manufacturing stage. The information is produced by design engineers who work at various geographically dispersed Research and Development Labs. The design data are transmitted over a wide-area communications network to the Engineering Release Function. D6 ensures the manufacturability of the pre-manufacturing product and is concerned with issues of “design for manufacturing”.

The various types of information produced by the ERF are called pre-manufacturing products. These products are inputs to the Manufacturing Function which produce intermediary products (glass prototypes) and final products (printed circuit boards). The glass prototypes serve as templates for the manufacture of the printed circuit boards.

3.1. Project Team Structure and Process

A multi-disciplined project team of 15–20 members from the 6 departments worked continuously on the EM project. The team consisted of IS specialists, pre-manufacturing experts, 3 managers from the ERF, experts from design engineering and manufacturing, and external specialists (action researchers). The skill and experience of team members varied. The experience of application programmers ranged from 10 to 25 yr, while the experience of the expert users ranged from 15 to 25 yr. Many expert users had worked most of their careers in Release Function. The experience of lower level managers ranged from 7 to 16 yr. Several had worked as application programmers in other business functions before joining the ERF. The primary responsibility of pre-manufacturing and other experts was model development and validation. They worked closely with the IS professionals to ensure the models adequately represented the functions and activities within the universe of discourse. The responsibility of IS professionals was application design, development, and implementation. The primary responsibility of the analyst was to model the Release Function and to serve as an interface between programmers and users. It was the responsibility of the managers to administer the project, i.e. provide financial and other resources required by the project and report back to higher level management.

The development team was divided into subgroups of 3–4 members to address specific problems. Members who were most knowledgeable about particular aspects of the manufacturing process usually ended up on related projects. Team responsibilities ranged from requirements analysis to implementation. Special teams, called action teams, were responsible for conducting fact finding missions about obscure aspects of the business. Weekly meetings were held to allow action teams to report back to the project team and to management on the status of project activities.

3.2. Requirements Definition Phase

The action teams’ fact finding missions were part of the requirements phase. They were initiated from discussions that emerged from the development and analysis of the model of the ERF. A decision was made to model the ERF in order to aid the understanding of the problem domain.

and provide insight for problem resolution. One analyst commented, "it's amazing how much I have learned from the modeling exercise." He went on to say, "this level of understanding could not have been achieved in the absence of the model." One manager commented, "probably the most important product of the modeling exercise is the process (i.e. the learning and understanding that occurred among project team members), and not the product (i.e. the actual model)." The model served as a communication vehicle for debate and discourse among team members. By analyzing aspects of the model and with further analysis, the lower level requirements such as data object, mini specs, application programs, database design and so on were derived.

The results of modeling the case-organization are now presented in summary. It is not practical to present in detail all the models we developed because this would require several pages and lead to information overload. Consequently, we present high level representations to give the reader a feel for the process. For example, in the case of the Object Business Model, we will present the first level of decomposition and omit the remaining six lower levels which are several pages long.

3.3. *The Enterprise Models*

In this section, we illustrate an approach to Object-Oriented Enterprise Modeling by using the case description and discussion of the models. The order in which the models are presented is as follows: (1) Global Model, (2) Business Model, (3) Message Flow Model, (4) Responsibility Model, (5) Data Model, (6) Database Model, and (7) Technology & Data Communication Model. We do not intend to justify the need for such views but instead have accepted them as given because of their wide acceptability and use (cf. Carlson, 1979; IBM, 1984; Martin, 1984; MacDonald, 1986).

3.3.1. *Global model*

The Global Model is used to define the universe of discourse of our modeling exercise. It provides the foundation for integrating the enterprise and for developing the remaining enterprise models. The primary objective of the Global Model is to capture, at the highest level of abstraction, the interactions among the organization systems and/or subsystems of interests. The model shows the primary information links (i.e. messages) upon which the systems are to be integrated. The messages represent the natural flow of information among the various systems. Developing the Global Model required the definition of major operating systems, also considered objects, that are embedded in the enterprise without regard to artificial boundaries such as departments, and defining the primary messages that are passed between them. Figure 4, below, is a representation of the Global Model of our host company. It describes the organizational context of the ERF, and its interactions with other organizational functions within the manufacturing enterprise. As can be seen, the Engineering Release function receives its primary inputs (product descriptions) from the Product Design and Development function, and generates outputs to various functions within manufacturing (e.g. Order Release, MRP, and so on).

3.3.2. *Business model*

The Business Model is derived by decomposing each subsystem (i.e. object) into its set of related activities/processes (i.e. lower level objects). The Business Model is a representation of the set of tasks/activities that are performed to support some aspect of the business. It documents the minimum set of activities or tasks that must be performed if the organization is to function efficiently. Moreover, it serves to identify information flows (messages) that are necessary to support organizational tasks. A model of the existing system is developed and analyzed to identify missing activities (information/message flows) to add process control mechanisms or to eliminate redundant and unnecessary activities. This analysis serves to eliminate, from the Business Model, the complexities that have been built into the business over the years (Clark, 1989). The objective is to simplify the Business Model and increase its efficiency before attempting automation. Figure 5, the Business Model, is a high level description of the six major processes within the ERF and the primary informational interactions. It must be stated that this is the highest level of abstraction of the Business Model. Each process represented in the Business Model is further decomposed at lower levels to identify and represent its individual tasks and their information needs.

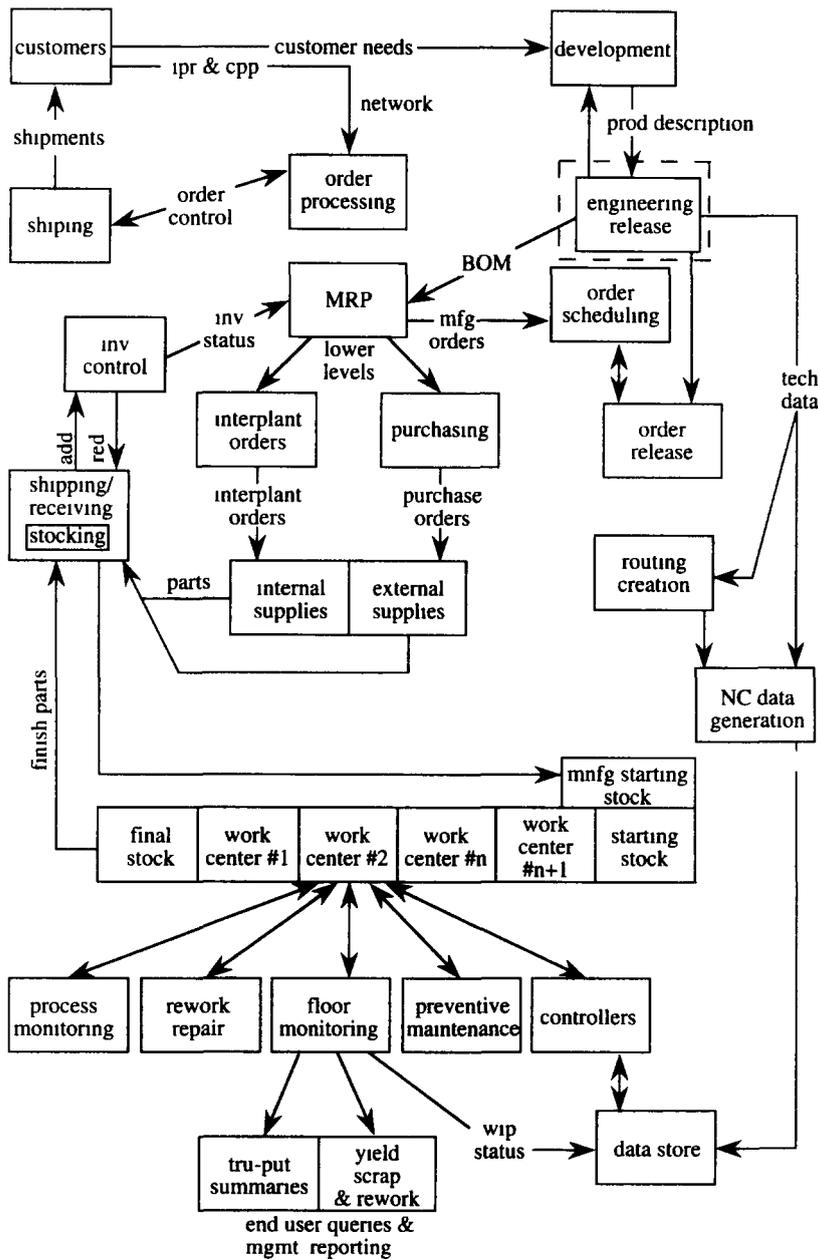


Fig 4 Global model of the manufacturing enterprises

3.3.3. Message flow model

The Message Flow Model describes information flows (messages) that are received or generated by activities or functions at different levels of abstraction. The natural starting point for constructing the model is the Business Model. The interactions among the business activities are documented in detail, describing the content of information exchange and timing. Figure 6 is a summary of the primary information flows among the processes depicted in the Business Model above.

3.3.4. Responsibility model

The Responsibility Model documents the role responsibility for information processing at various levels of the organization. This model makes explicit the organizational role relationship

and control of information. It documents which objects (individuals or machines) have access or control over which messages. Figure 7 is a Responsibility Model showing a small set of activities of the Engineering Release Function. On the left side of the figure the role is identified, and on the right the information that is either used or generated is identified. The legend and an example below Fig. 7 illustrates how the data on the right of the figure should be interpreted.

3.4. Conceptual Design

Conceptual design is the second phase of the EM process following requirements definition. It is the process of constructing the detail architecture models (Data Model, Database and Application Models, and the Data Communication Model) that are independent of implementation details, such as target communications network and database management systems. The primary input to conceptual design is the information requirements derived from the Business Model.

3.4.1. Data model

The Data Model is a description of the data which are used and generated by the tasks and their relationships. The data model is used to derive database schema during database design. To develop

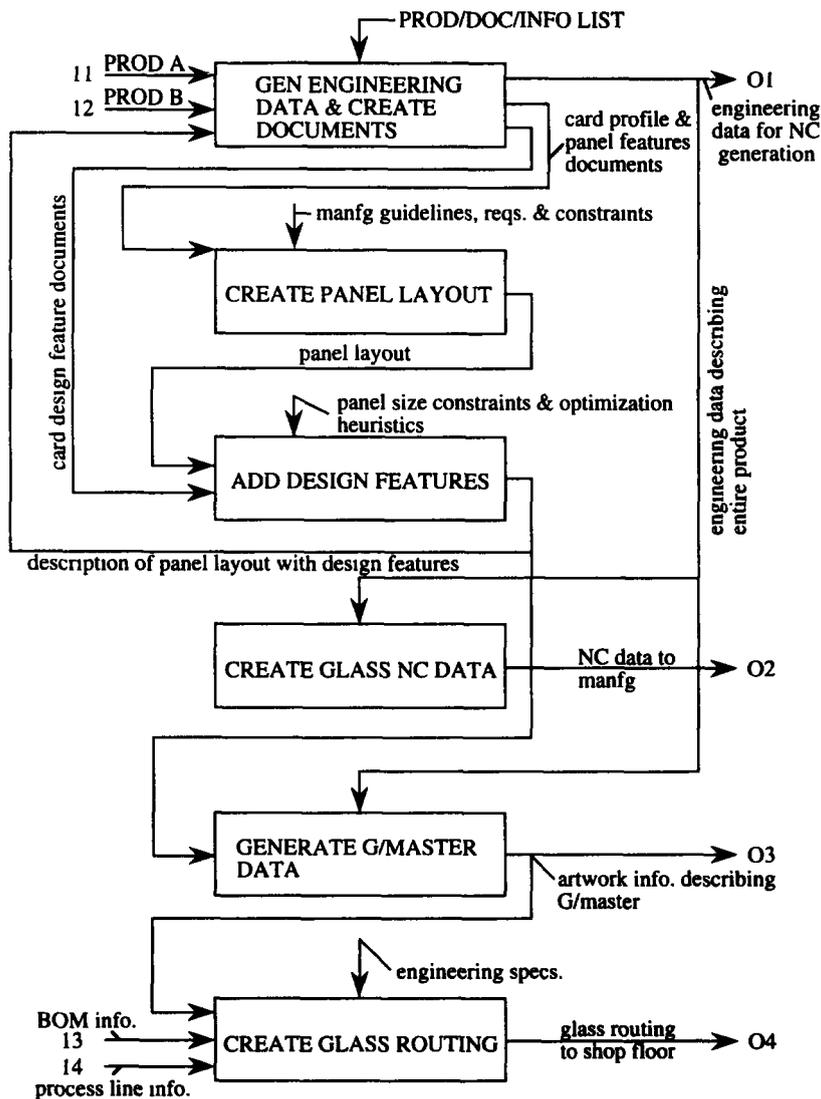


Fig. 5 Business model of the engineering release function

| INPUT | ENGINEERING RELEASE FUNCTION | OUTPUTS |
|--|------------------------------|--|
| Product descriptions single image Tech. data | Design Process | Product design data |
| | Design Process | Restructured engineering data |
| Prod description unit image | Product Process | Panel layout info |
| Process line info | Artwork Process | Artwork descriptions Artwork instructions |
| Machine availability | Routing Process | Routing info to shop floor |
| Process line info. | Routing Process | " |
| Cost info | Routing Process | " |
| Artwork description of product | Routing Process | " |
| Restructured engineering data | NC Process | NC data to manufacturing |

Fig 6 Object message flow model information flows among processes of the engineering release function

the data model, the analyst in collaboration with users, focused on the Business Model activities to identify the necessary supporting data objects. This exercise is aided by documents such as memos, forms, and reports used by individuals who perform the activities. A list of object classes is compiled and analyzed to remove redundancies and resolve naming problems. Depending upon the number of objects, clustering may be necessary to arrive at logical sets of objects (i.e. object classes and subclasses) which may be represented in several databases. The final step is to model each subclass of objects using an object-oriented data model. In developing the Data Model, the concept of inheritance can be utilized. For example, the root of the data model can exhibit behaviors common to an object class or subclass while lower level objects exhibit more specialized behaviors. Figure 8 shows the basic data and the relationships that must be managed in support

| Role | Information Flows |
|---|---|
| Operator Who Generates Engineering Data | Description of Product A (u) A1[1]I1 Description of Product B (u) A1[1]I1 Tech Data (u) A1[1]I2 Panel Layout with Design Features Description (u) A1[1]I3 Engineering Data Passed to NC Type Database (c) A1[1]O1 Information Needed to Decide Type of Docs to Gen (u) A1[1]O3 = I1[2] Engineering Data Req. to Create Docs (u) A1[1]O2 = I1[3] Decision Concerning Type of Docs to be Gen (c) A1[2]O1 |
| Operator Who Generates Documents: | Engineering Data Req. to Create Docs (u) A1[3]I1 Decision Concerning Type of Docs to be Gen (u) A1[3]I2 Created Documents (c) A1[3]O1 = A1[4]I1 |
| Operator Responsible For Verifying and Distributing Documents | Created Docs. (u) A1[4]I1 Card Profile & Panel Features Docs (c) A1[4]O1 Card Design Features Document (c) A1[4]O2 |
| Panel Designer: | Info About Panel Features (u) A2[1]I1 & A2[2]I2 Card Overall Dimensions (u) A2[3]I2 Card Panel Layout Information (u) A2[4]I1 Acceptable Panel Layout (c) A2[4]O2 List of Panel Sizes (u) A2[2]C1 Manufacturing Constraints (U) A2[2]C1 Manufacturing Guidelines for Panel Layout (u) A2[3]C1 |

Fig 7. Responsibility model for some engineering release activities Legend name of info + (use or create) + node# + [process#] + input# or output#
Example tech data + (u) + A1[1] + (I1) Note the A1, [1], and (I1) refer to the Business Model

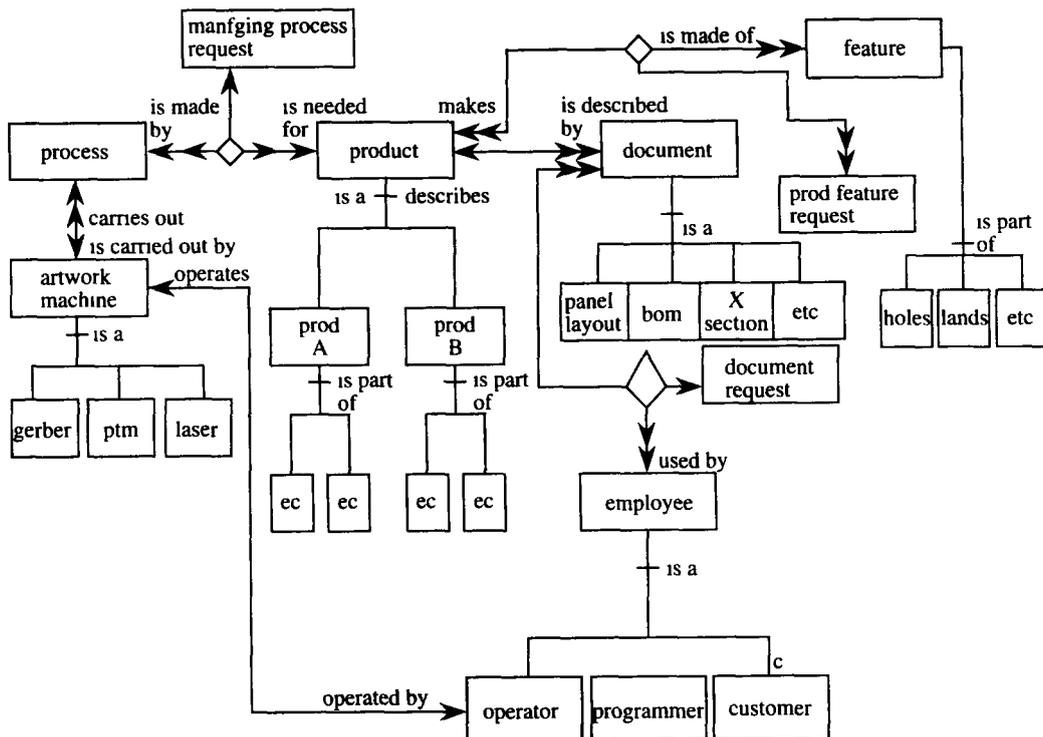


Fig 8 Data model

of the work done by the Engineering Release function. These data and their relationships will have to be maintained in the CIMIS database.

3.4.2. Database model

The Database Model, Fig. 9, describes the set of subject databases needed to support the business activities of the organization. It also identifies the application programs that should run against each of the databases. The data clustering done in the earlier data modeling phase now becomes the basis for deciding which database(s) to develop. Each cluster of objects (i.e. subclass of objects) may give rise to the design of separate subject databases (Martin, 1989; Weatherall, 1989). With further analysis, the information flows between databases may be derived. Based upon the type of information needed to support some aspects of the business, the team, with the help of users and managers, could then decide upon the application software required to perform business functions

3.5. Technical Design

The main objectives of the technical phase are: (a) to define the technological requirements in terms of hardware and software for the CIMIS, and (b) to develop alternatives for analysis and selection of the technology to be implemented. The output is a Technology and Communication Model for the CIMIS.

3.5.1. Technology and data communication model

This model describes the technology to be used and how they are linked via a data communication network. It makes explicit which systems are to communicate with each other and how the communication, i.e. message passing, should be handled. This provides support for the analysis and selection of the network, the software required, the communication protocols and standards to adopt, and the general topology of the communication network (Heid, 1986). This

allows for the pressing concerns mentioned above to be addressed early in the project development process.

To derive the Technology and Data Communication Model the team made an assessment of the type and quantity of data, the distance over which the data is to be transmitted, the performance of the network, the reliability of the network, and so on. In light of these concerns, the technology was chosen. The operating systems (OS) were chosen so they are compatible. This can seem trivial, but it may have severe implications. For example, two operating systems are to communicate with each other, and one supports 16 decimal places while the other supports twelve. As data are passed back and forth between the two systems, 4 places of accuracy is lost. This may provide sufficient error to cause a fatal accident in critical situations. In other situations, it is sufficient to corrupt the data and cause data integrity problems. The next step is to decide on the best local area network (LAN) configuration. This involved decisions regarding the reliability of the network, the performance, and so on. Figure 10 is a description of the technology configuration defined as the platform upon which the proposed CIMIS for our host organization was to be developed and implemented.

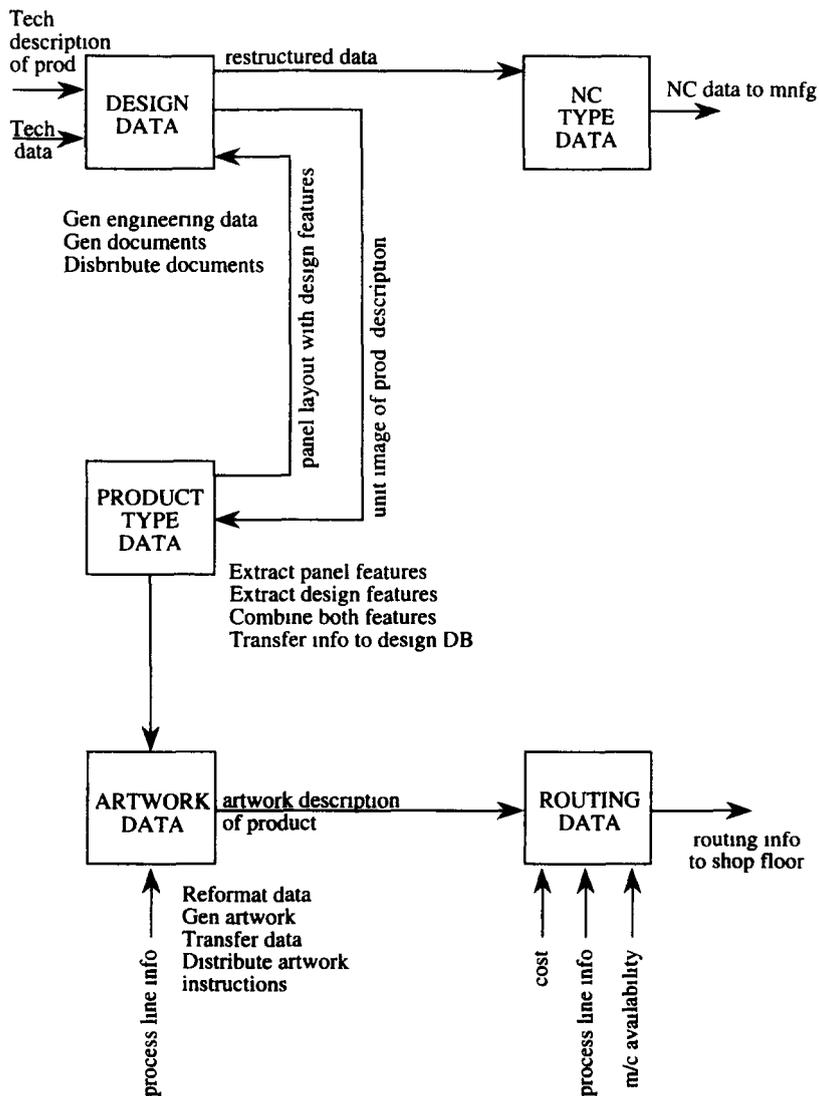


Fig 9 Database model for CIM database processing

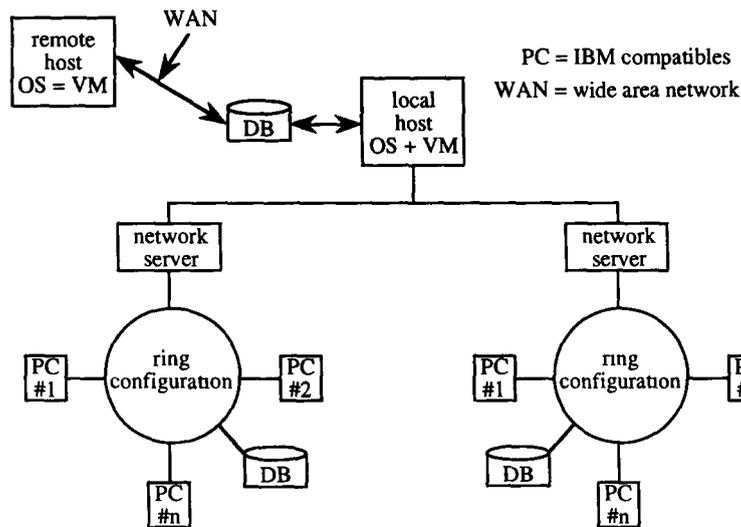


Fig 10 Technology and data communication model

3.6. Planning Phase

From the information gathered in the requirements definition, and the conceptual and technical design phases, a strategic plan and several tactical plans were developed. The strategic plan had a 5-yr horizon of implementation. The goal of which was to reduce the cycle-time from 12 days to 4 hr. This strategic goal led to the identification of several tactical plans. For each tactical plan one or more development projects were evaluated to ensure that they were congruent with the overall strategic goal. A list of the information systems projects is shown below.

Projects:

- (1) an on-line document distribution system;
- (2) a work-in-progress tracking system;
- (3) an automatic manufacturing process routing generation system;
- (4) a real-time document and data generating system;
- (5) a comprehensive pre-manufacturing product generating system.

3.7. Development and Implementation Phase

A brief comment on the development and implementation: at the present time the first four projects have been completed, and the fifth is in progress. The on-line document distribution system (DDS-1), electronically distributes documents to individuals via a computer network. This replaces the printing and distributing of hard-copy documents which was slow, time consuming and costly. The work-in-progress tracking system (WIPS), track jobs as they move through the enterprise. It records the time, place, and the person responsible for the job, and supports a level of accountability which did not exist before its implementation. The automatic process routing generation system (ARS-1) eliminates or minimizes levels of human intervention previously required to generate some of the manufacturing process routings for machines on the shop floor. This has helped to decrease cycle-time by reducing the time required to generate routings, and has reduced errors associated with the routing generation process. The real-time document and data generation system (DDGEN) replaces an inefficient batch-run system which operated on a 24-hr cycle. The new system allows for real-time response. This has also had an impact on productivity, and the reduction of cycle time. The comprehensive pre-manufacturing product generating system is a major development project which necessitates the integration of several different information technologies, and has implications for process re-engineering. Development is underway, and is expected to be completed by 1995.

4 CONCLUSION AND DISCUSSION

Object-oriented concepts provide a new ontology (i.e. a way of viewing the world) for conceptualizing information systems development, specifically enterprise modeling. We have illustrated the usefulness of object-oriented concepts (objects, inheritance, messages, instance variables and so on) and applied them to modeling the enterprise. We have found them useful for addressing certain issues and problems related to EM. For instance, by viewing the enterprise as a related set of objects reduces complexity, decreases the semantic gap, and reduces the planning–implementation gap. With the increasing availability of object-oriented database management systems, additional benefits will be gained. We will now discuss some important lessons learned from the modeling exercise

It was evident from the case study that organizational learning is the most important outcome of our exercise. The process of developing a single view of the organizational reality that satisfies all the stakeholders is extremely difficult. In any modeling activity where the analyst has to work with several users, s/he must confront several notions of reality. The problem is how to reconcile these often conflicting views of reality. We found that having a model in a format that every member of the group could understand allowed for debate where members can challenge apparently conflicting views of reality, eliminate misconceptions, and learn of other valid perspectives. By challenging existing views, fruitful discussion and, at times, heated debate occasionally resulted among team members. Participants of the modeling process have claimed that the critical process yielded a level of organizational learning that could not have been achieved in the absence of the model. The critical debate process helped to identify limitations of the existing business. At first, individuals were hesitant to acknowledge that limitations did exist. With the passing of time, team members became convinced and committed to correct such limitations. As a result, global objectives were defined and a 5-yr plan was developed to implement changes which would improve the efficiency and the effectiveness of the design, engineering release, and manufacturing processes.

Related to issues of organizational learning is the need to foster collaboration among team members. It became apparent during the modeling process that methods and techniques to support collaborative group dynamics are needed. Despite attempts to foster camaraderie among the team members, issues emerged which were not effectively dealt with because we had no group collaborative process structure. At times, the project stalled on issues of politics and competition among team members. It is important that the analyst be trained to better handle the group dynamics that normally emerge during project development. Another limitation faced during our action research was the lack of computer support for the modeling and analysis. Top management decided not to purchase new software tools. Consequently, we were forced to use CASE tools designed for less complex modeling exercises.

REFERENCES

- Agha G. A model of Concurrent computation in distributed systems. MIT AI Lab (1985)
- Bailin S. C. An object-oriented requirements specification method. *Commun ACM* 32(5) (1989)
- Banerjee J. et al. Data model issues for object-oriented application. *ACM Trans Office Information Systems* 5(1) 3–26 (1987)
- Banerjee S. K. Information systems design for CIM—A methodology. *Int Conf Computer-Aided Production Engineering* (Edited by J. A. McGeough), p. 347. Edinburgh (April 1986)
- Bulman D. M. An object-based development model. *Computer Language* (August, 1989)
- Business Systems Planning. Information Systems Planning Guide, IBM, GE20-0527-4 (1984)
- Carlson W. M. Business integration analysis integration technique—the new horizon. *Data Base* 10 (4), (1979)
- Chen M. and Nunamaker J. F. Jr. Integration of organizational and information system modeling: an object-oriented approach. In *HICS-22*, Vol. IV (1989)
- Ciampa D. *Manufacturing's New Mandate*. Wiley, New York (1988).
- Clark J. MRP to MRP II to JIT to CIM. The Journey. Technical Paper MS89-427, presented to SME, Michigan (1989)
- Coad P. and Yourdon E. *Object-Oriented Analysis*. Prentice-Hall, Englewood Cliffs, N.J. (1990)
- DeMarco T. *Structured Analysis and System Specification*. Prentice-Hall, Englewood Cliffs, N.J. (1979)
- Flatau U. Designing an information system for integrated manufacturing systems. In *Design and Analysis of Integrated Manufacturing Systems* (Edited by D. W. Compton). National Academy Press, Washington, D.C. (1988)
- Galbraith J. R. *Organizational Design*. Addison-Wesley, Reading, Mass. (1977)
- Grant D. An Approach To CIM Information Systems Architecture Definition. PhD Dissertation, SUNY-Binghamton (1991)

- Grant D, Ngwenyama O and Klien K Modeling for CIM information systems architecture definition *Computers Ind* 18(2) (1992)
- Gunn T *Manufacturing for Competitive Advantage*. Ballinger, Boston, Mass. (1987).
- Heid G. *Data Communication Network Devices*. Wiley, New York (1986)
- Hoza B J, Smith M. K and Tockey S. R An introduction to object-oriented analysis. In *Proceedings of the Fifth Structured Techniques Association Conference*, Chicago, May 8-11 (1989).
- Iivari J Object-oriented information systems analysis. a framework for object identification *Proceedings of the 24 Annual Hawaii International Conference on Systems Sciences* (1991).
- Iivari J Object-oriented design of information systems: the design process In *Object Oriented Approach in Information Systems* (Edited by F Assche, B Moulin and C Rolland) North Holland, Amsterdam (1991)
- Kroenke D *Management Information Systems* Mitchell, Santa Cruz, Calif (1989)
- Laagland P T M Modeling in Information Systems Development, Doctors, Dissertation, Vrije Universiteit Te Amsterdam (November 1983)
- Leavitt H Applied organizational change in industry structural, technological & humanistic approaches *Handbook of Organizations* (Edited by J March), p 3 Rand McNally, Chicago, Ill (1965).
- Levulis R J CIM—a perspective *Curr Awareness Bull Manuf Technol Inf Anal* 1(1), (3) (1985)
- MacDonald I G Information engineering In *Information Systems Methodologies* (Edited by T W Olle, H G Sol and A A Verrijn-Stuart) North Holland, Amsterdam (1986)
- Marca D and McGowan C L *Structured Analysis and Design Technique* McGraw-Hill, New York (1988)
- Martin J. *Managing The Data Base Environment*. Prentice-Hall, Englewood Cliffs, N J (1983)
- Martin J *Information Engineering* Prentice-Hall, Englewood Cliffs, N J (1989)
- Nalder G. and Robinson G Planning and designing and implementing advanced manufacturing technology In *Human Side of Advanced Manufacturing Technology* (Edited by T Wall, C W Clegg and N J Kemp) Wiley, New York (1987)
- Norman R Object-oriented systems design a progressive expansion of OOA *J Systems Mgmt*, 13-16 (August, 1991)
- Norman R Object-oriented systems analysis a methodology for the 1990s *J Systems Mgmt* 32-34 & 40 (July, 1991)
- Pendleton D A BMT a business modeling technology In *Information Economics* Prentice-Hall, New York (1982)
- Ross D T *et al* Structured analysis for requirements definition. *IEEE Trans Softw Engng* 3(1) (1977)
- Seifert D and Settles S F. Are industrial engineers working on the right things in manufacturing *IE Mag* 46-47 (February, 1989)
- Shlaer S and Mellor S J *An Object-Oriented Systems Analysis Modeling The World In Data* Yourdon Press, Englewood Cliffs, N J (1988)
- Solberg J J Integrated manufacturing systems an overview In *Design and Analysis of Integrated Manufacturing Systems* (Edited by D W Compton) National Academy Press, Washington, D C (1988).
- Ward P T How to integrate object orientation with structured analysis and design *IEEE Softw* 6(2), (1989)
- Weatherall A *CIM From Fundamentals to Implementation* Butterworth, London (1989)
- Yadav S A Methodology for Modeling an Organization to Determine and Derive Information Systems Requirements Ph D Dissertation, University Microfilms Int. (1981)
- Yadav S Determining an organization's information requirements: a state of the art survey *Data Base* 3-20 (Spring, 1983)
- Yeomans R W, Coudry A and Hagen P J W *Design Rules for a CIM System*. North Holland, Amsterdam (1986)