14:09:31 OCA PAD AMENDMENT - PROJECT HEADER INFORMATION 01/05/96

Active

Project #: E-27-651 Cost share #: Rev #: 13
Center # : 10/24-6-R6807-0A0 Center shr #: OCA file #:
Work type : RES
Contract#: DDM-8957861 Mod #: BR DTD 960103 Document : GRANT
Prime #: Contract entity: GTRC

Subprojects ? : Y CFDA: 47.041
Main project #: PE #: N/A

Project unit: TEXT ENGR Unit code: 02.010.130
Project director(s):
JAYARAMAN S TEXT ENGR (404)894-2490

Sponsor/division names: NATL SCIENCE FOUNDATION / GENERAL
Sponsor/division codes: 107 / 000

Award period: 891001 to 960331 (performance) 960630 (reports)

Sponsor amount	New this change	Total to date
Contract value	0.00	312,000.00
Funded	0.00	312,000.00
Cost sharing amount		0.00

Does subcontracting plan apply ?: N

Title: PYI - DESIGN AND DEVELOPMENT OF A MANUFACTURING ENTERPRISE ARCHITECTURE

PROJECT ADMINISTRATION DATA

OCA contact: Jacquelyn L. Bendall 894-4820

Sponsor technical contact

F. HANK GRANT
(202)357-5167

NATIONAL SCIENCE FOUNDATION
1800 G STREET, NW
WASHINGTON, DC 20550

Sponsor issuing office

DENISE YOUNG
(202)357-9602

NATIONAL SCIENCE FOUNDATION
DIVISION OF GRANTS AND CONTRACTS
1800 G STREET, NW
WASHINGTON, DC 20550

Security class (U,C,S,TS) : U
Defense priority rating : N/A
Equipment title vests with: Sponsor

ONR resident rep. is ACO (Y/N): N
NSF supplemental sheet
GIT X

Administrative comments -
ISSUED TO TRANSFER FUNDS IN THE AMOUNT OF $27.18 TO SUBPROJECT E-24-X28.

5R494

U

GEORGIA INSTITUTE OF TECHNOLOGY
OFFICE OF CONTRACT ADMINISTRATION

NOTICE OF PROJECT CLOSEOUT

Closeout Notice Date 07/12/96

Project No. E-27-651 Center No. 10/24-6-R6807-0A0

Project Director JAYARAMAN S School/Lab TEXT ENGR

Sponsor NATL SCIENCE FOUNDATION/GENERAL

Contract/Grant No. DDM-8957861 Contract Entity GTRC

Prime Contract No. ____

Title PYI - DESIGN AND DEVELOPMENT OF A MANUFACTURING ENTERPRISE ARCHITECTURE

Effective Completion Date 960331 (Performance) 960630 (Reports)

Closeout Actions Required:	Y/N	Date Submitted
Final Invoice or Copy of Final Invoice	N	
Final Report of Inventions and/or Subcontracts	N	
Government Property Inventory & Related Certificate	N	
Classified Material Certificate	N	
Release and Assignment	N	
Other ____	N	

Comments
LETTER OF CREDIT APPLIES. 98A SATISFIES PATENT REQUIREMENT.

Subproject Under Main Project No. ____

Continues Project No. ____

Distribution Required:

Project Director	Y
Administrative Network Representative	Y
GTRI Accounting/Grants and Contracts	Y
Procurement/Supply Services	Y
Research Property Managment	Y
Research Security Services	N
Reports Coordinator (OCA)	N
GTRC	Y
Project File	Y
Other ____	N
____	N

E-27-651
1

Design and Development of a Manufacturing Enterprise Architecture

Research Funded By

National Science Foundation
Division of Design and Manufacturing Systems
Washington, D.C. 20550

Under 1989 NSF-PYI Program

Award # DDM-8957861

Report Submitted By

Dr. Sundaresan Jayaraman
Georgia Institute of Technology
School of Textile & Fiber Engineering
Atlanta, Georgia 30332

404/894-2490

May 1990

SJ-PYI-AR1

Annual Progress Report -- Year 1

Design and Development of a Manufacturing Enterprise Architecture

Proposed Research Plan

Introduction: To be successful, competitive, and achieve excellence in today's global economy, a manufacturing enterprise must lead in successful use of the most advanced concepts and methods including Computer-Integrated Manufacturing (CIM). The scope of CIM transcends the traditional boundaries of the factory floor and encompasses the whole enterprise, giving rise to a Computer-Integrated Manufacturing Enterprise (CIME). CIME can be defined as one that utilizes computers for the engineering, planning, manufacturing, marketing and business functions of the enterprise, and for the integration of all these functions into a cohesive enterprise system through a common information/knowledge base.

The Manufacturing Enterprise Architecture (MEA): An important prerequisite for the successful implementation of CIM in a manufacturing enterprise is a detailed knowledge and understanding of the functions and information associated with the enterprise. Such a definition of the manufacturing enterprise is known as the *architecture* of manufacturing. A standard architecture would reduce the overall system complexity and enable users to build systems in increments. MEA is the framework that captures, represents and integrates the three major facets of an enterprise, *viz.*, function, information and dynamics.

Research Plan: The overall objective of this research effort is to design and develop MEA which will serve as a blueprint for the implementation of CIM in the factory of the future. There is a need for such a fundamental approach to the domain of manufacturing, especially in the context of CIM. During the initial phase of the research, the apparel/textile manufacturing sector will be used as the domain for the development of the Textile Apparel Manufacturing Architecture (TAMA). Methodologies currently available for the development of such an architecture, do not lend themselves well to the representation of experience-based knowledge. This opens up avenues for some interesting investigation since symbolic knowledge plays a crucial role in the successful operation of an enterprise (*e.g.*, in production planning and scheduling). TAMA will subsequently be generalized to encompass other manufacturing processes. Issues related to a domain- or industry-independent view of manufacturing will also be investigated. Thus, the five-year research objective is not only to have a successful implementation of CIM in textile/apparel manufacturing, but also to have a generic MEA for the factory of the future.

Summary of Current Activities

Development of Textile/Apparel Manufacturing Architecture: Work is being carried out on the development of Apparel Manufacturing Architecture (AMA), the initial domain for TAMA and eventually MEA. The IDEF methodology developed under the US Air Force's ICAM Program [1] is being used in the development of AMA. The three models -- Function, Information and Dynamics -- capture and represent the operations of an apparel enterprise. AMA will serve as a blueprint for the implementation of CIM in the apparel industry [2].

Domain-Independent Architecture: An important outcome of the research on AMA has been the concept of a domain-independent architecture for manufacturing. Basic concepts and preliminary ideas for further research in this area were presented at the IJCAI '89 Workshop on Manufacturing [3]. A copy of the paper is attached.

Improving the Modeling Methodology: One of the major drawbacks of the IDEF methodology is the lack of seamless integration between the function, information and dynamics models; consequently, efforts are being directed at laying the groundwork for a comprehensive methodology that will be suitable for developing MEA.

Hewlett-Packard Equipment: A major equipment and software grant valued at $437,630 has been received from Hewlett-Packard Company for use in the research activities. The hardware (HP 3000 925LX) has been installed and the Manufacturing Management Software is currently being loaded on to the System. This system will serve as the platform for implementing and testing the architectural concepts emerging during the course of the research.

References

[1] US Air Force Integrated Computer Aided Manufacturing Program Manuals, WPAFB, Ohio, 1981.

[2] Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part I: Basic Concepts and Methodology Selection", Textile Research Journal, May 1990.

[3] Jayaraman, S., "On a Manufacturing Enterprise Architecture", IJCAI '89 (International Joint Conference on Artificial Intelligence) Workshop on Manufacturing, Detroit, MI, August 21-25, 1989.

Paper Presented at The International Joint Conference on Artificial Intelligence (IJCAI '89) Workshop on Manufacturing, Detroit, MI, August 21-25, 1989.

On a Manufacturing Enterprise Architecture[1]

Dr. Sundaresan Jayaraman
School of Textile Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

* * *

For the Panel on <u>Integrated Manufacturing Architecture: What does it mean?</u>

* * *

Abstract

The development of a Manufacturing Enterprise Architecture (MEA) is a prerequisite for the successful implementation of Computer-Integrated Manufacturing (CIM) in an enterprise. A definition of MEA is proposed and its scope outlined. A domain- or industry-independent view of manufacturing is discussed in light of the product continuum that ranges from commodity-type items to specialized items. The role of knowledge and experience in operating an enterprise are examined. Several key issues germane to research in the area of integrated architectures for manufacturing are presented. These issues will be discussed during the workshop.

* * *

Introduction: In this rapidly evolving and highly competitive global economy, the term "manufacturing" stands redefined. No longer restricted to the actual production process (*e.g.*, milling, turning, cutting), it encompasses the design, development, production, distribution and marketing of the *right* product at the *right* price and at the *right* time (the three *R*s of a manufacturing enterprise). To be successful, competitive, and achieve excellence, a manufacturing enterprise must lead in successful use of the most advanced concepts and methods including Computer-Integrated Manufacturing (CIM). The scope of CIM transcends the traditional boundaries of the factory floor and encompasses the whole enterprise, giving rise to a Computer-Integrated Manufacturing Enterprise (CIME). CIME can be defined as one that utilizes computers for the engineering, planning, manufacturing, marketing and business functions of the enterprise, and for the integration of all these functions into a cohesive enterprise system through a common information/knowledge base.

The Manufacturing Enterprise Architecture (MEA): An important prerequisite for the successful implementation of CIM in a manufacturing enterprise is a detailed knowledge

---

[1]This research is being supported by a Presidential Young Investigator Award from the National Science Foundation, Washington, D.C.

and understanding of the functions and information associated with the enterprise [1]. Such a definition of the manufacturing enterprise is known as the *architecture* of manufacturing.

MEA is the framework that captures, represents and integrates the three major facets of an enterprise, *viz.*, function, information and dynamics. As shown in Figure 1, a functional representation of the enterprise is the foundation of the MEA. Information is the lifeline for the various functions and is the next level in the MEA. Though function and information are shown separately in the figure, they are closely interlinked and together form the foundation of the framework. Just as a solid engineering structure can be constructed only after the foundation has been laid, a thorough analysis of the dynamics of the enterprise (including *what-if* analysis and system simulation) can be performed effectively only after the function/information foundation has been laid. This function-information-dynamics representation of the existing operations of an enterprise is known as the *AS IS* architecture. It serves as a starting point for assessing, developing and implementing a *TO BE* architecture. A *TO BE* architecture is a representation of what the enterprise is "to be" in order to realize the three Rs.

Analysis of the Manufacturing Sector: The manufacturing sector can be viewed from different perspectives. The most common, albeit a narrow one, is based on the products produced, *viz.*, automobiles, aircraft, textiles, chemicals, machine tools, *etc.* A second view represents the segments served: transportation, food, clothing and machinery. Another commonly adopted classification is in terms of discrete parts and continuous manufacturing; examples of discrete parts manufacturing include garments, pens and automobiles, while sulfuric acid and paper manufacturing are examples of continuous processes. All these views emphasize the unique characteristics of the individual products and industry segments. However, they fail to highlight the infrastructure common to all of manufacturing (regardless of the product or industry segment), the infrastructure that is essential for attaining the three Rs.

Moreover, while the problems and issues faced by the different industries may differ in magnitude, the challenges themselves are similar: responding efficiently to the rapidly changing demands of consumers by producing and offering high quality products in the shortest possible time at competitive prices. For example, in the automotive sector, the models typically undergo major changes once every four years (with minor upgrades every year), while in the apparel sector, there are garments for each of the four seasons in a year. Thus, the apparel sector is characterized by a *higher* frequency of changes in the various functions (engineering and aesthetic design, manufacturing planning, scheduling, production, distribution, marketing, *etc.*) when compared to the automotive sector. However, the set of functions defining the enterprise and its operation is essentially the same. Concepts of "quick response", "zero defects" and "just in time" manufacturing are applicable to both sectors. This leads to another, and probably more useful, view of manufacturing.

A Domain-independent Perspective of Manufacturing: In a global sense, manufacturing is responsible for supplying a continuum of products ranging from commodity-type items (*e.g.*, socks, pens, television sets, etc.) to highly specialized or unique items (*e.g.*, tailored suits, fighter aircraft, *etc.*). This is a domain- or industry-independent perspective because almost every industry segment produces such a continuum of products. For example, in the apparel sector, socks and high-fashion evening gowns represent the two ends of the spectrum, while personal computers and supercomputers are representative of the two extremes in the computer industry. While, at first glance, it might seem odd to think of any commonality between the apparel and computer manufacturing sectors, there indeed is an infrastructure or architecture shared by the two fields. Of course, the structures built over this foundation will vary to accommodate the specifics of each industry.

Analysis of the Product Continuum: How does this product continuum influence the operations of an enterprise? In other words, what are the specific parameters that determine the position of the product in the continuum and how do they change from one end to the other (see Figure 2).

Information: One of the key parameters that changes in the product continuum is the associated information. As shown in Figure 2, the amount of information per unit product increases as one moves from a commodity-type product to a special product. Since a commodity-type product (*e.g.*, socks or ball-point pens) is a fairly stable or standardized product, the associated information (design, production rate, demand, *etc.*) also tends to be stable. However, at the other end, the specialized items themselves change rapidly and with them the information, or some significant fraction of the information. So, the rate of change of information is higher for the special end of the spectrum. Likewise, the type of information will also vary from one end to the other.

Extent and Type of Automation: Since commodity-type items will typically require fewer design changes and are produced in large quantities, the process lends itself well to automation. Moreover, it is easier to justify the large capital expenditures associated with this type of automation, commonly referred to as *fixed* automation. On the other hand, specialized products will require frequent changes, are produced in fewer quantities and typically at lower production rates, that *fixed* automation may not only be uneconomical but also impractical. The specialized items call for production equipment that need to be flexible, giving rise to *flexible* automation.

Product Demands and Quick Response: Commodity-type items have a fairly steady demand which can be forecast with a high degree of certainty. Consequently, production and process planning tasks are simpler and can be automated. In the case of specialized items, however, the demand cannot be forecast accurately and the enterprise must have the ability to respond quickly to the product's performance in the market. For example, in the apparel sector, if a new style of women's wear is selling well, the point-of-sale information must be utilized by the enterprise to gear up its production within days to meet the anticipated demand. So, the production and process planning tasks assume increased significance in the context of "quick response." Moreover, these tasks tend to be highly dynamic under such circumstances. Likewise, the "just in time" manufacturing strategy varies according to the position of the product in the continuum.

The Role of Experience and Knowledge: The functions (and the related information) in a manufacturing enterprise can be broadly classified into two major categories--formal and informal. As the name implies, in the formal class, the functions are well-defined, structured and represented in the system. An example of this class is the tool path on the Numerical Control or NC machine (defined), the set of instructions for the operation and the format in which the instructions are transmitted to the machine. In the second category, the functions are not always well-structured and represented in the system. This knowledge, typically embodied in the human component of the enterprise, is acquired over time by learning from experience. An example of this is the knowledge associated with the production planning process based on product demand forecasts. Intelligence in an enterprise is distributed: from the microchip in the machine on the factory floor to the human expert working as a strategic planner. Both human and machine intelligence are central to the successful operation of the enterprise.

Need for MEA and its Role: The preceding discussion intends to make clear that there is a need for MEA so that the enterprise can deal with the wide range of issues and strategies encountered in the real world. The architecture, developed by adopting a systems approach to manufacturing, can serve as a blueprint for the effective implementation of new technologies, including computers, which are central to the successful operation of the enterprise. The architecture can serve as a communication vehicle in an enterprise both during the analysis of the enterprise operations and subsequently during the implementation of changes resulting from the analysis. It can be used to develop specifications and standards for the seamless integration of the various islands of automation in an enterprise.

Scope of MEA: The activities of a manufacturing enterprise can be broadly classified into three major categories: (1) Strategic or long-term decision-making; examples include capital investment and expansion decisions. (2) Tactical decision-making related to the day-to-day workings of the enterprise; examples include production and process planning. (3) Operational activities whereby the tactical decisions are implemented; examples include product assembly. Since the purpose of the architecture is to represent the various activities of the enterprise, its scope should include all the three categories. The architecture must accommodate the continuum of products. And finally, for it to be complete, the architecture should allow for the representation of experience-based knowledge associated with the human element in the enterprise.

Research in Progress: Under a DoD-sponsored research effort at Georgia Tech, the architecture for an apparel manufacturing enterprise is being developed with the active participation of a major apparel manufacturer. Based on a set of evaluation criteria developed for the selection of the modeling methodology, the US Air Force's IDEF methodology was selected [2]. The function modeling has been carried out and information modeling is in progress [3, 4].

Long-term Research Goals: One of the ultimate objectives of this research effort is to enhance and modify this apparel manufacturing architecture to make it a domain-independent architecture for the factory of the future. This endeavor is being supported by the National Science Foundation. A major shortcoming of the IDEF methodology is that it does not allow for the representation of experience-based knowledge. This opens up some exciting avenues for the incorporation of AI techniques in the existing methodology. So, another long-term objective of this research endeavor is to develop a suitable methodology (and a software tool) that will facilitate the development of the complete or "ideal" manufacturing enterprise architecture.

Discussion Issues: It should be clear by now that there are several key issues that need to be investigated in relation to the development of an integrated architecture for a manufacturing enterprise. These include:

* Is the proposed definition of MEA adequate?
  SJ: Yes, will be elaborated.

* Can the domain-independent view of manufacturing be used to develop an architecture for manufacturing?
  SJ: Yes, but needs to be explored further.

* Will a single architecture be adequate for the continuum of products?
  SJ: Probably Yes.

* Should the three viewpoints of an enterprise activities (strategic, tactical and operational) be integrated into a single architecture? If so, can they be?
  SJ: Yes; with some effort.

* Do the current methodologies accommodate experience-based knowledge? If not, how can this be accomplished?
  SJ: No; will be elaborated.

* Can the industry be convinced to take this "fundamental" (or long-term) approach to manufacturing and CIM?
  SJ: Yes, the architectural approach is gaining acceptance in the apparel industry.

* Once the architecture is developed, how should it be "validated?"
  SJ: Team of industry experts: positive feedback from the apparel industry.

These issues will be discussed in detail during the presentation at the workshop.

Acknowledgements: The U.S. Defense Logistics Agency (DLA) is sponsoring the development of the apparel manufacturing architecture under a research grant (DLA-900-87-D-0018). The author thanks Mr. Don O'Brien, Mr. Dan Gearing and Ms. Helen Kerlin, all of DLA, for making this endeavor possible and Mr. Rajeev Malhotra, graduate research assistant working towards his Ph.D., for his participation in the development of the architecture. The author acknowledges the receipt of a Presidential Young Investigator Award from the National Science Foundation (NSF) for pursuing the long-term objectives of this research endeavor. The views and ideas expressed herein are those of the author and do not constitute an endorsement either by DLA or NSF.

References Cited

[1] Jayaraman, S., Design and Development of a Generic Architecture for Apparel Manufacturing, Georgia Tech Research Proposal submitted to Defense Logistics Agency, Cameron Station, VA, April 1988.

[2] Jayaraman, S., Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Technical Report, Georgia Institute of Technology, Atlanta, Georgia, July 1988.

[3] Jayaraman, S., and Malhotra, R., Design and Development of a Generic Architecture for Apparel Manufacturing, Technical Report SJ-TR-ARCH-8904, Georgia Institute of Technology, Atlanta, Georgia, April 1989.

[4] Jayaraman, S., and Malhotra, R., Apparel Manufacturing Architecture: The Function Model, AMTC Quarterly, May 1989, pp. 1-9, Georgia Institute of Technology, Atlanta, Georgia.

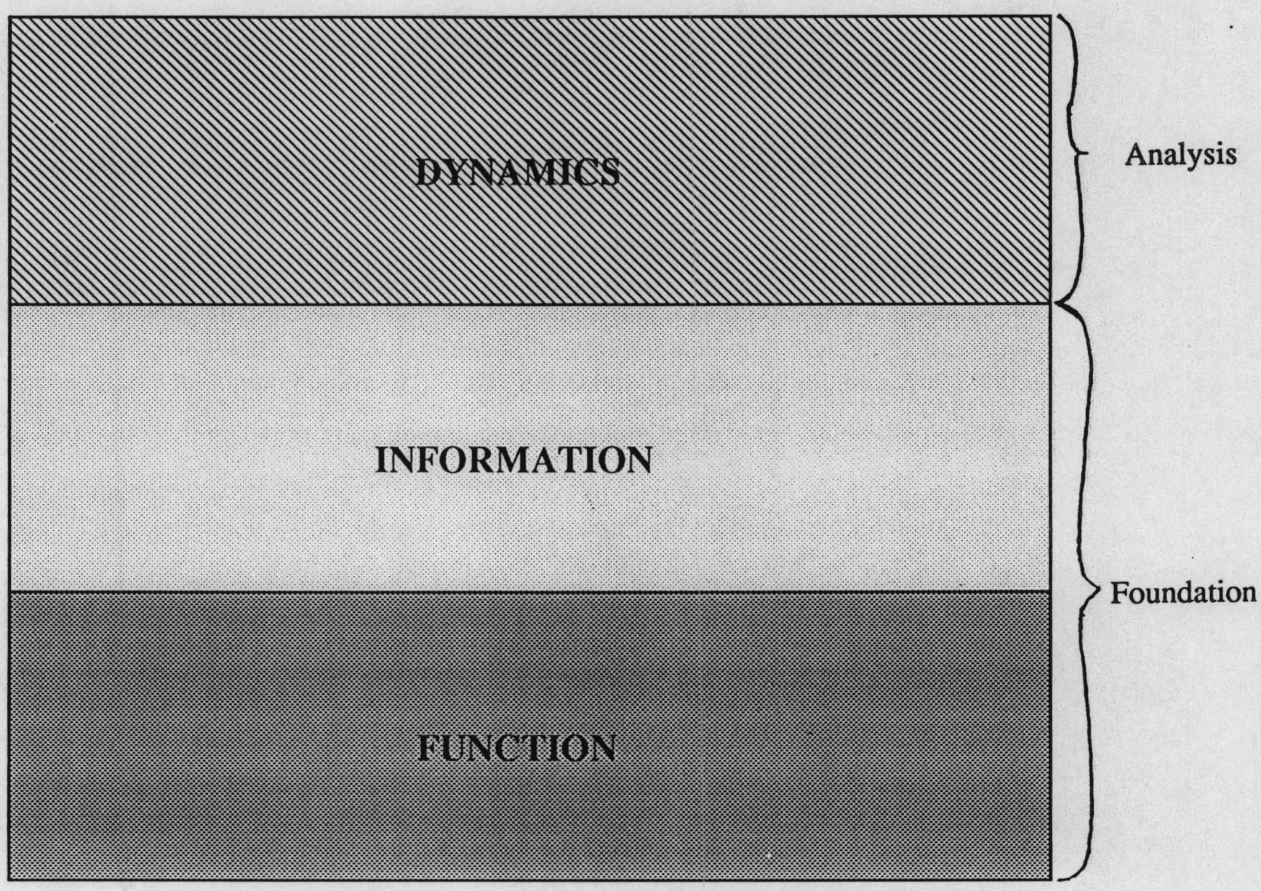


Figure 1. The Manufacturing Enterprise Architecture

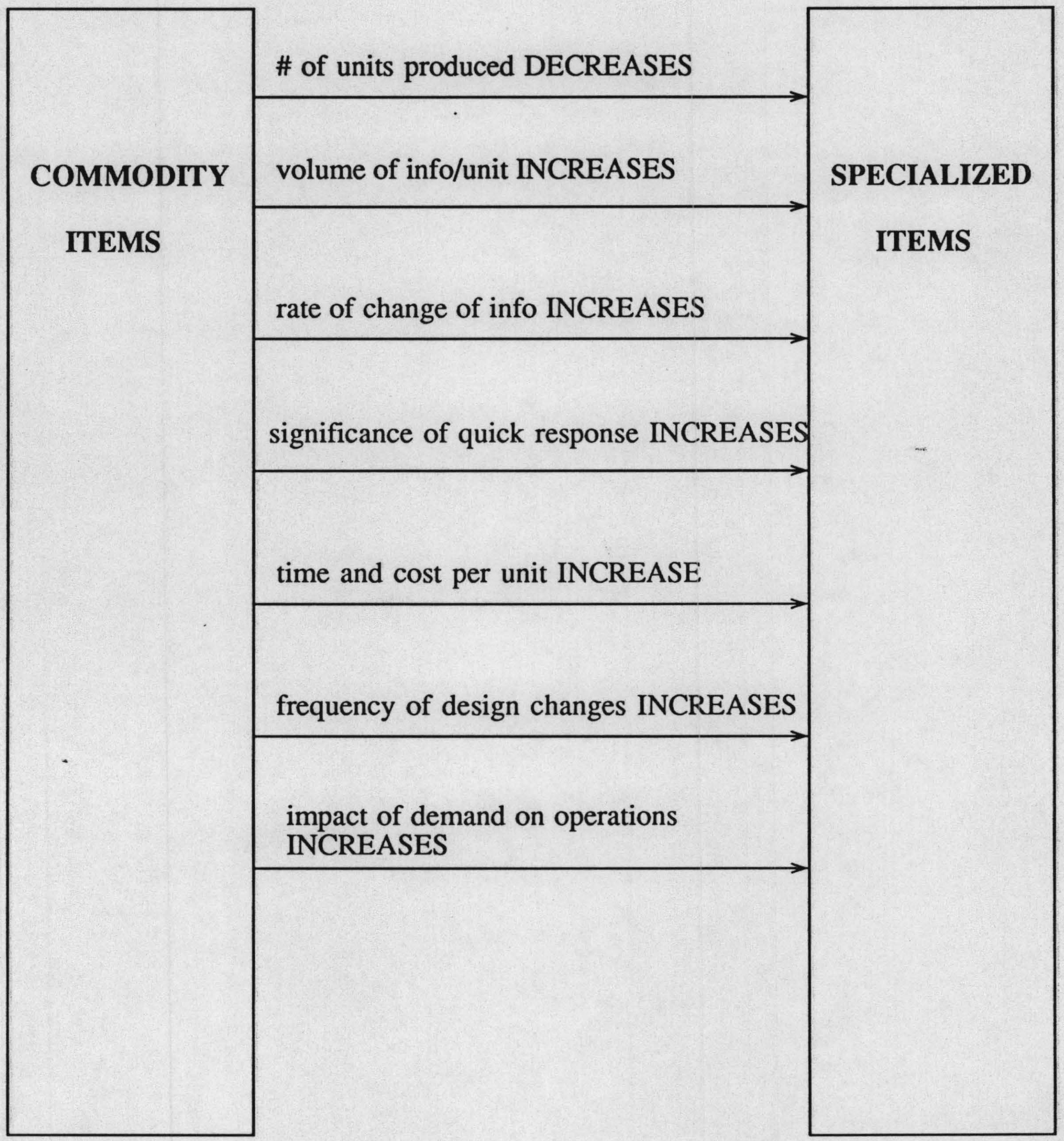


Figure 2. The Product Continuum

E-27-651
2

Design and Development of a Manufacturing Enterprise Architecture

Research Funded By

National Science Foundation
Division of Design and Manufacturing Systems
Washington, D.C. 20550

Under 1989 NSF-PYI Program

Award # DDM-8957861

Report Submitted By

Dr. Sundaresan Jayaraman
Georgia Institute of Technology
School of Textile & Fiber Engineering
Atlanta, Georgia 30332

404/894-2490

May 1991

SJ-PYI-AR2

Table of Contents

Annual Progress Report -- Year 2

Design and Development of a Manufacturing Enterprise Architecture

Proposed Research Plan

Introduction: To be successful, competitive, and achieve excellence in today's global economy, a manufacturing enterprise must lead in successful use of the most advanced concepts and methods including Computer-Integrated Manufacturing (CIM). The scope of CIM transcends the traditional boundaries of the factory floor and encompasses the whole enterprise, giving rise to a Computer-Integrated Manufacturing Enterprise (CIME). CIME can be defined as one that utilizes computers for the engineering, planning, manufacturing, marketing and business functions of the enterprise, and for the integration of all these functions into a cohesive enterprise system through a common information/knowledge base.

The Manufacturing Enterprise Architecture (MEA): An important prerequisite for the successful implementation of CIM in a manufacturing enterprise is a detailed knowledge and understanding of the functions and information associated with the enterprise. Such a definition of the manufacturing enterprise is known as the *architecture* of manufacturing. A standard architecture would reduce the overall system complexity and enable users to build systems in increments. MEA is the framework that captures, represents and integrates the three major facets of an enterprise, *viz.*, function, information and dynamics.

Research Plan: The overall objective of this research effort is to design and develop MEA which will serve as a blueprint for the implementation of CIM in the factory of the future. There is a need for such a fundamental approach to the domain of manufacturing, especially in the context of CIM. During the initial phase of the research, the apparel/textile manufacturing sector will be used as the domain for the development of the Textile Apparel Manufacturing Architecture (TAMA). Methodologies currently available for the development of such an architecture, do not lend themselves well to the representation of experience-based knowledge. This opens up avenues for some interesting investigation since symbolic knowledge plays a crucial role in the successful operation of an enterprise (*e.g.*, in production planning and scheduling). TAMA will subsequently be generalized to encompass other manufacturing processes. Issues related to a domain- or industry-independent view of manufacturing will also be investigated. Thus, the five-year research objective is not only to have a successful implementation of CIM in textile/apparel manufacturing, but also to have a generic MEA for the factory of the future.

Summary of Activities in Year Two of the Award

Development of Textile/Apparel Manufacturing Architecture: Work has been completed on the development of an apparel manufacturing architecture (AMA), the initial domain for TAMA and eventually MEA. The IDEF methodology developed under the US Air Force's ICAM Program [1] has been used for the development of AMA. The three models -- Function, Information and Dynamics -- capture and represent the operations of an apparel enterprise. AMA was developed in cooperation with major apparel manufacturers and it will serve as a blueprint for the implementation of CIM in the apparel industry [2, 3]. The work on AMA received additional support from the US DoD under grant # DLA-900-87-D-0018.

Modeling Methodology: During the course of this research, several shortcomings in the IDEF methodology have been identified and a new methodology termed IFEM (integrated framework for enterprise modeling) has been proposed. This work forms part of a doctoral dissertation completed in March 1991 at Georgia Tech by Rajeev Malhotra, one of Dr. Jayaraman's graduate students [5]. The proposed schema can serve as the foundation for the development of manufacturing system modeling software.

Domain-Independent Architecture: An important outcome of the research on AMA has been the concept of a domain-independent architecture for manufacturing [4]. Further work using object oriented programming (OOP) techniques has been initiated. A paper has recently been accepted for the Workshop on OOP in AI to be held during the 1991 American Association for Artificial Intelligence Conference in July [5]. The copy of the paper covering the new methodology and domain-independent enterprise architecture is attached.

Hewlett-Packard Equipment: The HP 3000 925LX and Manufacturing Management software obtained through an equipment grant from Hewlett-Packard is being used as the test bed for AMA. Work on implementing AMA using the information entities and the associated schema as the foundation has been initiated. Dr. Malhotra has spent time learning the system and will be involved in the implementation.

Education-related Activities: The graduate student who worked on the development of AMA (Rajeev Malhotra) recently received his PhD degree. He has since joined the team as a research scientist to assist in transforming some of the IFEM concepts to the development of software tools and also to implement AMA on the HP manufacturing management system. Another PhD student is working on the OOP aspects of MEA for his dissertation research. Yet another graduate student has been working on the knowledge-based systems aspect of MEA. In short, several graduate students are working on MEA and closely related topics.

The research activities are also finding their way into the classroom, both at the undergraduate and graduate levels. In the senior textile engineering design course, students are being introduced to hierarchical cell modeling techniques (e.g., IDEF), while at the graduate level, information processing aspects in textile science and engineering including CIM are being taught. In short, efforts are being directed both at education of engineers/educators for the future, and research in CIM.

Textbook Publication: While *not* supported by PYI funds, a textbook entitled Computer-Aided Problem Solving for Scientists and Engineers written by Dr. Jayaraman has recently been published by McGraw-Hill [7]. The text is aimed at freshman/sophomore science/engineering students taking the first course in computing. The PYI award certainly provided the necessary freedom and flexibility to pursue this activity that hopefully will be beneficial to science/engineering students, especially in computing. Copies of the front and back covers of the book are attached.

Equipment Grant from Sun Microsystems: Sun Microsystems has awarded an equipment grant worth $73,145 in support of Dr. Jayaraman's CIM research. A SPARCstation 470 has been procured and installed. The system is serving as the fileserver for the network of Sun workstations in the research laboratory. Several graduate students and research scientists in Dr. Jayaraman's group are making extensive use of the equipment. Again, the PYI Award played a major role in the award of the grant.

References

[1] US Air Force Integrated Computer Aided Manufacturing Program Manuals, WPAFB, Ohio, 1981.

[2] Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part I: Basic Concepts and Methodology Selection", Textile Research Journal, vol. 60, No. 5, pp. 248-254, 1990.

[3] Malhotra, R., and Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part II: The Function Model", Textile Research Journal, vol. 60, no. 6, pp. 351-359, 1990.

[4] Jayaraman, S., "On a Manufacturing Enterprise Architecture", IJCAI '89 (International Joint Conference on Artificial Intelligence) Workshop on Manufacturing, Detroit, MI, August 21-25, 1989.

[5] Srinivasan, K., and Jayaraman, S., "MEA: An Object-oriented Framework for Enterprise Modeling", Proceedings of the AAAI '91 Workshop on OOP in AI, Anaheim, CA, July, 1991, *to appear*.

[6] Malhotra, R., "An Architecture for an Apparel Manufacturing Enterprise", PhD Dissertation, Georgia Institute of Technology, Atlanta, Georgia, March 1991.

[7] Jayaraman, S., Computer-Aided Problem Solving for Scientists and Engineers, New York: McGraw-Hill, Inc., 1991.

Abstract of Current & Projected Activities

The overall objective of this research effort is to design and develop a Manufacturing Enterprise Architecture which will serve as a blueprint for the implementation of CIM in the factory of the future. The apparel manufacturing sector has been used as the initial domain and an apparel manufacturing architecture (AMA) has been developed. Based on this work, a new methodology termed IFEM (integrated framework for enterprise modeling) has been developed. Work on implementing AMA on the HP platform has been initiated.

Research on developing the proposed methodology (IFEM) further will be continued. Object oriented programming techniques will be utilized for developing MEA. The HP system will continue to serve as the implementation vehicle for testing out AMA and MEA concepts.

* * *

Budget -- Year 3

(Matching funds for all five years utilizing HP's equipment grant was approved by Dr. Louis Martin-Vega, NSF Program Director, in June 1990.)

Attachments

MEA: AN OBJECT-ORIENTED FRAMEWORK FOR MODELING A MANUFACTURING ENTERPRISE[1]

(Accepted for *Workshop on Object-Oriented Programming in AI*, AAAI-91, Anaheim, CA)

K. Srinivasan and Sundaresan Jayaraman[2]
Georgia Institute of Technology
School of Textile & Fiber Engineering
Atlanta, GA 30332-0295

ABSTRACT

Manufacturing Enterprise Architecture (MEA) is proposed as an object-oriented framework for modeling a manufacturing enterprise. MEA comprises three models encompassing the major facets of an enterprise, viz., function, information and dynamics. The three models are entity model, activity model and knowledge-beliefs model. These models are being designed to be independent of the manufacturing domain (textiles, automobiles, etc.) and viewpoint (manager, designer, etc.). The advantages of object-oriented programming in implementing MEA are discussed. Some yet-to-be-resolved representational problems in MEA are also presented.

INTRODUCTION

Computers are playing an increasing role in improving the productivity of a manufacturing enterprise. The three broad functions performed by computers in the field of manufacturing are:

1. Automation of manufacturing processes;
2. Assistance in human performance of management functions;
3. Integration of different functions through an underlying information framework.

Computers in general, and the field of Artificial Intelligence in particular, have contributed significantly to automation of manufacturing and related processes such as design and diagnosis. There are well known intelligent systems available for assisting in management functions such as scheduling [1]. However, it is essential to have a clear understanding of *all* the functions of a manufacturing enterprise and its problems so that these systems do not form islands of automation, but contribute effectively towards achieving the goals of the enterprise. The objective of Manufacturing Enterprise Architecture (MEA) is to provide a comprehensive model for promoting the understanding of an enterprise, and utilizing this model as an integrated framework for problem-solving and decision-making.

Domain And Viewpoint Independent Model

The area of manufacturing may be classified into different domains based on the product, viz., automobiles, computers, textiles, etc. The activities performed within an enterprise can be classified into three categories based on their viewpoint, viz., strategic activities (e.g., investment decisions), tactical activities (e.g., production planning) and operational activities (e.g., part inspection). Jayaraman has proposed a model of a manufacturing enterprise which is independent of the domain of the enterprise [2]. Such a model is desirable, as irrespective of the domain, any manufacturing enterprise has the same goals, such as high profitability and increased flexibility; the nature of functions performed is also essentially the same. The progress made so far in the development of such a model is discussed here.

---

1. This research is being funded in part by a US Department of Defense grant No. DLA 900-87-D-0018-0001 and by a National Science Foundation Presidential Young Investigator Research Award No. DDM-8957861.
2. To whom correspondence should be addressed.

STRUCTURE OF MEA

MEA consists of three models viz., the entity model, the activity model and a model to represent expert knowledge and beliefs about a manufacturing enterprise (Figure 1). The arrows between models stand for *refers to* relationship. Thus, the interdependence between the three models is apparent in the figure.

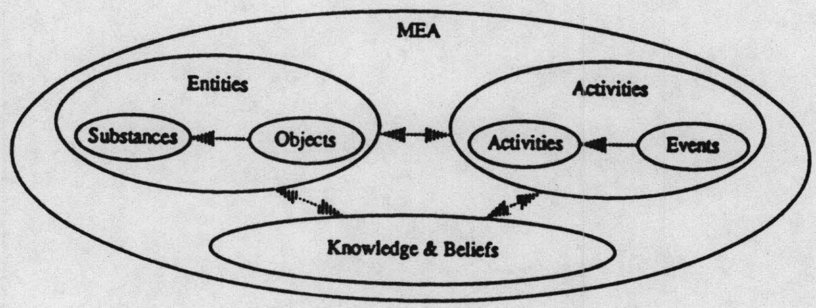


Figure 1. Structure of MEA

Entity Model

A part of the class hierarchy in the entity model is shown in Figure 2. The solid arrows represent *isA*[n] or inheritance relation between classes. For example, `Conceptual Entity` *isA*[n] `Entity` which implies all instances of `Conceptual Entity` will have all the attributes defined for the class `Entity`. The dotted arrows, as in Figure 1, represent *refers to* relationships.

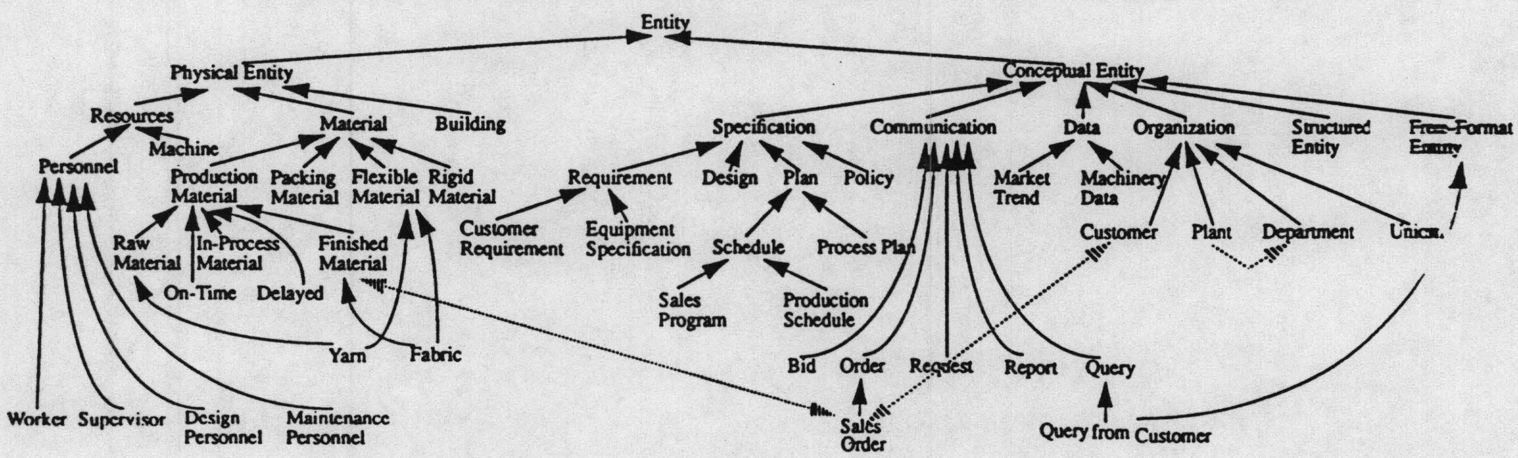


Figure 2. Part of MEA Entity Model

In MEA, the class `Entity` is defined as follows:

```
Class Entity
Slots
     Index         : <An alphanumeric string>
     Input to      : <A list of Activities>
     Control to    : <A list of Activities>
     Mechanism to  : <A list of Activities>
     Output from   : <A list of Activities>
```

Except index, which will have a unique value for each class, other slots can have lists of zero or more elements. Information such as the more general class of entities and list of attributes can be obtained from some form of Metaobject Protocol supported by Object-Oriented Programming Languages (OOPLs) [3].

Redundancy in Representation As can be seen in Figure 2, most of the *refers to* relationships between classes are two-way relationships. For example, the class definition for `Sales Order` will include a slot for the `Customer` who placed the order. Similarly, the `Customer` class will have a slot for storing a list of `Sales Orders` from a particular instance of `Customer`. Such redundant representation has been intentionally chosen for two reasons: First, the entity descriptions are semantically more complete and second, reasoning is made more efficient. However, with such redundant representation more care is required for consistency maintenance. OOPLs provide good support for such consistency checking. For example, in CLOS, *before, after* or *around* methods for *accessor* methods can ensure that when a slot value for one class is changed, corresponding changes are also made in slot values of related classes [3]. Assertions in Eiffel will also serve the purpose well [7].

Convergence in Entity Representation It is obvious that for MEA to be a realistic model of a manufacturing enterprise, a large number of entity classes is required. At first sight, it may appear that representing such a large number of entities and their interrelationship will make MEA too large to handle. However, the commonality among seemingly very disparate entities, coupled with the inheritance feature of OOPLs to capture such commonality, results in a convergence in the size of the entity model. For example, in Figure 2, `Policy` and `Equipment Specification` are seemingly two unconnected entities. However, both serve essentially the same purpose - specifying something. This commonality is implicitly shown by having both these entity classes as specializations of a general `Specification` class. The `Specification` class has a slot whose value is a list of *attribute-value* pairs. For example, an element in this list for class `Policy` is *"attribute*: targeted customer; *value*: middle-aged, urban, affluent male". Likewise, an element in the list for class `Equipment Specification` can be *"attribute*: level of automation; *value*: fully programmable". Such convergence in representation has been reported in literature [3].

Multiple Points of View - A Problem to be Overcome In an enterprise, the same entity may be viewed differently in different contexts. For example, `Yarn` is a `Raw Material` for the manager of a weaving plant. From a structural engineer's viewpoint, it is a `Flexible Material`. `Yarn` inherits slots such as `Vendor` and `Lead Time` from `Raw Material` class, and slots such as, `Tensile Strength` from `Flexible Material` class. With multiple inheritance supported in most OOPLs, synthesizing such complex objects from multiple viewpoints is quite simple. Problems arise when specific applications are interested in only a particular viewpoint of an object. For example, if a process planner is interested in a `Yarn` object with only some slots from each of its parents, current OOPL technology enforces the planner to "take the gorilla even if all she/he wants is the banana." No solution has been finalized for implementation in MEA at current time for this problem of defining different and possibly overlapping windows to objects for different applications.

Substance vs. Object How is the abstract concept `Cloth` related to a real piece of it, say, a beam of denim fabric woven on a particular loom and on a particular day? In MEA all entities, in general, are Substances, i.e., the class definitions include only slots to represent properties intrinsic to the entity. In our example, the class `Cloth` will have slots such as `Strength` and `Fiber Composition`. A `Beam Of Cloth` will have additional properties such as `Length` and `Loom Number`.

Activity Model

The activity model is a representation of the various functions performed in operating a manufacturing enterprise. Part of the class hierarchy in the activity model is shown in Figure 3. The solid arrows, as before, represent *isA*[n] or inheritance relationship, e.g., `Supervise Workers` *isA*[n] `Informal Activity`. The dotted arrows represent *is a sub-activity of* relationship, e.g., `Retrieve Parts` *is a sub-activity of* `Assemble Parts`. Activities such as `Operate a Manufacturing Enterprise` and `Manufacture Product` are compound activities whose sub-activities may belong to different classes, but by themselves cannot be assigned to any class except the most general `Activity` class. While the activity model is based on the $IDEF_0$ methodology [4], it enhances the expressiveness of the $IDEF_0$ methodology by the following: (1) Adding an inheritance hierarchy to capture the commonality among various activities; (2) Representing not only the static aspects of the activities, but also the dynamic aspects of it in the form of methods; and (3) Integrating with the entity model.

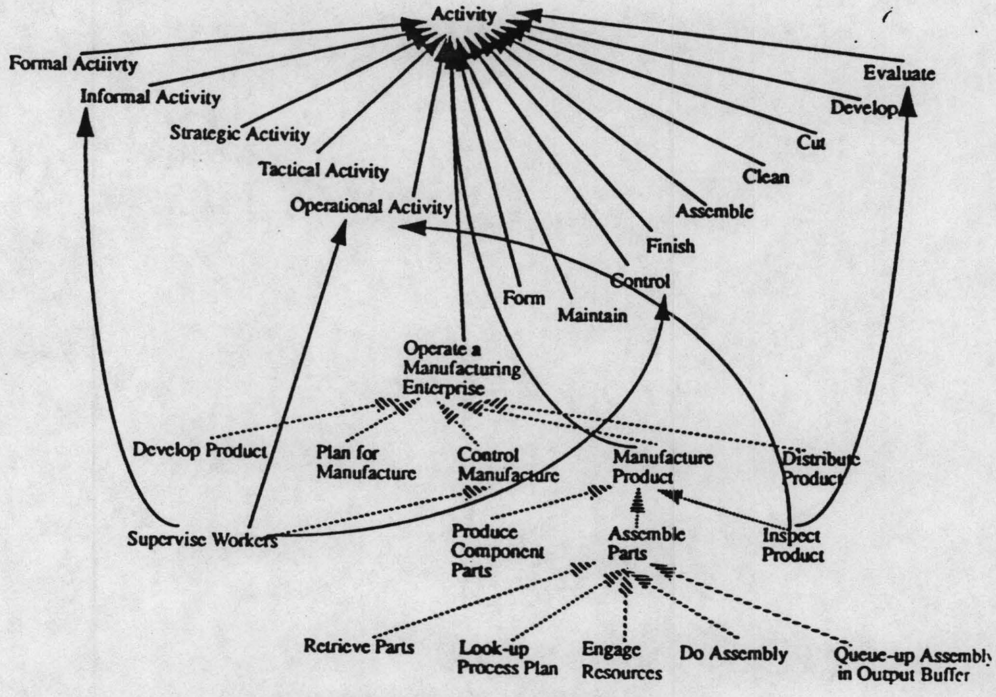
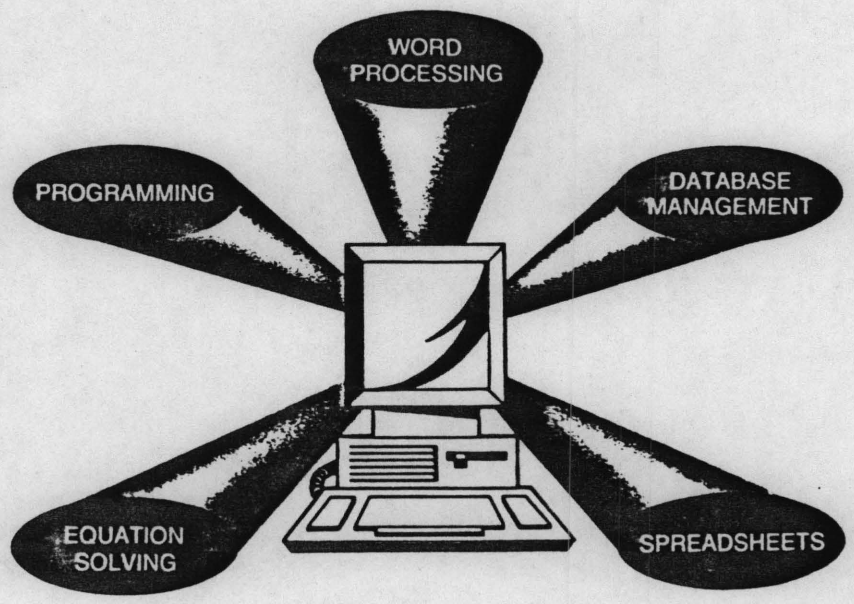


Figure 3. Part of MEA Activity Model

The class Activity is defined as follows in MEA:

```
Class Activity
Slots
     Index                : <An alphanumeric string>
     Input                : <A list of Entities>
     Control              : <A list of Entities>
     Mechanism            : <A list of Entities>
     Output               : <A list of Entities>
     Is Sub-Activity of   : <A list of Activities>
     Has Sub-Activities   : <A list of Activities>
```



Computer-Aided Problem Solving for Scientists and Engineers enables students and professionals in science and engineering to effectively harness the power of computer-aided problem solving and productivity tools. A few highlights:

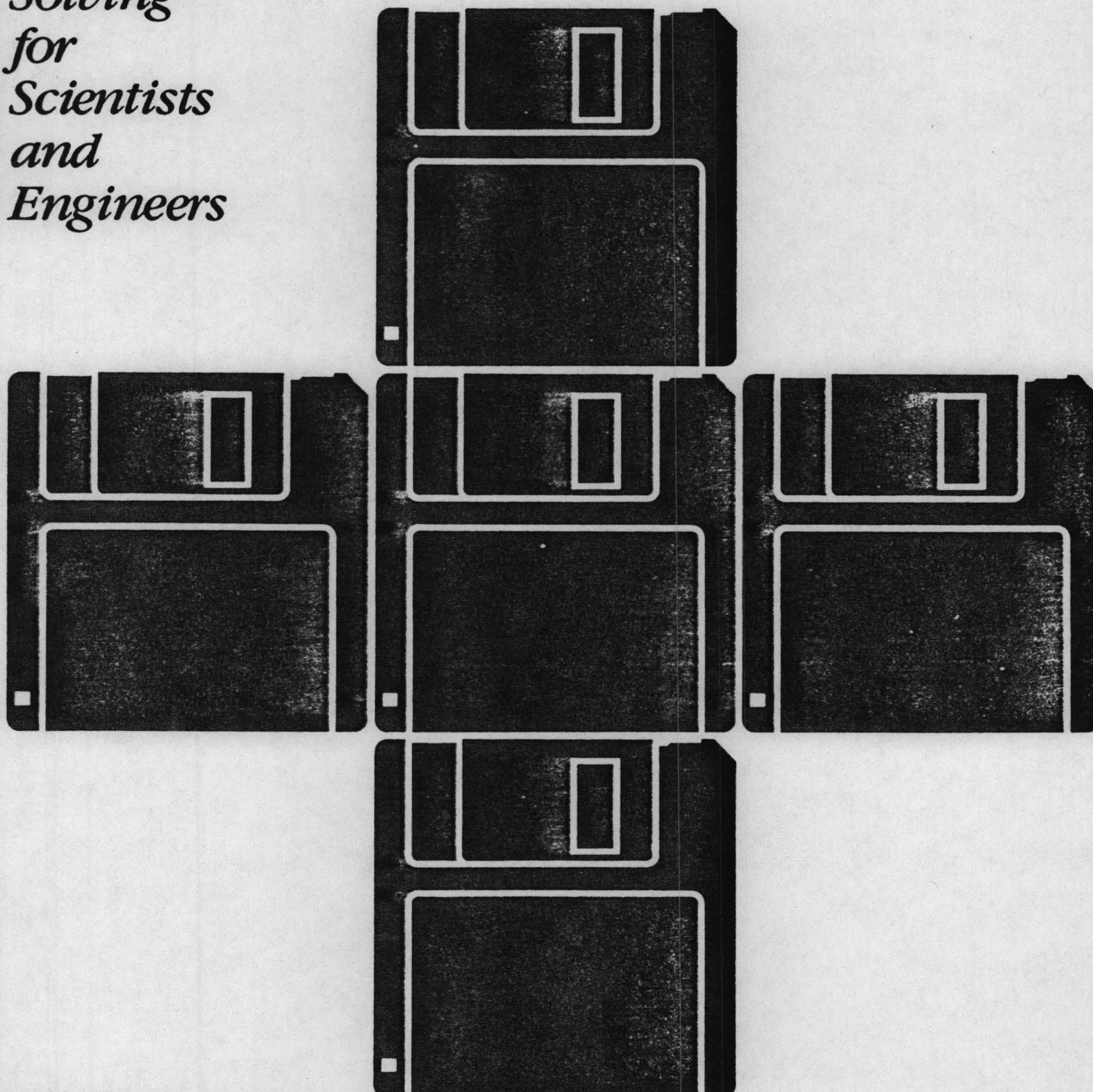
- Introduces the basics of computing hardware and software
- Presents a structured methodology for quantitative problem solving in science and engineering
- Illustrates the underlying concepts of problem solving and productivity tools through numerous science and engineering examples
- Provides generic methodologies for effectively using the following classes of programs:
  - Spreadsheets • Equation Solving • Database Management • Word Processing
- Organizes commonly used commands for major programs in quick reference tables
- Presents the fundamentals of computer programming, including equivalent BASIC and FORTRAN statements for various programming constructs

ISBN 0-07-032353-4

90000

9 780070 323537

Computer-Aided Problem Solving for Scientists and Engineers



Sundaresan Jayaraman

REFERENCES

1. Fox, M.S., Smith, S.F., ISIS: a Knowledge-Based System for Factory Scheduling, Expert Systems, vol. 1, no. 1, 1984.

2. Jayaraman, S., On a Manufacturing Enterprise Architecture, IJCAI 1989, Workshop on Integrated Architectures for Manufacturing, August 24, 1989.

3. Keene, S., Object-Oriented Programming in Common Lisp: A Programmer's Guide to CLOS, Addison-Wesley Publishing Company, Inc., December 1988.

4. Kiczales, G., Bobrow, D., Common Lisp Object System Specification: Chapter 3. Metaobject Protocol, Draft No. 11, July 30, 1990.

5. Lenat, D.B., Guha, R.V., Building Large Knowledge-Based Systems, Addison-Wesley Publishing Company, Inc., December, 1989.

6. Materials Laboratory, Air Force Wright Aeronautical Laboratories, Integrated Computer-Aided Manufacturing (ICAM) Function Modeling Manual ($IDEF_0$), June 1981.

7. Meyer, B., Object-Oriented Software Construction, Prentice Hall International, 1988.

8. Jayaraman, S., Source Selection in Apparel Procurement at DoD: A Knowledge-Based Approach, Technical Report No. SJ-TR-PROC-8812, Georgia Institute of Technology, Atlanta, Georgia, 30332, December 1988.

Primitive Activities as Methods In Figure 3, a few of the primitive operations which are sub-activities of `Assemble Parts` are shown. In MEA, these primitive operations are represented as methods. `Assemble Parts` will then have a method which will contain a collection of these methods. The ability to combine methods of classes at different levels in a hierarchy (e.g., the rich method-combination features of CLOS) make it possible to attach primitive activities common for all types of `Assemble Parts` activities (e.g., `Retrieve Parts`) to be included in the method for the `Assemble Parts` class. Primitive activities such as, `Set-Up Equipment` and `Do Assembly` are methods of more specific activity classes (not shown in Figure 3). The method corresponding to the primitive activity `Do Assembly` will depend on the use of the model. In a real-time simulation, it will just be a call to a "sleep" function. In an actual control system, it may be a robot control program.

Comparison with Entity Model The characteristics of the entity model discussed earlier also hold for the activity model. There is a convergence in representation of apparently disparate activities; similar to substance-object distinction is the activity-event distinction and the activity classes in MEA, except some primitive level activities, do not make any assumptions on the domain of the enterprise or the viewpoint of the model.

Knowledge-Beliefs Model

Under a U.S. Defense Logistics Agency research grant, we have developed an Apparel Enterprise Evaluation Framework (AEEF), to evaluate the manufacturing capabilities of an apparel enterprise [5]. This framework uses a class hierarchy to represent some of the entities and activities in an apparel manufacturing enterprise, and production rules to represent expert knowledge and beliefs about evaluating an enterprise. The class definitions also express certain beliefs such as *Quality Capability should be given 45% relative importance in deciding the Overall Capability of an enterprise* and *Machinery Features should be given 25% relative importance in deciding the Quality Capability.* This is done by having a `Weight` slot attached to all classes.

A typical production rule in AEEF is "*If* Raw Material is supplied by a vendor listed in Approved Supplier List, *Then* There is no need to inspect the Raw Material." Such rules can be incorporated easily into an object-oriented representation framework as assertions on methods and slots. The assertion language can be based on production rules or first order logic. We believe that a significant part of the knowledge-beliefs framework is domain-independent. However, using the framework for a different domain will require a full revision of parameters supporting the framework and this is a nontrivial knowledge acquisition task.

CONCLUSIONS

An object-oriented framework is being developed for modeling entities and activities involved in a manufacturing enterprise. Possible means for incorporating expert knowledge and beliefs into this framework have been proposed. Some yet-to-be resolved problems such as multiple viewpoints of the same objects have been highlighted for further discussions to identify plausible solutions.

E27-651
3

Design and Development of a Manufacturing Enterprise Architecture

Research Funded By

National Science Foundation
Division of Design and Manufacturing Systems
Washington, D.C. 20550

Under 1989 NSF-PYI Program

Award # DDM-8957861

Report Submitted By

Dr. Sundaresan Jayaraman
Georgia Institute of Technology
School of Textile & Fiber Engineering
Atlanta, Georgia 30332

404/894-2490

May 1992

SJ-PYI-AR3

Table of Contents

Annual Progress Report -- Year 3

Design and Development of a Manufacturing Enterprise Architecture

Proposed Research Plan

Introduction: To be successful, competitive, and achieve excellence in today's global economy, a manufacturing enterprise must lead in successful use of the most advanced concepts and methods including Computer-Integrated Manufacturing (CIM). The scope of CIM transcends the traditional boundaries of the factory floor and encompasses the whole enterprise, giving rise to a Computer-Integrated Manufacturing Enterprise (CIME). CIME can be defined as one that utilizes computers for the engineering, planning, manufacturing, marketing and business functions of the enterprise, and for the integration of all these functions into a cohesive enterprise system through a common information/knowledge base.

The Manufacturing Enterprise Architecture (MEA): An important prerequisite for the successful implementation of CIM in a manufacturing enterprise is a detailed knowledge and understanding of the functions and information associated with the enterprise. Such a definition of the manufacturing enterprise is known as the *architecture* of manufacturing. A standard architecture would reduce the overall system complexity and enable users to build systems in increments. MEA is the framework that captures, represents and integrates the three major facets of an enterprise, *viz.*, function, information and dynamics.

Research Plan: The overall objective of this research effort is to design and develop MEA which will serve as a blueprint for the implementation of CIM in the factory of the future. There is a need for such a fundamental approach to the domain of manufacturing, especially in the context of CIM. During the initial phase of the research, the apparel/textile manufacturing sector will be used as the domain for the development of the Textile Apparel Manufacturing Architecture (TAMA). Methodologies currently available for the development of such an architecture, do not lend themselves well to the representation of experience-based knowledge. This opens up avenues for some interesting investigation since symbolic knowledge plays a crucial role in the successful operation of an enterprise (*e.g.*, in production planning and scheduling). TAMA will subsequently be generalized to encompass other manufacturing processes. Issues related to a domain- or industry-independent view of manufacturing will also be investigated. Thus, the five-year research objective is not only to have a successful implementation of CIM in textile/apparel manufacturing, but also to have a generic MEA for the factory of the future.

Summary of Activities in Year Three of the Award

Enterprise Modeling Methodology: During the earlier phase of this research, several shortcomings in the IDEF methodology were identified and a new methodology termed IFEM (integrated framework for enterprise modeling) was proposed. Further work has since been carried out in this area. The *conceptual* schema proposed earlier [1, 2] for MEA has been implemented in software using CLOS, an object oriented programming (OOP) language. A paper based on this work has been accepted for publication in the proceedings of the Workshop on AI in Enterprise Integration to be held during the 1992 American Association for Artificial Intelligence Conference in July 1992 [3]. A copy of the paper is attached.

Utilization of Apparel Manufacturing Architecture: The Apparel Manufacturing Architecture developed earlier [4, 5] is being utilized in an apparel plant to model the planning and purchasing systems of the enterprise. The resulting function and information models have been used as the basis for setting up the prototype information system. The system is currently being evaluated at the plant. This work is also providing the necessary data for testing MEA discussed earlier. The work on AMA received additional support from the US DoD under grant # DLA-900-87-D-0018.

Hewlett-Packard Equipment: The HP 3000 925LX and Manufacturing Management software obtained through an equipment grant from Hewlett-Packard is being used as the test bed for AMA. Work on implementing AMA using the information entities and the associated schema as the foundation has been carried out. During the course of this effort, several limitations of the system were identified.

NATO Advanced Study Institute: Dr. Jayaraman was invited to give a lecture at the NATO Advanced Study Institute on Advancements and Applications of Mechatronics Design in Textile Engineering held in Side, Turkey, during April 3-16, 1992. A copy of Dr. Jayaraman's lecture entitled *Computer-Aided Design and Manufacturing: A Textile-Apparel Perspective* is attached [6]. At the ASI, Dr. Jayaraman had the opportunity to interact with fellow researchers from other NATO countries and explore areas for future collaboration in research.

Education-related Activities: Mr. Rajeev Malhotra, the graduate student who received his PhD degree working on the development of AMA, has recently joined United Parcel Service as an information systems design engineer. After his PhD, Dr. Malhotra continued as a research scientist in the group and worked on applying AMA and the IFEM methodology in a real-world apparel plant. Since Dr. Malhotra will be utilizing many of the concepts and knowledge gained during his study at Georgia Tech, the PYI program has contributed to the development of human resources for the industry and the dissemination of technical expertise through such individuals.

Mr. K. Srinivasan is a PhD student currently working on the development of MEA for his dissertation. He is expected to defend his PhD dissertation proposal in summer. Another student is using simulation techniques to study the relative merits of different methods of work flow on the apparel plant floor, e.g., bundle and modular cells. Another graduate student is examining the role of various structural interlacement schemes in manufacturing light-weight composites with better interlaminar mechanical properties. This work is being supported by NASA Langley. Yet another graduate student has been working on the knowledge-based systems aspect of MEA. In short, several graduate students are working on MEA and closely related topics.

As part of the Georgia Tech contingent, Dr. Jayaraman participated in the TQM University Challenge hosted in May 1992 by Milliken & Company in Spartanburg, South Carolina. It was gratifying to find that many of the research topics being pursued in Dr. Jayaraman's group are indeed finding their way into the textile plant (e.g., quick response, design for manufacturability, systems analysis).

In addition, the research activities are also finding their way into the classroom: In the senior textile engineering design course, students are being introduced to the concepts of design for quality, manufacturability and life-cycle, quick response, CIM and hierarchical cell modeling techniques (e.g., IDEF). In short, efforts are being directed both at education of engineers/educators for the future, and research in CIM.

Publication of Instructor's Manual: While *not* supported by PYI funds, an Instructor's Manual for Dr. Jayaraman's textbook entitled Computer-Aided Problem Solving for Scientists and Engineers [7] was published last Fall by McGraw-Hill [8]. The text itself is aimed at freshman/sophomore science/engineering students taking the first course in computing. The PYI award certainly provided the necessary freedom and flexibility to pursue this activity. Copies of the front and back covers of the Manual are attached.

Use of Systems from Sun Microsystems: The equipment received under a grant from Sun Microsystems in support of Dr. Jayaraman's CIM research is being used extensively in the research lab by students and research scientists.

References

[1] Jayaraman, S., "On a Manufacturing Enterprise Architecture", IJCAI '89 (International Joint Conference on Artificial Intelligence) Workshop on Manufacturing, Detroit, MI, August 21-25, 1989.

[2] Srinivasan, K., and Jayaraman, S., "MEA: An Object-oriented Framework for Enterprise Modeling", Proceedings of the AAAI '91 Workshop on OOP in AI, Anaheim, CA, July, 1991.

[3] Srinivasan, K., and Jayaraman, S., "Design and Development of an Enterprise Modeling Methodology", Proceedings of the AAAI '92 Workshop on AI in Enterprise Integration, San Diego, CA, July, 1992, *to appear*.

[4] Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part I: Basic Concepts and Methodology Selection", Textile Research Journal, vol. 60, No. 5, pp. 248-254, 1990.

[5] Malhotra, R., and Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part II: The Function Model", Textile Research Journal, vol. 60, no. 6, pp. 351-359, 1990.

[6] Jayaraman, S., "Computer-Aided Design and Manufacturing: A Textile-Apparel Perspective", Invited Lecture Delivered at the NATO Advanced Study Institute on Advancements and Applications of Mechatronics Design in Textile Engineering, Side, Turkey, April 3-16, 1992.

[7] Jayaraman, S., Computer-Aided Problem Solving for Scientists and Engineers, New York: McGraw-Hill, Inc., 1991.

[8] Jayaraman, S., "Instructor's Manual to Accompany Computer-Aided Problem Solving for Scientists and Engineers", New York: McGraw-Hill, Inc., 1991.

Abstract of Current & Projected Activities

The overall objective of this research effort is to design and develop a Manufacturing Enterprise Architecture which will serve as a blueprint for the implementation of CIM in the factory of the future. The apparel manufacturing sector has been used as the initial domain and an apparel manufacturing architecture (AMA) has been developed. Based on this work, a new methodology termed IFEM (integrated framework for enterprise modeling) has been developed. These concepts are being transformed into software. Work is also in progress on modeling an apparel plant and using this data to test and evaluate the MEA framework.

Research on developing the proposed methodology (IFEM) further will be continued. Implementation work utilizing object oriented programming techniques to develop MEA will be continued. Since investment in technology should precede the implementation of advanced technologies, an additional (and new) area will be examined, viz., the roles of financial and managerial considerations in implementing advanced technologies in a manufacturing enterprise. Faculty members from the School of Management will be participating in this phase of the research. The HP system will continue to serve as the implementation vehicle for testing out AMA and MEA concepts.

* * *

Design and Development of a Manufacturing Enterprise Architecture

Research Funded By

National Science Foundation
Division of Design and Manufacturing Systems
Washington, D.C. 20550

Under 1989 NSF-PYI Program

Award # DDM-8957861

Report Submitted By

Dr. Sundaresan Jayaraman
Georgia Institute of Technology
School of Textile & Fiber Engineering
Atlanta, Georgia 30332

404/894-2490

April 1993

SJ-PYI-AR4

Table of Contents

I. Annual Progress Report -- Year 4

Design and Development of a Manufacturing Enterprise Architecture

1. Overall Proposed Research Plan

Introduction: To be successful, competitive, and achieve excellence in today's global economy, a manufacturing enterprise must lead in successful use of the most advanced concepts and methods including Computer-Integrated Manufacturing (CIM). The scope of CIM transcends the traditional boundaries of the factory floor and encompasses the whole enterprise, giving rise to a Computer-Integrated Manufacturing Enterprise (CIME). CIME can be defined as one that utilizes computers for the engineering, planning, manufacturing, marketing and business functions of the enterprise, and for the integration of all these functions into a cohesive enterprise system through a common information/knowledge base.

The Manufacturing Enterprise Architecture (MEA): An important prerequisite for the successful implementation of CIM in a manufacturing enterprise is a detailed knowledge and understanding of the functions and information associated with the enterprise. Such a definition of the manufacturing enterprise is known as the *architecture* of manufacturing. A standard architecture would reduce the overall system complexity and enable users to build systems in increments. MEA is the framework that captures, represents and integrates the three major facets of an enterprise, *viz.*, function, information and dynamics.

Research Plan: The overall objective of this research effort is to design and develop MEA which will serve as a blueprint for the implementation of CIM in the factory of the future. There is a need for such a fundamental approach to the domain of manufacturing, especially in the context of CIM. During the initial phase of the research, the apparel/textile manufacturing sector will be used as the domain for the development of the Textile Apparel Manufacturing Architecture (TAMA). Methodologies currently available for the development of such an architecture, do not lend themselves well to the representation of experience-based knowledge. This opens up avenues for some interesting investigation since symbolic knowledge plays a crucial role in the successful operation of an enterprise (*e.g.*, in production planning and scheduling). TAMA will subsequently be generalized to encompass other manufacturing processes. Issues related to a domain- or industry-independent view of manufacturing will also be investigated. Thus, the five-year research objective is not only to have a successful implementation of CIM in textile/apparel manufacturing, but also to have a generic MEA for the factory of the future.

2. Summary of Activities in Year Four of the Award

Enterprise Modeling Methodology: During the earlier phase of this research, several shortcomings in the IDEF methodology were identified and a new methodology termed IFEM (integrated framework for enterprise modeling) was proposed. Further work has since been carried out in this area. The *conceptual* schema proposed earlier [1, 2, 3] for MEA has been implemented in software using CLOS, an object oriented programming (OOP) language [4]. The framework for *activity* and *entity* modeling has been completed and work is currently in progress to expand the framework to encompass *dynamics* modeling. An algorithm for establishing precedence relationships in the activity model has been developed and is being implemented. A paper on this topic has been accepted for presentation at the upcoming AAAI '93 Workshop on *Modeling at Large* [5].

Textile Manufacturing Architecture: Work is in progress to expand the scope of the Apparel Manufacturing Architecture (AMA) developed early during the research program [6, 7] to encompass the fiber, yarn and fabric manufacturing segments of the textile-apparel complex. The work on AMA received additional support from the US DoD under grant # DLA-900-87-D-0018. Research has shown that AMA is generic enough at higher levels (e.g., product development, production planning, and distribution) to represent other sectors of manufacturing besides apparel. At the lower levels, however, information specific to the domain needs to be modeled. To further confirm this hypothesis, the current research focuses on modeling the fiber, yarn and fabric facets of manufacturing. These models are being integrated with AMA to accomplish one of the research objectives of developing a comprehensive TAMA.

Utilization of Apparel Manufacturing Architecture: Additional efforts have also been carried out to utilize AMA in yet another apparel plant (in addition to the one reported last year). The research involved developing a model for the *dual use* of the production capacity for military and commercial purposes. In a non-mobilization mode, the plant supplies the military and a major commercial customer, McDonald's. However, in the event of a mobilization, the production capacity normally utilized for McDonald's will be diverted to supply the military. Thus, the MEA research effort has involved industry partners to test out the research results and this arrangement has been mutually beneficial: industry feedback has been useful to enhance the quality of the research while the industry's operations have benefitted from the utilization of the latest research findings. The developed model is also being used to test MEA.

Justification of Investments in Information Systems and Technologies: Information is the lifeblood of an enterprise, especially when a manufacturing enterprise needs to rapidly reconfigure itself -- change designs, materials, styles, production techniques, etc. -- in response to consumer demands and market trends [8]. The ability to successfully harness this valuable resource in a timely and well-coordinated fashion calls for investments in information systems and technologies. The life-cycle view of an information system/technology (IS/IT) project has been used to explore issues related to the development of a methodology for justifying investments in information systems and technology [9]. As part of this effort, discussions have been held with two major categories of organizations: those that use information systems (either developed in-house or developed by outside consultants), and those who develop information systems for clients. Preliminary findings indicate that there is no specific or well-defined methodology used by organizations to justify investments in IS/IT.

Hewlett-Packard Equipment: The HP 3000 925LX and Manufacturing Management software obtained through an equipment grant from Hewlett-Packard is being used as the test bed for AMA. Work on implementing AMA using the information entities and the associated schema as the foundation has been carried out. During the course of this effort, several limitations of the system were identified.

Use of Systems from Sun Microsystems: The equipment received under a grant from Sun Microsystems in support of Dr. Jayaraman's CIM research is being used extensively in the research lab by students and research scientists.

AMEF EI Focus Group: Dr. Jayaraman was invited to join the Enterprise Integration Group of AMEF (Agile Manufacturing Enterprise Forum) at the Iacocca Institute (Lehigh University). As part of this group, he will be involved in developing a blueprint for enterprise integration in the context of an agile manufacturing enterprise. The work on MEA and the related real-world applications in textile/apparel plants are expected to facilitate this activity.

Problems, Favorable or Unusual Developments: No problems were encountered during the course of the research effort. On the other hand, the enthusiastic response of the manufacturing sector to the research results has had a positive influence on the work. Also, the invitation to serve on the AMEF EI Focus Group is another recognition of the quality of the work being pursued under this research effort. Finally, the support of HP and Sun Microsystems during the course of the PYI award has been extremely encouraging.

Significance of the Research Activities: The significance of the research work carried out can be assessed in terms of its contributions to the field of information systems in manufacturing, enterprise modeling methodologies and the real world of textile/apparel manufacturing. The contributions to education and human resources development are discussed in the next section. The manufacturing enterprise architecture is a reference model for building manufacturing information systems that is independent of the domain at the higher levels. IFEM represents a novel approach to enterprise modeling methodologies and helps develop an integrated view of a manufacturing enterprise. Since the research results are being tested in the real world, the work is contributing to enhancing the competitiveness of the manufacturing (specifically, textile/apparel) industry. Also, the refereed publications, refereed workshop presentations (at AAAI) and real world case studies are indicators of the significance of the research work being carried out.

3. Education & Human Resources Development Activities

Mr. K. Srinivasan is a PhD student currently in the final stages of his dissertation on the development of MEA. He successfully defended his PhD dissertation proposal last summer. Another graduate student, Ms. Yin Zhou, successfully completed her M.S. thesis examining the role of simulation in studying the relative merits of different methods of work flow on the apparel plant floor, e.g., bundle and modular cells. She is currently employed as an engineer in an engineering consulting firm in Atlanta, Georgia. Another graduate student, Ms. Jill Davis, also successfully defended her M.S. thesis in which she studied the effects of various structural interlacement schemes on the mechanical properties of light-weight composites. This work has also been supported by NASA Langley. Another graduate student, Mr. Badri Narasimhan, has recently begun his M.S. thesis research and is developing the textile manufacturing architecture discussed earlier. Dr. Rajeev Malhotra, who earned his Ph.D. with Dr. Jayaraman while working on this project (during the early years) is presently an information systems designer at UPS (United Parcel Service). In short, several graduate students have worked, and are working, on MEA and closely related topics.

In addition, as in the past years, the research activities are also finding their way into the classroom: In the senior textile engineering design course, students are being introduced to the concepts of design for quality, manufacturability and life-cycle, quick response, CIM and hierarchical cell modeling techniques (e.g., IDEF). In short, efforts are being directed both at education of engineers/educators for the future, and research in CIM.

References

[1] Malhotra, R., and Jayaraman, S., "An Integrated Framework for Enterprise Modeling", Journal of Manufacturing Systems, vol. 11, no. 6, pp. 426-441, 1992.

[2] Jayaraman, S., "On a Manufacturing Enterprise Architecture", IJCAI '89 (International Joint Conference on Artificial Intelligence) Workshop on Manufacturing, Detroit, MI, August 21-25, 1989.

[3] Srinivasan, K., and Jayaraman, S., "MEA: An Object-oriented Framework for Enterprise Modeling", Proceedings of the AAAI '91 Workshop on OOP in AI, Anaheim, CA, July, 1991.

[4] Srinivasan, K., and Jayaraman, S., "Design and Development of an Enterprise Modeling Methodology", Proceedings of the AAAI '92 Workshop on AI in Enterprise Integration, San Diego, CA, July, 1992.

[5] Srinivasan, K., and Jayaraman, S., "On Precedence Relationships in an Aggregation Hierarchy of Activities", Proceedings of the AAAI '93 Workshop on Modeling at Large, Washington, D.C., July, 1993, *to appear*.

[6] Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part I: Basic Concepts and Methodology Selection", Textile Research Journal, vol. 60, No. 5, pp. 248-254, 1990.

[7] Malhotra, R., and Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part II: The Function Model", Textile Research Journal, vol. 60, no. 6, pp. 351-359, 1990.

[8] Jayaraman, S., "Computer-Aided Design and Manufacturing: A Textile-Apparel Perspective", Invited Lecture at NATO Advanced Study Institute on Advancements and Applications of Mechatronics Design in Textile Engineering, Side, Turkey, April 3-16, 1992. Also, to appear in a NATO ASI Book, Kluwer Academic Publishers, The Netherlands.

[9] Etheridge, S., Gilley, D., Narasimhan, L., Narasimhan, S., and Jayaraman, S., "Justification of Information Technology Projects", SJ-WP-IT-9209, Working Paper, Georgia Institute of Technology, Atlanta, Georgia 30332-0295.

II. Abstract of Current & Projected Activities

The overall objective of this research effort is to design and develop a Manufacturing Enterprise Architecture which will serve as a blueprint for the implementation of CIM in the factory of the future. The apparel manufacturing sector has been used as the initial domain and an apparel manufacturing architecture (AMA) has been developed. Based on this work, a new methodology termed IFEM (integrated framework for enterprise modeling) has been developed. These concepts are being transformed into software. Work is also in progress on modeling an apparel plant and using this data to test and evaluate the MEA framework.

Research on developing the proposed methodology (IFEM) further will be continued. Implementation work utilizing object oriented programming techniques to develop the dynamics component of MEA will be continued. Work on modeling the fiber, yarn and fabric manufacturing sectors will also be continued. Since traditionally non-manufacturing sectors can benefit from the application of concepts from manufacturing research, the possibility of expanding MEA concepts to such domains will be examined during the coming year. Interactions with industry partners to implement research results will also be continued. Finally, ideas and plans for continuing this research after accomplishing the present research objectives will also be developed.

* * *

Attachments

An Integrated Framework for Enterprise Modeling

Rajeev Malhotra and **Sundaresan Jayaraman**, Georgia Institute of Technology, Atlanta, GA

Abstract

The IDEF modeling methodology of the US Air Force's integrated computer aided manufacturing (ICAM) program is a powerful tool for analysis, specification and design of integrated manufacturing systems. IDEF consists of methods for modeling the function structure, the data needed to support the functions, and the dynamic behavior of functions of a manufacturing enterprise. The resulting function, information, and dynamics models provide three distinct but complementary views of the system being modeled. A major deficiency of IDEF is the lack of cohesion between the three views whereby a single consistent description of the system is difficult to obtain, especially when the modeled domain is large and complex. Among its other limitations are difficulty in capturing the semantics of real-world systems in the information model, and a dynamics-modeling language unsuitable for modeling flexible manufacturing systems.

In this paper, we propose an integrated framework for enterprise modeling (IFEM) that extends the IDEF methodology to include methods that overcome the above-mentioned shortcomings of IDEF. The use of IFEM and its advantages over IDEF are illustrated using examples from a reference architecture developed for a computer-integrated apparel manufacturing enterprise.

Keywords: *Enterprise Modeling, Function Modeling, Data Modeling, IDEF Methodology, Computer-integrated Manufacturing (CIM), Relational Data Model*

Introduction

The term "computer-integrated manufacturing" (CIM) was first used by Harrington[1] to describe a control-and-communication structure that tied (integrated) various components of a manufacturing enterprise into a single cohesive system by facilitating prompt and efficient exchange of information between these components. Since information exchange is what integrates the manufacturing system components together, an enterprise-wide information system through which all the manufacturing system components communicate with each other is the core of a CIM system.[2] Typically, a manufacturing enterprise consists of a large number of interacting functional components making the design of a CIM information system a complex task. As CIM has moved from a vision to reality, the need for a well-defined methodology for design and analysis of CIM systems has become evident.[3-5] A few methodologies have evolved from the area of management information systems (MIS) designed to address this need. One such methodology is the integrated computer aided manufacturing definitions (IDEF)[6-8] methodology of the US Air Force.

IDEF is a set of three methodologies—$IDEF_0$[6], $IDEF_1x$[7] and $IDEF_2$[8]. An IDEF0 function model provides a structured representation of the functions that the enterprise performs and the interconnections that exist between these functions. The IDEF1x information-modeling methodology is used to define the structure of the information needed to support the functions. $IDEF_2$ is the dynamics-modeling methodology for modeling the time-varying behavior of the system components. The function, information, and dynamics models developed using the IDEF methodology are, collectively or individually, referred to as the architecture of the system because they are used to understand, analyze, and communicate how the various constituents of the system fit together and interact. The methodology is intended for modeling existing (AS IS) as well as proposed (TO BE) systems.

Our experience with IDEF in developing a reference architecture for an integrated apparel manufacturing enterprise (AMA)[9] has revealed many deficiencies in the methodology, some of which have also been documented by other users of the methodology.[10,11] Among them are weak cohesion between the function, information, and dynamics models; difficulty in capturing complete semantics of the modeled domain in the information model; and a dynamics–modeling language poorly suited for modeling CIM systems in which flexible manufacturing is implemented. In this paper, we discuss a framework for enterprise modeling that extends the IDEF methodology to overcome its shortcomings and to integrate the function, information, and dynamics models into a single cohesive representation of the system.

The shortcomings of the IDEF methodology and the need for enhancements to it are discussed in Section 2. The concepts underlying the proposed integrated framework for enterprise modeling (IFEM) are presented in Section 3. In Section 4, a part of the apparel manufacturing architecture is presented to illustrate the use of IFEM for modeling manufacturing enterprises and its advantages over the IDEF methodology.

Shortcomings of IDEF Methodology

Weak Cohesion Between IDEF Components

The $IDEF_0$, $IDEF_1x$ and $IDEF_2$ models represent different but complementary views of the same system.[12] For example, a machine is viewed as a resource in the $IDEF_2$ dynamics model, as a mechanism to perform an activity in the $IDEF_0$ function model, and as an entity about which data are maintained in the $IDEF_1x$ information model. Although the IDEF approach of viewing a manufacturing system in terms of its function structure, information structure and dynamics facilitates the modeling task by letting the modeler concentrate on one aspect of the system at a time, it also points to the need to integrate the three models into a single consistent architecture of the system. One of the major shortcomings of the IDEF methodology is that the relationships between the entities common to the three models are not captured in the IDEF architecture. As a result, the IDEF function, information and dynamics models lack cohesion. The modelers have to informally maintain these relationships to develop a consistent representation of the system—a task which becomes exceedingly difficult as the size and complexity of the modeled domain increase.

Limitations of $IDEF_1x$

$IDEF_1x$ information methodology is based on Codd's relational data model[13] in which data is defined as a set of independent tables and is a further development of the Entity-Relationship (E-R) data-modeling approach.[14] Although the relational model provides a conceptual definition of data that is independent of how the data may actually be stored on computer media, it has some inherent limitations in capturing the semantics of real-world entities.[15]

Poor Abstraction: The model limits attributes that describe the properties of an entity to single values of basic data types such as numbers and strings. Since the use of modeler-defined data types as domains for attributes is not permitted, entities cannot be defined as composites of simpler entities. Thus, the level of abstraction in the $IDEF_1x$ models is poor. Also, since only single-valued attributes are permitted, an attribute with multiple values, e.g., items purchased on a purchase order, has to be represented as a separate child entity which is related to the parent entity by migrating the parent's key attributes to the child. Thus, in an $IDEF_1x$ model, a real-world entity is fragmented into multiple entities (normalization) making the model difficult to understand.

Difficulty in Capturing Domain Constraints: Although the integrity constraints are well-captured in the $IDEF_1x$ model through the use of key-migration and specification of cardinality of relationships, the constraints that express semantics of the attributes (domain constraints) are difficult to express. For example, a constraint such as *the salary of an employee cannot exceed that of his supervisor* cannot be expressed in an $IDEF_1x$ model.

Instance-Identification Through Attribute Values: A subtle conceptual problem in modeling data using the relational-paradigm-based $IDEF_1x$ methodology results from the use of key attribute values to identify an instance of an entity. The implication is that an instance of an entity can exist only if key attribute values have been assigned to it. However, real-world artifacts can and often exist without having key attribute values assigned to them. For example, a purchase order, which is identified by a purchase order number, may come into existence and be fully prepared before a purchase order number, its key attribute, is assigned to it. To overcome this problem, an instance-of-an-entity needs to be referenced using an identifier that is independent of the values of its key attributes.

Limitations of $IDEF_2$

An $IDEF_2$ model is a discrete-event simulation model of a manufacturing system conceptually similar to general-purpose simulation languages such as SIMAN[16], GPSS[17], SIMSCRIPT[18] and SLAM[19]. These languages model the flow of entities such as materials and documents through a network of processing stations; an entity is processed at each station for a finite period of time and waits in a queue at the station if the resource associated with the station is busy. Some limitations of the general-purpose simulation languages are discussed by Adiga et al.[20] The limitations specific to $IDEF_2$ that make it unsuitable for modeling flexible manufacturing systems are as follows:

Lack of Abstraction and Parameterization: A serious limitation of the $IDEF_2$ methodology is a lack of means for abstracting repetitive details in the model into sub-models that are represented once and referenced in many places. The methodology also lacks the means for referencing resources, entities and process sequences as generic parameters.[10] With the lack of parameterization and abstraction, the process sequences have to be explicitly built into the entity flow networks—making it difficult to model manufacturing systems through which entities having different processing sequences could flow. As a result, an $IDEF_2$-model is a dynamics-model representing the flow of a specific type entity through a specific configuration of the manufacturing system. This approach is not consistent with the flexible manufacturing approach adopted in CIM systems, whereby the system is capable of being reconfigured dynamically for processing a variety of entities requiring different processing sequences.

Inflexible Dispatching Model: Another significant limitation of $IDEF_2$ in modeling CIM systems is its lack of flexibility in representing dispatching rules. In $IDEF_2$, the priority of an entity in a queue can be determined only at the time the entity is placed in the queue. In a CIM system with flexible manufacturing, the processing priority of an entity is often determined based on the existing state of the system at the time a resource becomes available to process the waiting entities. Due to this inflexibility in dispatching scheme, $IDEF_2$ is not well-suited for modeling CIM systems.

IFEM Static Architecture Concepts

Based on the limitations of the IDEF methodology discussed in the previous section, the following extensions and modifications to the IDEF methodology are needed:

1. To ensure that the function-, information-, and dynamics-models represent exactly the same domain and are consistent with each other, the relationships between the views of the system provided by these models must also be defined.
2. There is a need for a higher-level layer in the information model in which the entities appear in a composite form closer to their real-world equivalents, and constraints on data are better-expressed.
3. There is a need for a dynamics-modeling language that facilitates abstraction and parameterization to permit modeling temporal interactions between the activities performed by an enterprise system—without committing the model to the flow of a specific type entity through the system. The language should also provide clear and straightforward means for modeling various types of dispatch strategies.

These extensions to the IDEF methodology have been incorporated in the IFEM modeling framework proposed in this paper. In IFEM, the function-, information-, and dynamics-models of the system are integrated into a single framework by sharing the entities common to these models through cross-references between the models. A composite-view layer added to the $IDEF_1x$ information model not only addresses the shortcomings of the $IDEF_1x$ methodology, but also provides the means for integrating the information and function models into a *static* architecture. The inputs, controls, outputs, and mechanisms (ICOMs) for the functions modeled in the function model are defined in terms of the composite objects defined in the extended information model. In IFEM, instead of using $IDEF_2$ for dynamics-modeling, the function-model is extended to model the temporal interactions between the activities carried out in an enterprise. Thus, the IFEM dynamics model is an integral part of the function model.

Extensions to IDEF$_1$x

In the higher-level layer added to the information model, *views* composed of data elements from one or more IDEF$_1$x entities are constructed. These views represent the real-world entities in their composite form. The IFEM views implement the concept of abstract data types, which is the basis of semantic data models[21-23] developed to overcome the shortcomings of the relational model in capturing the semantics of real-world entities in a database schema. The IFEM views are defined as data structures in which the characteristics of the real-world entities are represented as *features* (*Figure 1*). Features are a more general case of the attributes that represent the characteristics of the entities in IDEF$_1$x models. The domain of a feature can be a basic datum-type such as a number or character string, or a composite-type defined by any of the views in the model. In addition, a feature can be a function that returns a value derived from other features in the view. The first-normal form constraint in IDEF$_1$x is also relaxed in the views, allowing a feature to be multi-valued. Constraints on features are expressed using conditional expressions to qualify the feature definitions.

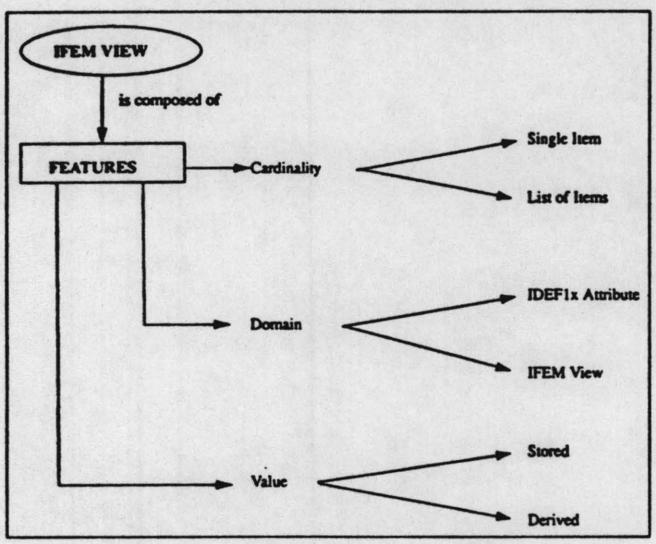


Figure 1
Structure of an IFEM View

The definition of a view consists of its unique identification number, its name, the names of its features and their definitions. *Figure 2* provides an example of an IFEM view to illustrate how various types of features are defined.

Basic Type: To associate IFEM views with the base IDEF$_1$x information model, the features with values of basic data types are defined in terms of their equivalent attributes in the IDEF$_1$x entity definitions. For example in *Figure 2*, the feature "Number" is defined as:

Number: E1.PONumber;

This definition implies that feature *Number* in IFEM view, F1/PURCHASE_ORDER is the same as the attribute PONumber in IDEF$_1$x entity, E1/PURCHASE_ORDER.*

Composite Type: A feature representing a composite object is defined in terms of the IFEM view that defines the object. For example, the value of the feature *Vendor* in the view F1/PURCHASE_ORDER is defined to be an object of type VENDOR as follows:

Vendor: F5 {VENDOR};

Through this feature definition, the relationship of the entity PURCHASE_ORDER to entity VENDOR is expressed. This definition does not require the migration of primary keys between the entities to express the relationship. Thus the entities can be modeled without identifying the primary keys.

Multi-valued: Multi-valued features are defined as a list of objects of a particular type. For example,

```
F1/PURCHASE_ORDER

Features:
   Number        : E1.PONumber;
   IssueDate     : E1.POIssueDate;
   DueDate       : E1.PODueDate;
   Vendor        : F5{VENDOR};
   Status        : E1.POStatus
                   WHERE Status IN [ISSUED, SHIPPED, CLOSED];
   Item          : LIST (F2{PO_ITEM});
   Amount        : DERIVED (Item.UnitCost, Item.Quantity);
```

Figure 2
An IFEM View Definition

*IFEM view names are prefixed with letter "F" while letter "E" is used as a prefix for IDEF$_1$x entity names.

the feature *Item* in the view F1/PURCHASE_ORDER is defined to have one or more purchased items as its value as follows:

Item: LIST (F2 {PO_ITEM});

Derived: A derived feature is defined as a function of the other features from which it is derived. For example, the feature Amount in F1/PURCHASE_ORDER represents a value derived from the features UnitCost and Quantity of the purchased items and is defined as follows:

Amount: **DERIVED** (Item.UnitCost, Item.Quantity); The procedure for deriving the value is not specified as it is implementation-specific. A comment is optionally used to explain what the derived feature represents.

Constrained: Constraints on the attribute values are stated using declarative expressions. For example, the constraint that the status of a purchase is either ISSUED, SHIPPED or CLOSED is expressed as:

Status: POStatus IN [ISSUED, SHIPPED, CLOSED];

Integration of $IDEF_0$ and $IDEF_1x$ Models

Integration of function and information models is achieved by associating ICOM interfaces to the functions with composite views in the information model. To better understand the semantics of ICOM interfaces before attempting such associations, the ICOMs are categorized based on the type of information they represent.

Classification of ICOMs: There are two orthogonal classifications for the ICOMs: structure-based and persistence-based. ICOMs may represent entities that are well-understood in terms of their structure, or abstract ideas and knowledge whose structure is either poorly understood or difficult to define. For example, knowledge about fashion trends constrains the garment design function, but it is difficult to represent this knowledge as structured information. In IFEM, ICOMs representing unstructured entities are classified as *free-form* (*F*) and the ones representing structured entities are classified as *structured* (*S*).

An ICOM may represent information of a transient nature, (e.g., start, stop and acknowledge signals, sent from one process to another as part of a hand-shaking protocol) that ceases to be of interest as soon as it is accepted by the receiver. Alternatively, an ICOM may represent persistent entities such as documents, parts and resources about which information needs to be maintained for a longer period of time. In IFEM, ICOMs are classified as *transient* (*T*) or *persistent* (*P*), depending on the nature of information they represent. Each ICOM is assigned a code specifying the type of entity it represents. For example the type code S/P in *Figure 3* signifies a structured and persistent entity.

Structural Definition of an ICOM: To associate ICOMs with their equivalent IFEM views in the information model, the structure of ICOMs is defined in terms of IFEM views. Since the information model contains the definitions of entities about which data are maintained and whose structure can be modeled, the structures of only those ICOMs that are of structured and persistent type are defined. An example of an ICOM definition from the integrated function and information model (static architecture) is shown in *Figure 3*. The ICOM *Issued Purchase Order* which is described as a purchased order that has been prepared and released to a vendor is defined structurally as follows:

F1 PURCHASE_ORDER WHERE Status IS ISSUED;

The above definition includes a constraint expression to express the proper semantics.

The definition of an ICOM may include only a selected set of features from a view instead of the entire view. For example, an ICOM representing a vendor's name and address could be defined as follows:

(Name, Address) FROM F5 {VENDOR};

The above definition implies that only the Name and Address features of the view F5/VENDOR are included in the definition of this ICOM.

IFEM Dynamics Modeling Methodology

In IFEM, the dynamic behavior of the functional components of the system and the dynamic interactions between these functions are modeled by extending the $IDEF_0$ function model after it has been integrated with the information model. The resulting dynamics model is completely integrated with the function and information models, and captures the dynamics behavior of all the functions of the enterprise included in the scope of the model instead of only those that participate in the flow of a particular type of entity.

The IFEM dynamics model consists of *scripts* that describe how each lowest-level function in the function model hierarchy behaves and interacts with other functions. The scripts capture the temporal interactions between the inputs, outputs, controls, and mechanisms of a function. The sequence of dynamic actions performed by a function to transform its *input* entities into its *output* entities is expressed in a script for the function. The *mechanism* entities represent the resources that must be available for the transformations to take place, and *control* entities constrain the transformation process. Examples of dynamic actions include engagement and release of resources, retrieval of entities from input queues, release of transformed entities into output queues, etc. From the viewpoint of the IFEM dynamics model, an ICOM interface represents a channel along which entities can be moved between functions interconnected through the interface. The entities awaiting processing can queue up in the ICOM channel. The IFEM approach to dynamics modeling is illustrated with the help of a simple example.

IFEM Dynamics Modeling Approach: Consider the example of a model of a flexible machining system (FMS). One of the functions of a FMS is to load a part on a machine (*Figure 4*). The input to this function is the part to be loaded (I1) and the output is a part loaded on the machine (O1). The resources used are a loading robot (M1) and a machine on which the part is loaded (M2). The function is controlled by the process sequence of the part (C1). The dynamics script for this function expressed in textual form is as follow:

1. Begin loading when a part is available at I1 and a loading robot is available at M1.
2. Select a part from input I1.
3. Lookup the process sequence for the part from control C1 to determine on which machine the part should be loaded next.
4. Engage that machine from M2 when it becomes available.
5. Engage an available robot M1 to carry out the loading.
6. Load part I1 on machine M2.
7. Release the loading robot M1.
8. Release the part loaded on the machine as output O1.
9. End loading

In this description of the *Load Machine* function, entities and resources, e.g., parts and robots, are referenced using parameters; there is no reference to any specific part or robot. When the function is activated for loading a particular part, the selection of a specific robot and a machine is made based on the process sequence for the part. This description also provides a high level of abstraction in the model because it describes the loading of any part on any machine in generic terms. In the system being described, it is possible that two or more parts are being processed in the system at any given time and many machines get loaded simultaneously. In $IDEF_2$, this situation would be modeled by having one node for each loading station in the system. In IFEM, this situation is treated as multiple concurrent

Issued Purchase Order

Type: S/P

Structure:

F1{PURCHASE_ORDER}WHERE Status IS "ISSUED";

Description:

A document sent to a vendor listing items to be purchased, required quantities and delivery dates.

Figure 3
An IFEM Definition for an ICOM Interface

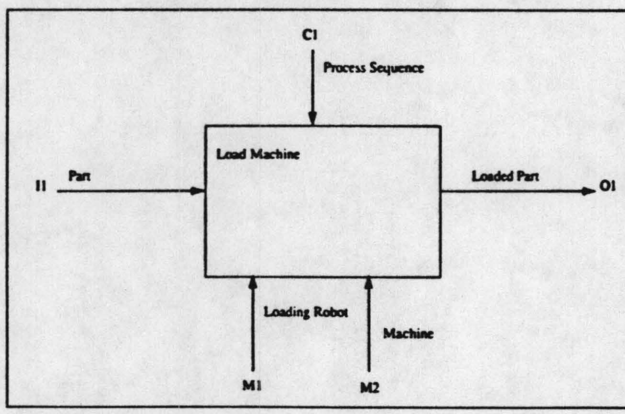


Figure 4
The *Load Machine* Function

activations of the same function *Load Machine*, and not as multiple nodes. The number of concurrent activations depends on the availability of resources and other conditions. For example, the function *Load Machine* is activated when a loading robot is available and a part is waiting to be loaded. The conditions that trigger the activation of a function are specified in IFEM as part of the dynamics model. Treating multiple processes as activations of a single function permits development of models that are independent of actual size of the system. For example, since the IFEM description of the dynamics of the *Load Machine* function is independent of the number of robots and machines in the FMS, adding an extra robot or a machine to the FMS being modeled does not require alterations to its dynamics model.

The feature that makes abstract and parameterized descriptions possible and distinguishes IFEM dynamics model from an $IDEF_2$ model is that the process sequence is not considered a part of the function's dynamic description, but is treated as a data input to it. Thus, the IFEM dynamics modeling approach is particularly well-suited for modeling flexible data-driven processes that typically constitute CIM systems.

To ensure consistency of the dynamics model with the static architecture, the dynamics scripts adhere strictly to the context provided by the ICOM interfaces of each function. A function has access to only those entities that are represented by its inputs and controls it can process (i.e., alter) and release only those entities that are represented by its outputs and it can use only those resources that are represented by its mechanisms.

IFEM Dynamics Modeling Language: The dynamic actions modeled in the scripts are represented using a set of primitives listed in *Figure 5*. The ENGAGE and DISENGAGE primitives apply to the mechanisms' interfaces and represent grabbing and release of the resources. The RETRIEVE and COLLECT primitives are used to pick entities from input or control interfaces for processing. The ASSIGN and PROCESS primitives are used to alter the state of the processed entities by changing the values of their attributes. The processed entities are released to an output interface using the RELEASE primitive. The LOOKUP primitive is used to search for entities at an interface. A script begins with the TRIGGER primitive that specifies the condition activating a function; the script ends with an END, which terminates a particular activation. The modeling language also provides control primitives for conditional branching and repetitive actions. In general, a primitive has the following syntax:

Operation ICOM [*Selection Criteria*] [*Delay*];

Operation is the dynamic action that the primitive represents and ICOM is the interface on which the action is performed. The entities affected by the action are selected from the ICOM interface based on the specified *Selection Criteria*. If the time taken to complete an action is of interest, it is represented as *Delay*. Consider the following example:

RETRIEVE I1 [I1.Color IS 'RED'] [D1];

This primitive represents retrieval of all entities, whose feature *Color* has value 'RED', from input interface I1. The time taken to complete this action

ENGAGE Mx [*selection criteria*];	Engage resources from interface Mx to carry out a process
DISENGAGE Mx [*selection criteria*];	Free up resources previously engaged from interface Mx
RETRIEVE Ix (or Cx) [*selection criteria*] [*delay*];	Retrieve entities from interface Ix or Cx for processing
COLLECT Ix (or Cx) [*selection criteria*] [*delay*];	Retrieve entities and add them to a set of previously retrieved entities
ASSIGN Ox.*Feature*. <- *value* [*selection criteria*] [*delay*];	Assign a value to the specified feature of an entity associated with interface Ox
Process Ox [*selection criteria*] [*delay*];	User-defined primitive for operating on entities associated with interface Ox
RELEASE Ox [*selection criteria*] [*delay*];	Release processed entities at interface Ox
RELEASE Ox.*Feature* <- *value* [*selection criteria*] [*delay*];	Release entities after assigning a value to the specified feature
LOOKUP Ix (or Cx, Ox, Mx) [*selection criteria*] [*delay*];	Lookup data representing entities associated with an ICOM interface
WAIT [*duration* / UNTIL *time*];	Wait for a specified duration or until a specified time
TRIGGER *condition*;	Activate a function when the specified condition is met
END;	Terminate a function
IF [*condition*] (*block 1*) ELSE (*block 2*);	Choose between alternate sequences of actions on the basis of the specified condition
WITH *p* (*block 1*) ELSE (*block 2*);	Choose between alternate sequences of actions depending on probability *p*
REPEAT [*control*] (*block*);	Repeat a sequence of actions
Control:	
[*n* TIMES]	Repeat *n* times;
[EACH *X* IN *collection*]	Repeat for each item *X* in the *collection*;
[WHILE *condition*]	Repeat while *condition* holds true;
[UNTIL *condition*]	Repeat until *condition* becomes true;

Ix: Input Cx: Control Ox: Output Mx: Mechanism
Selection Criteria: Criteria based on which entities are selected for an action
Delay: Time taken to carry out an action
Block: A sequence of actions

Figure 5
The IFEM Dynamics Modeling Primitives

is D1. As illustrated by the above example, the removal of entities from an input queue takes place in an order determined by the selection criterion and not necessarily in the order in which the entities are placed in the queue. Selection criteria can be used to specify a range of dispatching rules—from simple first-in-first-out to a selection based on a complex set of conditions.

IFEM Architecture for Apparel Manufacturing

We have used IFEM to develop a reference architecture for a computer-integrated apparel manufacturing enterprise. The architecture consists of the information-, function-, and dynamics-models of the enterprise. As a first step in the development of the architecture, modeling and analysis of the operations of an existing (AS IS) apparel enterprise were carried out using the $IDEF_0$ function-modeling methodology.[24,25] The entities processed by the enterprise and represented as ICOM interfaces in the function model were described in detail in the model glossary. Through the analysis of this AS IS function model and the contents of its glossary, a single integrated definition of the enterprise data was developed using the $IDEF_1x$ information-modeling methodology. This model serves as the conceptual schema for the integrated database that would support the operations of the proposed (TO BE) apparel enterprise. The apparel enterprise functions that access the data defined in the $IDEF_1x$ model were modeled in the function model. The interfaces of these functions to the enterprise data were modeled using the IFEM extensions to IDEF. A part of the architecture[26] is discussed here to illustrate how IFEM is used in enterprise modeling.

Function Model

Figure 6 depicts the top-level functions of the apparel manufacturing enterprise. Garment styles are developed using the inputs from the customers (A1). Production is planned for the developed styles and the necessary materials are procured (A3). Manufacturing orders are prepared, scheduled and released to the manufacturing plants, and their progress monitored (A4). Garments are manufactured, inspected and sorted (A5). Manufactured garments are packed and shipped to customers to fulfill orders (A6). The information on materials, quality standards and manufacturing resources required to support the design, planning and manufacturing functions is created and maintained (A2). The information passed between the functions is represented by the interface arrows on the diagram.

Each top-level function is detailed further into its sub-functions. For example, the diagram detailing the *Manufacture Garment* function is shown in *Figure 7*. Fabric is cut and all the materials necessary for producing garments for a scheduled order are packaged together and shipped to the manufacturing plants (A51). The production schedule is distributed to each manufacturing plant (A52). The garments are produced from the cut fabric parts and other materials shipped to them (A53). Quality audit is performed on the assembled garments (A54). The *Produce Garments* function (A53) is detailed further in *Figure 8*. Machines and operators are assigned to carry out the production tasks for an order (A531). Sewing and finishing operations are performed on the cut parts (A532). Finished garments are graded and sorted, and sent for quality audit (A533). Accessories such as belts, tags and hangers are attached to the garments that pass the quality audit (A534).

The *Sew-and-Finish-Garments* function (A532) is carried out in manufacturing modules whose functions are shown in *Figure 9*. This function models the activities of the sewing-and-finishing shopfloor. The cut fabric parts and construction materials required to produce garments for an order are represented as input interface *Cut Package Shipment*. The control interface to this function—*Assignments—Plant Resources*—represents the assignment of resources for each step involved in producing garments for the order. The resources assigned to produce garments are grouped into functional modules, each containing one or more workstations and designated to perform a set of operations. For example, six modules may be assigned for an order—one each for front, back, waistband and final assembly of the garment, and two for finishing. One or more operators are assigned to operate workstations in each module. The outputs from the function are finished garments and production status information represented by output interfaces *Garments—Finished* and *Production Status*, respectively.

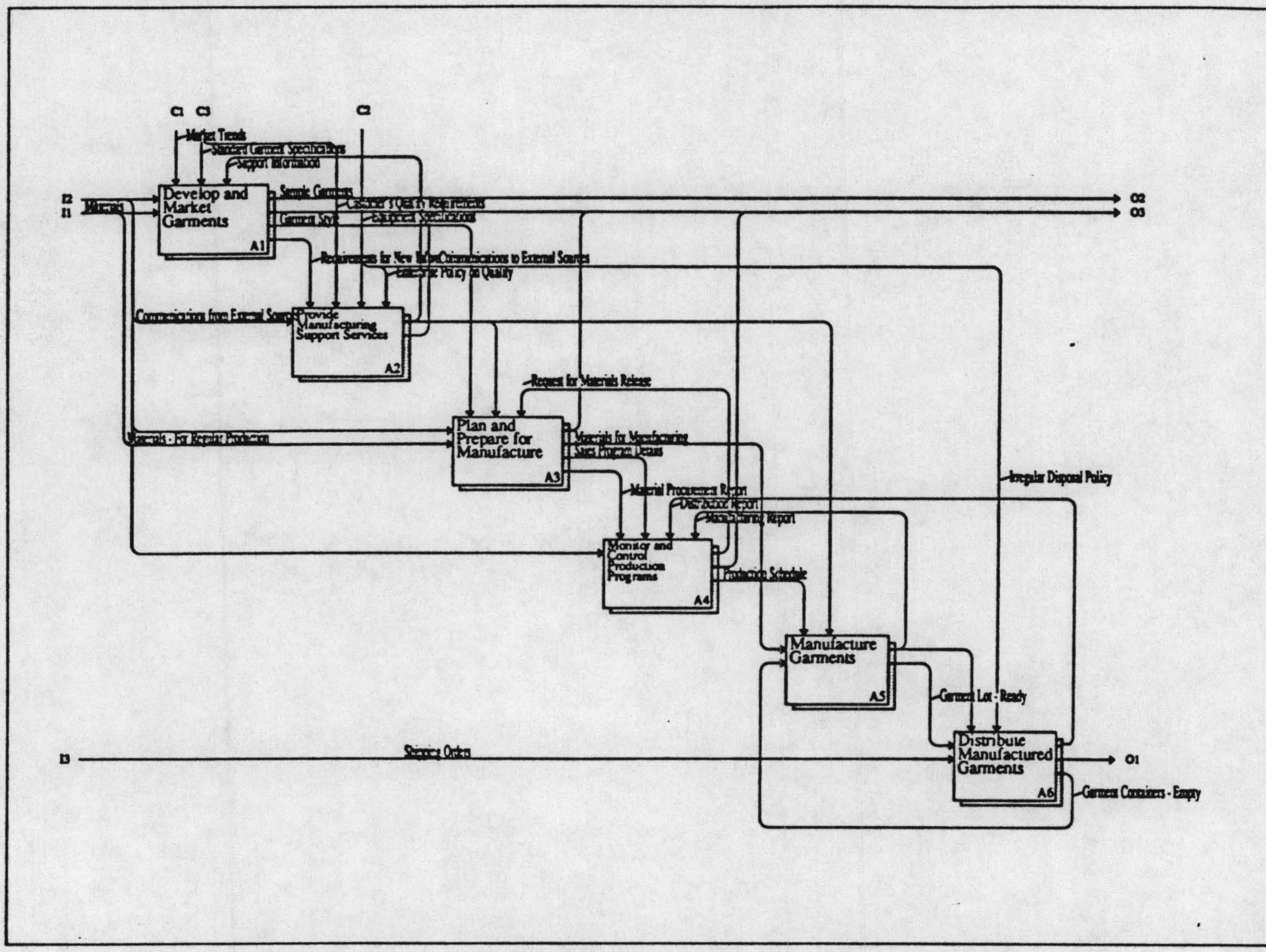


Figure 6
Top-Level Function Diagram of Apparel Manufacturing Enterprise

The *Sew-and-Finish-Garments* function is detailed further to represent the basic shopfloor activities. The production operations are carried out at the sewing and finishing modules assigned to the order. Garment sub-assemblies for an order are held in a storage buffer from where they are taken to the modules for processing and brought back by transport devices serving the modules. The movement of sub-assemblies to the modules is controlled by the shopfloor controller that routes the sub-assemblies to the appropriate modules and ensures the correct sequencing of operations.

Information Model

Base IDEF$_1$x Model: The entities such as machines, operators, garment sub-assemblies, and production schedules involved in the garment manufacturing functions (discussed above) are defined in the IDEF$_1$x information model that complements the function model. *Figure 10* shows the entities for scheduling a production order in a plant and for allocation of manufacturing resources. An order scheduled in a plant for a particular period is represented as E77/PLANT_SCH_ITEM. The entity E29/EQUIP_GROUP consists of workstations, storage buffer and transporters logically grouped together to work as a line or a module (*Figure 11*). The lines or modules assigned for producing garments for an order are represented as E78/ASSIGNED_EQUIP. One or more lines or modules may be reserved for an order. For example, four modules may be reserved for a production order, one each for trouser front assembly, back assembly,

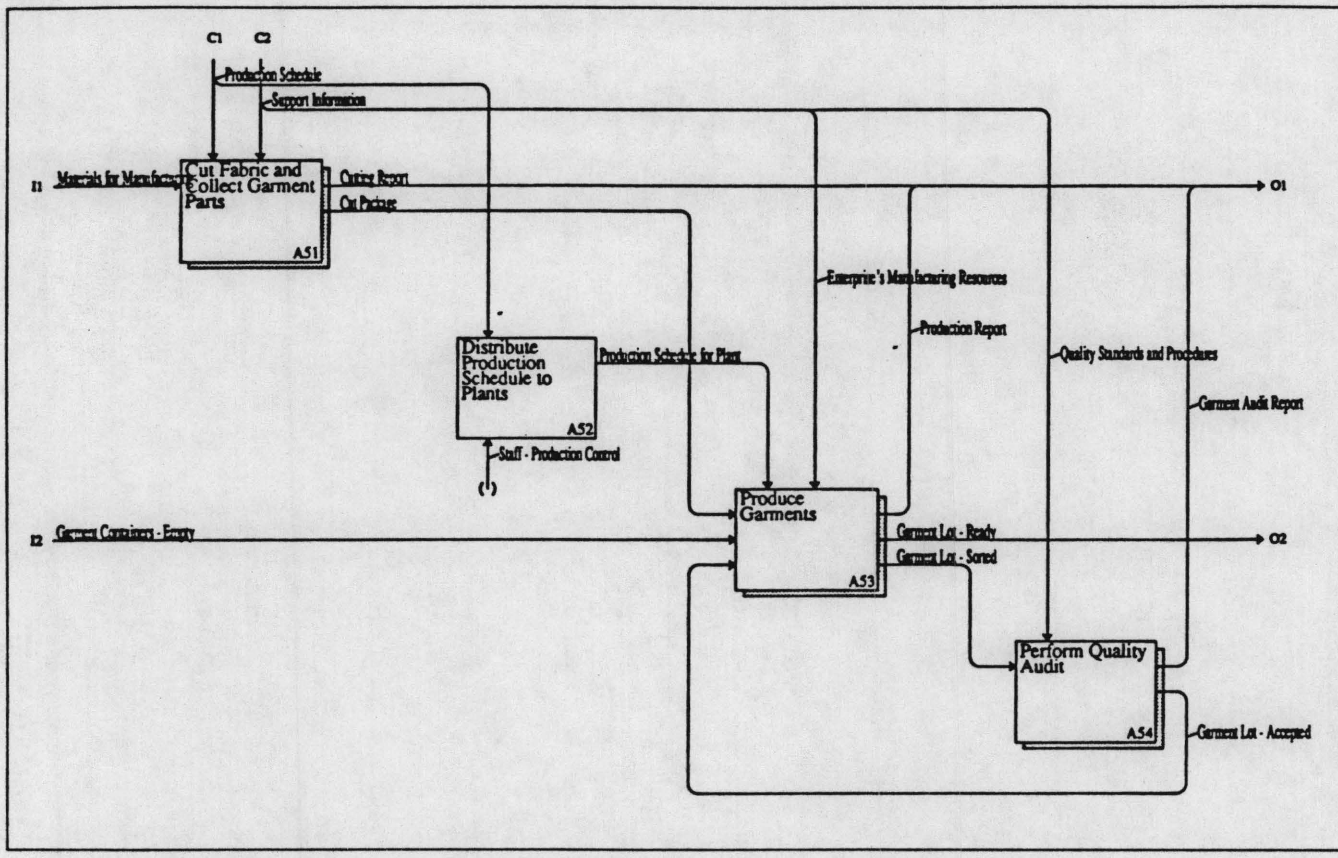


Figure 7
The *Manufacture Garment* Function

final assembly and finishing. Each process step assigned to a module is represented as the entity E81/WORK_ASSIGNMENT. Each module is assigned one or more operators. Garments produced for an order are represented as E79/GARMENT_UNIT (*Figure 12*). A garment is assembled from parts represented as E101/GAR_SUBASSEMBLY. During the manufacturing process, parts are progressively assembled into higher-level subassemblies till the complete garment is obtained.

IFEM Views: *Figure 13* shows the IFEM views that represent the composite forms of the $IDEF_1x$ entities representing production schedule, assigned resources, garments, and garment sub-assemblies. These views have been used to define the structure of the ICOM interfaces of the functions involved in sewing and finishing of garments. A few of these definitions taken from the model glossary are shown in *Figure 14*. For example, the ICOM *Assignment-Plant Resources* represents a structured and persistent data entity, and is defined as:

Structure: (Assgnmnt) FROM
F77 {SCH_PROD_ORD};

The feature *Assgnmnt* in the view F77/SCH_PROD_ORD is defined as a list of resource assignments (F78/PROD_ASSGNMT) for the scheduled production order (*Figure 13*). Based on the above definition of *Assignment-Plant Resources*, the function *Control Sewing and Finishing Production* has access to the data on equipment, operators and operations assigned to a production order through this ICOM interface.

Similarly, the other ICOM interfaces in the function model are classified according to the types of entities they represent and, where appropriate, defined in terms of IFEM views. For the function and information models to be consistent, if an ICOM represents structured and persistent entities, the definitions of these entities *must* exist in the information model. The information model underwent numerous revisions to remove any inconsistencies

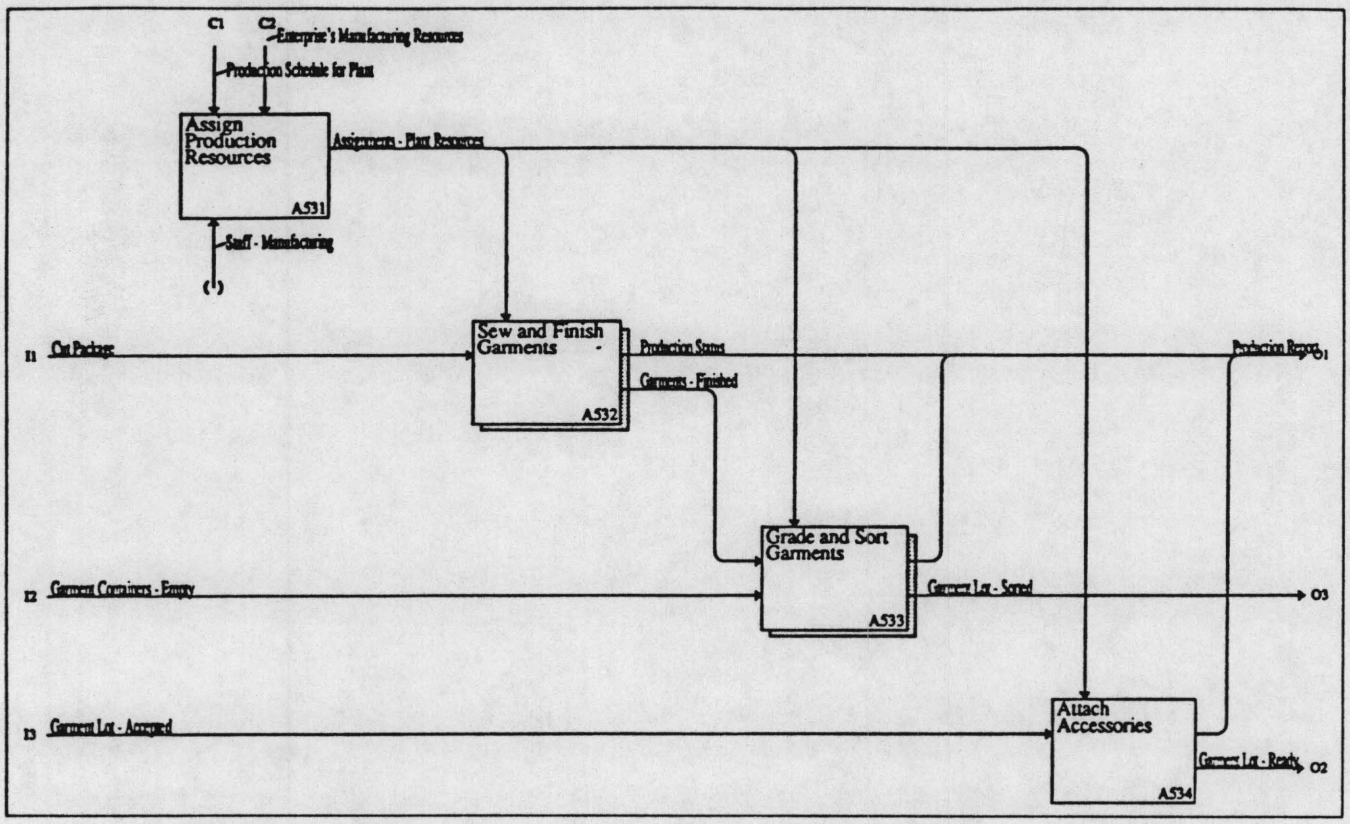


Figure 8
The *Produce Garments* Function

that were discovered while defining the ICOM interfaces in terms of information model entities. Additionally, precise and unambiguous definitions of ICOM interfaces resulted in greater clarity in the function model.

The Dynamics Model

To illustrate the use of IFEM for dynamics-modeling, part of this model covering the functions (*Figure 9*) involved in transforming cut fabric parts into finished garments is discussed. The script corresponding to each lowest-level function under the *Sew-and-Finish-Garments* node describes how that function is activated and how it behaves once it is activated. The structure definitions of the entities available to each function at its interfaces (ICOMs) are contained in the function-model glossary, a part of which is shown in *Figure 14*.

The Control Function: The shopfloor control activities are represented by the function *Control Sewing and Finishing Production* (A5321). The script describing the dynamics of this function is shown in *Figure 15*. This function is activated when the entity *Assignments—Plant Resources* is available at the control interface C1. This entity is retrieved from C1 by the RETRIEVE primitive. The retrieved entity contains a list of assignments *Assgnmt* for individual modules to be used for producing garments in the order. The assignments for modules (Equip) with function 'SEWING' or 'FINISHING', are queued at the output interface O2.

The sequence of actions inside the REPEAT loop is carried out until the status of the entire order is changed to 'FINISHED'. From C3, the subassemblies that belong to the order being processed and selected by the *Select1* module for further processing, are retrieved. The entities retrieved from C3 are processed by the *Proc1* module and released at the interface O3 by the RELEASE primitive. Next, the sewing and finishing assignments for this order whose status is not 'DONE' are looked up from the interface C2. If C2 does not contain any assignment with status not 'DONE', i.e., all the assignments are finished, the entity *Production*

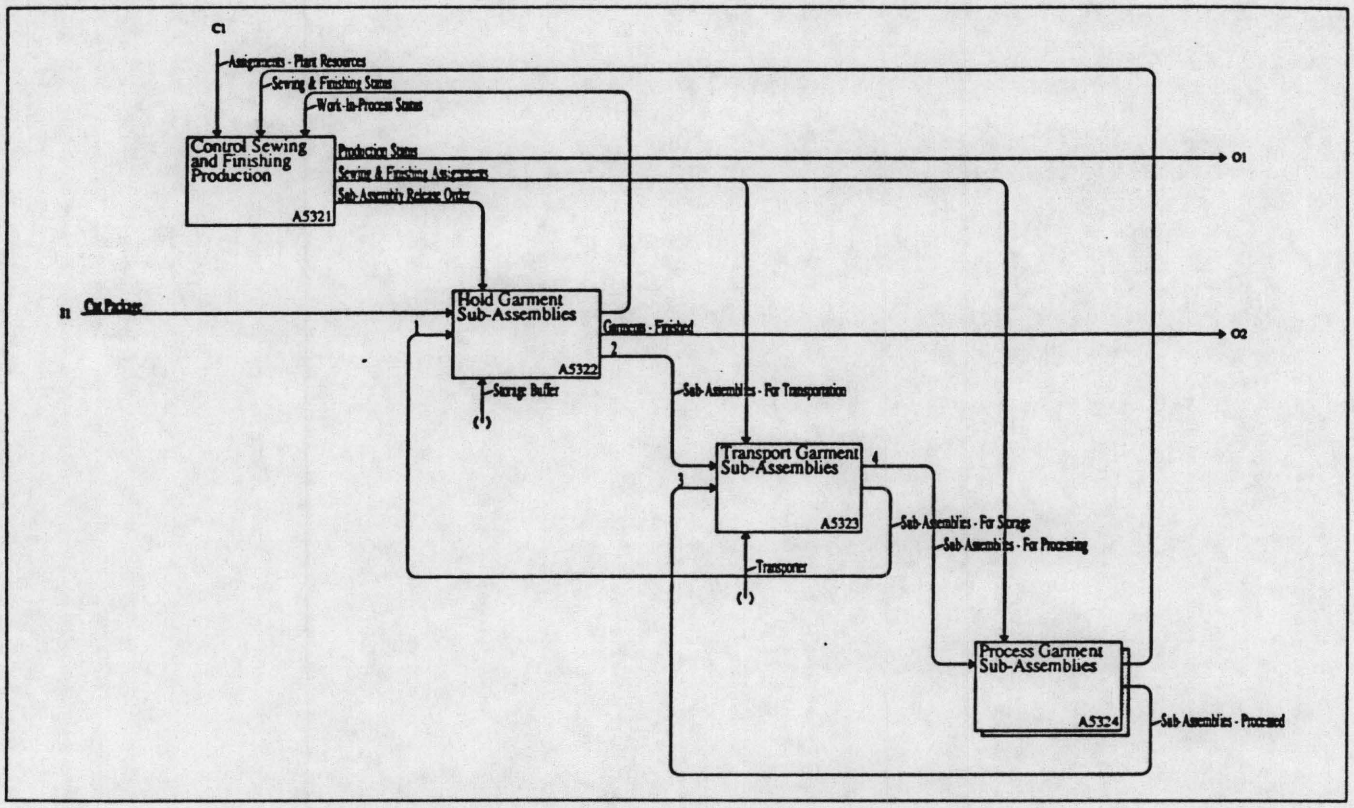


Figure 9
The *Sew and Finish Garment* Function

Status is released at the interface O1 after updating its status to 'FINISHED'. Once the REPEAT loop is exited, the function activation is terminated by the END primitive.

In the dynamics script for the *Control Sewing* and *Finishing Production* function, modules for performing two types of tasks are identified: *control* and *processing*. The control module *Select1* is a selection function that encapsulates the logic used by the controller to select sub-assemblies for further processing. The processing module *Proc1* encapsulates the procedure for assigning the next processing location to which the selected sub-assemblies have to be routed.

The Buffering Function: The function *Hold Garment Sub-Assemblies* (A5322) represents buffering of in-process sub-assemblies. The script describing the dynamics of this function is shown in

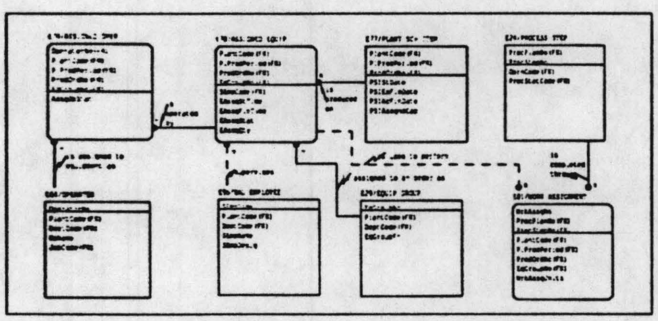


Figure 10
$IDEF_1x$ Entities for Assigning Manufacturing Resources to Orders

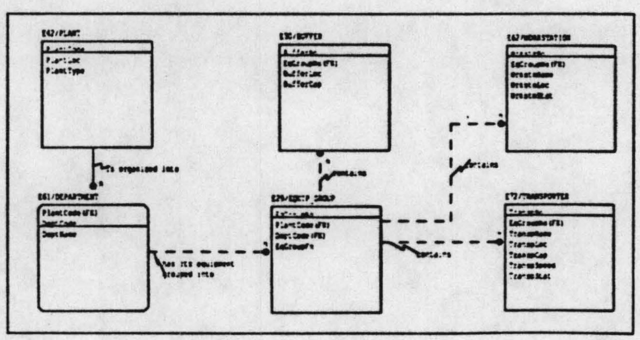


Figure 11
$IDEF_1x$ Entities Describing Manufacturing Resources

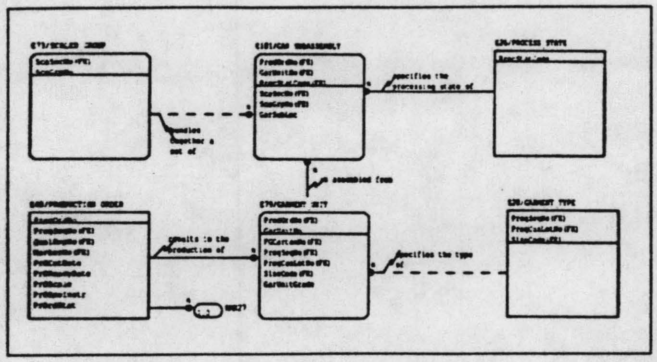


Figure 12
IDEF$_1$x Entities Describing a Garment and its Components

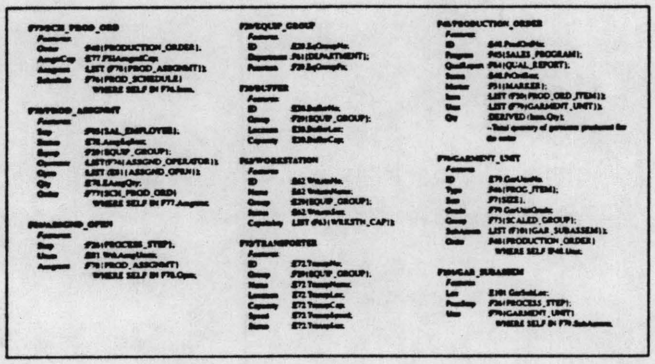


Figure 13
IFEM Views of the IDEF$_1$x Entities

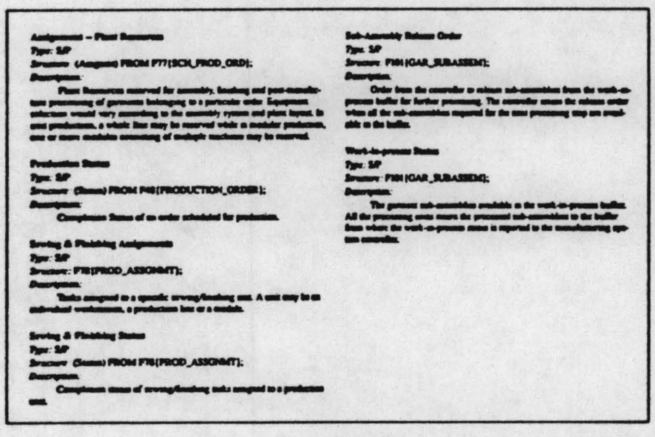


Figure 14
Definitions of ICOM Interfaces in Terms of IFEM Views

Figure 16. This function is activated by the arrival of the entity *Cut Package* at the input interface I1. A storage buffer is engaged from interface M1. A cut package is retrieved from input I1. All the sub-assemblies contained in the entity retrieved from I1 are released as work-in-process status at the

```
Control Sewing and Finishing Production

Description:
   Control the real-time operation of sewing/finishing lines or mod-
   ules. Move each garment unit through the entire sequence of process
   steps while tracking its status. Also ensure that the process steps
   are performed in the correct sequence specified on the process plan.

Interface:
   C1: Assignments - Plant Resources;
   C2: Sewing & Finishing Status;
   C3: Work-In-Process Status;
   O1: Production Status;
   O2: Sewing & Finishing Assignments;
   O3: Sub-Assembly Release Order;

Dynamics:
   Select1:[Selects sub-assemblies for further processing];
   Proc1 :[Assigns the next processing location to selected sub-as];
   TRIGGER: C1;
   RETRIEVE C1;
   REPEAT [EACH X IN C1.Assgnmt]
      IF [X.Equip.Function IN ('SEWING', 'FINISHING')]
         RELEASE O2 [O2 IS X];
   REPEAT [UNTIL C1.Order.Status IS 'FINISHED']
      (
      RETRIEVE C3 [C3.Unit.Order IS C1.Order AND Select1];
      Proc1 O3 [O3 IS C3];
      RELEASE O3;
      LOOKUP C2 [C2 IN C1.Assgnmt AND C2.Status NOT 'DONE' AND
              C2.Equip.Function IN ('SEWING', 'FINISHING')];
      IF [C2 IS NULL]
         RELEASE O1.Status <- 'FINISHED' [O1 IS C1.Order];
      )
   END;
```

Figure 15
Dynamics Description Script for the Control *Sewing and Finishing* Production Function

output O1; this output provides the *Control Sewing* and *Finishing Production* function with a list of sub-assemblies available in the buffer.

The sequence of actions within the REPEAT loop is executed until the status of the order for which the cut package was retrieved from I1 becomes 'FINISHED'. The sub-assemblies marked for further processing by the *Control Sewing* and *Finishing Production* function are retrieved by the RETRIEVE primitive. The retrieved sub-assemblies are released for transportation to the manufacturing modules at the output interface O3. The processed sub-assemblies transported back from the manufacturing modules are retrieved from I2. These sub-assemblies are released for the *Control Sewing* and *Finishing Production* function at O1 by the RELEASE primitive.

When the REPEAT loop is exited, the finished garments are released at the interface O2 by the RELEASE primitive. Next, the storage buffer engaged from M1 is released and the function activation is terminated.

The Transport Function: The movement of sub-assemblies between the modules and storage is represented by the function *Transport Garment Sub-Assemblies* (A5323). The script describing the

```
Hold Garment Sub-Assemblies

Description:
   Hold the garment sub-assemblies between process steps. Update the
location of each garment sub-assembly received in the buffer.

Interface:
   I1: Cut Package;
   I2: Sub-Assemblies - For Storage;
   C1: Sub-Assembly Release Order;
   O1: Work-In-Process Status;
   O2: Garment - Finished;
   O3: Sub-Assemblies - For Transportation;
   M1: Storage Buffer;

Dynamics:
   TRIGGER :I1;
   ENGAGE M1;
   RETRIEVE I1;
   RELEASE O1 [O1 IN I1.Unit.SubAssem];
   REPEAT [UNTIL I1.Status IS 'FINISHED']
      (
      RETRIEVE C1 [C1.Unit.Order IS I1];
      RELEASE O3 [O3 IN C1];
      RETRIEVE I2 [I2.Unit.Order IS I1];
      RELEASE O1 [O1 IS I2];
      )
   RELEASE O2 [O2 IN I1.Unit];
   DISENGAGE M1;
   END;
```

Figure 16
Dynamics Description Script for the *Hold Garment Sub-Assemblies* Function

```
Transport Garment Sub-Assemblies

Description:
   Move garment sub-assemblies between storage buffer and processing
units.

Interface:
   I1: Sub-Assemblies - For Transportation;
   I2: Sub-Assemblies - Processed;
   C1: Sewing & Finishing Assignments;
   O1: Sub-Assemblies - For Processing;
   O2: Sub-Assemblies - To Buffer;
   M1: Transporter;

Dynamics:
   D1        :[Time to transport];
   TRIGGER  :C1;
   RETRIEVE C1;
   REPEAT [UNTIL C1.Status IS 'DONE']
      (
      ENGAGE M1 [M1.Group IS C1.Equip];
      #Transport from storage to processing unit
      RETRIEVE I1 [I1.Loc IS C1.Equip];
      RELEASE O1 [O1 IS I1] [D1];
      #Transport from processing unit to storage
      RETRIEVE I2 [I2.Loc IS C1.Equip];
      RELEASE O2.Loc <- NULL [O2 IS I2] [D1];
      DISENGAGE M1;
      )
   END;
```

Figure 17
Dynamics Description Script for the *Transport Garment Sub-Assemblies* Function

dynamics of this function is shown in *Figure 17*. This function is activated when the entity *Sewing & Finishing Assignments* becomes available at the control interface C1. This entity is retrieved from C1.

The sequence of actions within the REPEAT block is executed until the status of the entity retrieved from C1 becomes 'DONE'. A transporter resource, e.g., a trolley or a conveyor, that belongs to the manufacturing module to be used for the assignment (retrieved from C1) is engaged. The sub-assemblies released for further processing by the *Hold Garment Sub-Assemblies* function, and routed to the manufacturing module served by this transporter, are retrieved from I1 and released at the interface O1 for processing at the module. The processed sub-assemblies are retrieved from I2. The *Loc* attribute of these sub-assemblies is assigned the value NULL before releasing at O2 for the *Hold Garment Sub-Assemblies* function. The transporter resource engaged from M1 is released. When the REPEAT block is exited, the function is terminated by the END primitive.

The Processing Function: The activities of a manufacturing module are modeled as the function *Process Garment Sub-Assemblies* which is broken down further. The dynamics of this function are described by the scripts developed for its sub-functions. This function represents a manufacturing module that performs the assigned process steps on the input sub-assemblies and returns them as *Sub-assemblies—Processed*. When the assigned work is completed, the status of the work assignment is updated to 'DONE'.

Conclusion

As a language for modeling CIM systems, IFEM addresses the shortcomings of the IDEF methodology and enhances its expressive power. In IFEM, the information- and function-models are integrated into a static architecture of the system being modeled by defining the inputs, outputs, controls and mechanisms for each function in terms of entities defined in the information model. The IFEM dynamics model is developed as an extension of the static architecture of the system. Consistency checking between the integrated models is facilitated in the following ways:

1. Precise meaning is imparted to ICOMs based on the rigorous definitions of the entities present in the information model.
2. It is easy to determine whether all the data necessary to support the functions modeled in the function model are defined in the information model.

3. The IFEM dynamics model uses the entity and function definitions from the static architecture and strictly adheres to the context provided by the static architecture.

The IFEM view-layer added to the information model provides a higher-level abstraction of data in the form of composite views representing the real-world entities being modeled. Meaningful constraints reflecting the semantics of real-world entities being modeled are expressed. The IFEM view layer provides the means for integrating the function and information models.

In IFEM, a dynamics-modeling approach radically different from $IDEF_2$ is adpted. Instead of modeling a sequence of steps involved in processing one particular entity, the temporal interactions between the inputs, controls, outputs and mechanisms of functions are modeled, yielding a description that is not tied to the process-sequence of any specific entity. The IFEM dynamics-modeling methodology also permits greater flexibility in modeling dispatching rules through the use of complex selection criteria for picking entities from queues for processing.

By defining the structure of ICOM interfaces to the functions in terms of entities defined in the information model, the expressive power of the function model is also enhanced. In IFEM architecture, the functions of the enterprise are viewed as applications that reference or manipulate the data maintained in the enterprise database. Incidently, the functions that physically transform entities, e.g., drill holes in a part, move products from storage to packing area, etc., also transform the data entities that are abstract representations of the corresponding physical entities. Thus, an IFEM function model not only depicts the function structure of the enterprise being modeled, but it also specifies the interface of each function to the enterprise database that forms the core of a CIM system.

The classification of ICOM interfaces to functions based on the nature of information represented by them plays a very important role in the IFEM modeling process. Enterprise functions with ICOMs representing free-form information cannot be completely automated as a human is required to process the free-form information. Thus the methodology can also be effectively used to identify the functional areas where complete automation is not feasible.

Acknowledgments

This work was supported through research funds from the US Defense Logistics Agency (DLA-900-87-D-0018), and in part by an NSF Presidential Young Investigator Award (DDM-8957861). The participation of Oxford Slacks of Monroe, GA, in the development of the apparel manufacturing architecture is gratefully acknowledged. The authors also thank the reviewers for their comments on an earlier version of the paper.

References

1. J. Harrington Jr., "A Look Into Tomorrow," *Automation*, May 1960, pp. 66-70.
2. W.B. Beeby, "The Heart of Integration: A Sound Data Base," *IEEE Spectrum*, May 1983, pp. 44-8.
3. Computer Aided Manufacturing—International, "The Architecture for Discrete Part Manufacturing Enterprise," *CIM Review*, Fall 1988, pp. 7-13.
4. G.T. Mackulak, "High Level Planning and Control: An $IDEF_0$ Analysis for Airframe Manufacture," *Journal of Manufacturing Systems*, Vol. 3, No. 2, 1984, pp. 121-32.
5. R.W. Yeomans, A. Choudry, and P.J.W. ten Hagen, *Design Rules for a CIM System*, North Holland, Netherlands, 1985.
6. "Integrated Computer-Aided Manufacturing (ICAM), Function Modeling Manual $IDEF_0$," Materials Laboratory, Air Force Wright Aeronautical Laboratories, AFSC, Wright-Patterson AFB, OH, 1981.
7. "Integrated Information Support System (IISS), Information Modeling Manual $IDEF_1x$," *ICAM Project Priority 6201*, D. Appleton Company Inc., Manhattan Beach, CA, 1985.
8. "Integrated Computer-Aided Manufacturing (ICAM), Dynamic Modeling Manual $IDEF_2$," Materials Laboratory, Air Force Wright Aeronautical Laboratories, AFSC, Wright-Patterson AFB, OH, 1981.
9. R. Malhotra, "An Architecture for an Apparel Manufacturing Enterprise," *Doctoral Dissertation*, Georgia Institute of Technology, Atlanta, March 1991.
10. A.N. Godwin, J.W. Gleeson, and D. Gwillian, "An Assessment of the IDEF Notations as Descriptive Tools," *Information Systems*, Vol. 14, No. 1, 1989, pp.13-28.
11. D. Brown, "Minutes of the IDEF-U.G. Research Issues Group," *Proceedings of the IDEF Users Group Conference*, Scottsdale, AZ, May 1989.
12. R.R. Bravoco and S.B. Yadav, "A Methodology for Modeling the Dynamic Structure of an Organization," *Information Systems*, Vol. 10, No. 3, 1985, pp. 299-317.
13. E.F. Codd, "A Relational Model of Data for Large Shared Data Banks," *Communications of ACM*, Vol. 13, No. 6, June 1970, pp. 377-87.
14. P.P.S. Chen, "The Entity-Relationship Model—Towards a Unified View of Data," *ACM Transactions on Database Systems*, Vol. 1, No. 1, March 1976, pp. 9-36.
15. D.S. Bower, "From Database to Information Base: Some Questions of Semantics and Constraints," *Information and Software Technology*, Vol. 31, No. 8, October 1989, pp. 402-10.
16. C.D. Pegden, "Introduction to SIMAN," System Modeling Corporation, Sewckley, PA, 1982.
17. G. Gordon, "System Simulation," Prentice Hall Inc., Englewood Cliffs, NJ, 1978.
18. E.C. Russell, "SIMSCRIPT II.5 and SIMGRAPHICS: A Tutorial," *Proceeding of the 1988 Winter Simulation Conference*, edited by M. Abrams, P. Haigh, and J. Comfort, Institute of Electrical and Electronic Engineers, San Diego, CA, 1988.
19. J.J. O'Reilly and W.R. Lilegdon, "SLAM II Tutorial," *Proceeding of the 1988 Winter Simulation Conference*, edited by M. Abrams, P. Haigh, and J. Comfort, Institute of Electrical and Electronic Engineers, San Diego, CA, 1988.

20. S. Adiga and R. Glassey, "Object-Oriented Simulation to Support Research in Manufacturing," *Technical Report ESRC 88-20*, University of California at Berkeley, CA, December 1988.

21. M. Hammer and D. McLeod, "Database Description With SDM: A Semantic Data Model," *ACM Transactions on Database Systems*, Vol. 6, No. 3, September 1981.

22. M.R. Stonebraker, "The Design of the POSTGRES Storage System," *Proceedings of the 13th VLDB Conference*, Brighton, UK, 1987.

23. P. Lyngbaek and W. Kent, "A Data Modeling Methodology for the Design and Implementation of Information Systems," *Proceedings of 1986 International Workshop on Object-Oriented Database Systems*, Pacific Grove, CA, September 1986.

24. S. Jayaraman, "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part I: Basic Concepts and Methodology Selection" *Textile Research Journal*, Vol. 60, No. 6, 1990, pp. 247-54.

25. R. Malhotra and S. Jayaraman, "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part II: The Function Model," *Textile Research Journal*, Vol. 60, No. 6, 1990, pp. 351-359

26. S. Jayaraman and R. Malhotra, "Design and development of a Generic Architecture for Apparel Manufacturing," Volume I: The Function and Dynamics Models, Volume II: The Information Model, *Textile Report SJ-TR-ARCH-9104*, Georgia Institute of Technology, Atlanta, GA, April 1991.

Author Biographies

Rajeev Malhotra received his Bachelor of Technology degree in Textile Engineering from Indian Institute of Technology, New Delhi in 1983. He received his MS and PhD degrees from Georgia Institute of Technology, Atlanta, GA, in 1986 and 1992, respectively. Prior to pursuing graduate studies, he worked as a process control engineer for a major synthetic fiber manufacturer in India. He is currently employed with United Parcel Service as a systems design engineer. His research interests include methodologies for modeling large human-machine systems and their application in analysis and design of enterprise-wide integrated information systems.

Sundaresan Jayaraman, Associate Professor at the Georgia Institute of Technology in Atlanta, GA, received his B.Tech and M.Tech degrees in textile technology from the University of Madras, India and the PhD degree from North Carolina State University, Raleigh, NC. He was involved in the design and development of TK!Solver, the first equation solving program from Software Arts, Inc., and the predecessor of TK Solver Plus. Prior to joining Georgia Tech, Jayaraman worked as a product manager at Lotus Development Corp., Cambridge, MA and Software Arts, Inc., Wellesley, MA.

Dr. Jayaraman's current research interests are in the areas of computer-integrated manufacturing, enterprise modeling methodologies, knowledge-based systems development, and engineering design of textile structures including three-dimensional woven composites. He is a recipient of the 1989 Presidential Young Investigator Award from the National Science Foundation for his research in the area of computer-aided manufacturing.

On Precedence Relationships in an Aggregation Hierarchy of Activities

(To be presented in the *AAAI-93 Workshop on Modeling in the Large*, Washington, D.C., July, 1993)

K. Srinivasan and Sundaresan Jayaraman[1]
Georgia Institute of Technology
School of Textile & Fiber Engineering
Atlanta, GA 30332-0295

Abstract

Explicit representation of precedence relationships between activities is required for management decision making and is useful for analyzing information flow between activities. Precedence relationships cannot be propagated top-down in an aggregation hierarchy of activities and its bottom-up derivation is a complex task. Hence it cannot be specified *a priori* and requires computer assistance in derivation for any enterprise model. A procedure is proposed for deriving precedence relationships from the representation of flows between activities in Enterprise Modeling Framework (EMF). It is shown how the representation framework for entities and activities in EMF makes such derivation efficient. Some modifications to the precedence graph generated using this procedure are suggested before network analysis methods can be applied to it.

1. Introduction

Most function modeling methodologies such as the $IDEF_0$ [ICAM 81], CIM-OSA [Jorysz 1990] and the S-F-C paradigm [Caselli 92] use a hierarchical decomposition (i.e., an aggregation hierarchy) of activities. Hierarchical decomposition has the advantage of allowing one to study a part of the enterprise being modeled without losing the overall context. However, it does not explicitly denote the sequence or the precedence relationship between functions. While $IDEF_0$ and CIM-OSA do not address this issue, the S-F-C paradigm assumes a left-to-right execution ordering of activities on a diagram. This assumption is not always valid and too simplistic to allow for cycles in the precedence graph. Representation of the sequence between activities is required for making several management decisions such as scheduling and hence lack of it is a serious shortcoming in an activity model.

The precedence relations are dictated by two types of constraints: (i) the constraints imposed by the flow of materials and information between activities; and (ii) the constraints imposed by resource availability. The impact of resource constraints will be evident only in an instantiated, enterprise-specific model and is best studied by simulation. Enterprise Modeling Framework (EMF) [Srinivasan 92] uses the materials and information flow constraints to generate an explicit representation of the precedence relationship between functions and augments the activ-

1. To whom correspondence should be addressed.

ity model with this information.

Section 2 of this paper provides an overview of EMF. The main focus is on showing how activities and the entities flowing between them are represented. The procedure proposed for deriving the precedence relationship between activities is described in Section 3. It is shown that *a priori* specification of precedence relationships is not practical and the task is too complex to be performed manually. It also explains how the proposed procedure exploits redundancy in information about flow of entities between activities in EMF to efficiently derive precedence relationships. Section 4 suggests some methods for breaking cycles in the precedence graph to make it suitable for applying techniques such as critical path method.

2. Enterprise Modeling Framework: An Overview

EMF is proposed as an object-oriented framework for developing integrated models of the three major facets of an enterprise, viz., function, information and dynamics [Srinivasan 91]. It comprises an Entity Model based on a Semantic (Object-Oriented) Data Model, an Activity Model with both specialization and aggregation hierarchies, and a Knowledge Model to represent expert knowledge and heuristics about the domain. Expert knowledge can be represented as separate knowledge bases (e.g., using rules) or can be represented as an integral part of entity and activity models (e.g., as assertions on object attributes or as methods attached to different entities and activities).

Figure 1 shows the basic framework for defining an Entity and an Activity in EMF.

Class Entity		**Class Activity**	
Slots		**Slots**	
Index	: <An alphanumeric string>	***Index***	: <An alphanumeric string>
Name	: <An alphanumeric string>	***Name***	: <An alphanumeric string>
Generic Entities	: <A list of Entities>	***Generic Activities***	: <A list of Activities>
Classification Basis	: <An alphanumeric string>	***Parent***	: <A list of Activities>
Input to	: <A list of Activities>	***Children***	: <A list of Activities>
Control to	: <A list of Activities>	***Inputs***	: <A list of Entities>
Output from	: <A list of Activities>	***Controls***	: <A list of Entities>
Mechanism to	: <A list of Activities>	***Outputs***	: <A list of Entities>
Documentation	: <Description of the Entity>	***Mechanisms***	: <A list of Entities>
		Documentation	: <Description of the Activity>
		Preceded by	:<A list of Activities>

Figure 1. EMF Entity and Activity Class Definitions

The following features of this framework are relevant to the discussions in the subsequent sections:

• Information about flow between activities (as Inputs, Controls, Outputs and Mechanisms, collectively called as ICOMs) is represented redundantly in both the Entity and Activity class definitions. As can be seen in Section 3, this redundancy simplifies derivation of precedence relationships. The user does not have to pay the usual penalty for redundancy -- consistency maintenance -- as EMF takes care of it for the user.

• A class of entities may be further classified based on several criteria resulting in orthogonal sets of subclasses. For example, Customer Order can be classified as Stock Order and Special Order based on whether the order is for a regular product or a new product. On the other hand, Customer Order can be classified as Past-Due, Critical and Non-Critical Orders based on the deadline for fulfilling them. While the sets of instances of subclasses based on the same criterion will be mutually exclusive, nothing can be said about subclasses based on different criteria. As will be explained in Section 3, it is necessary to find if two sibling classes belong to the same set (i.e., have the same basis for classification) or if they belong to orthogonal sets (i.e., have different bases for classification) to derive the sequence between activities in which they are involved. The slot *Classification Basis*[1] helps in such identification: classes belonging to the same set will have the same value in this slot.

• The slot *Preceded by* of an Activity class contains the list of all the activities to be carried out before the current activity. In EMF, value for this slot is automatically derived by the system from other information contained in the Entity and Activity class slots.

3. Derivation of Precedence Graph

In this section, we will consider the features which make derivation of precedence relationships a complex task. Then we will look at a procedure for deriving the relationships. We will use the activities shown on Figures 2 and 3a-c for illustration.

Figure 2 shows the context of three activities using a pseudo-$IDEF_0$ notation: Accept Customer Orders, Prepare Production Orders and Follow up on Customer Orders; Figures 3a-c show decomposition of the three activities into their sub-activities[2].

A first cut definition of precedence relationship is as follows: <u>all the activities that generate an entity which forms either an input or control to another activity have to precede that activity</u>. It can be expressed in FOPL as follows:

$$\forall(x, y, z)\,[\text{ACTIVITY}(x) \wedge \text{ENTITY}(y) \wedge \text{ACTIVITY}(z) \wedge (\text{input}(y, A) \vee \text{control}(y, A)) \wedge (\text{output-from}(y, x) \wedge \neg \exists z[\text{consists-of}(x, z)]) \Rightarrow \text{preceded by}(A, x)]$$

1. Most object-oriented programming languages do not provide elegant means to group sibling classes. Providing such a facility may be an interesting problem for language designers to consider.

2. For brevity, only information relevant to the current discussion is included in the figures; having only 2 boxes in Figure 3a is a violation of the basic 3-6 boxes rule in $IDEF_0$.

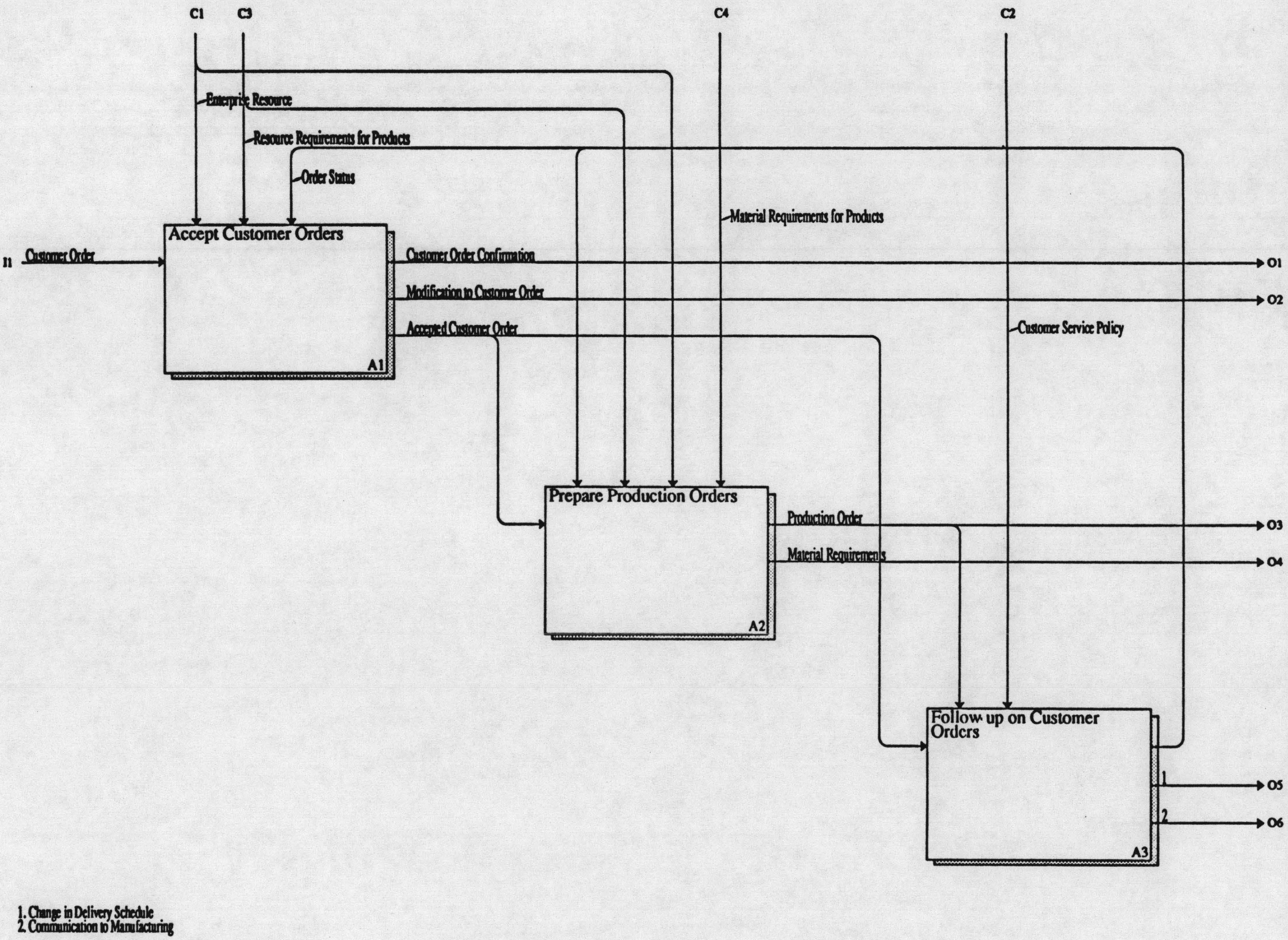


Figure 2. Details of the Activity Process Customer Orders

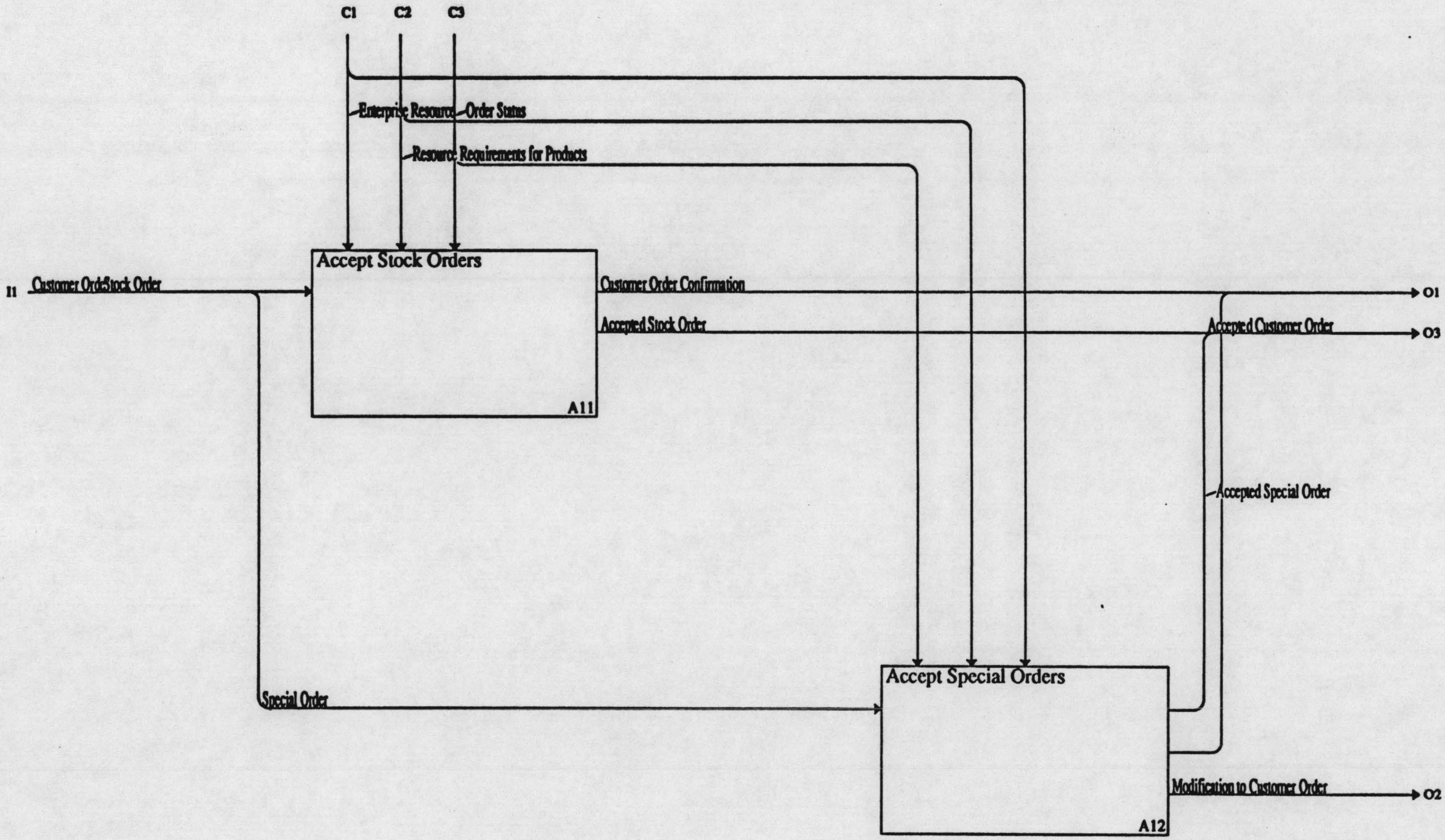


Figure 3a. Details of the Activity Accept Customer Orders

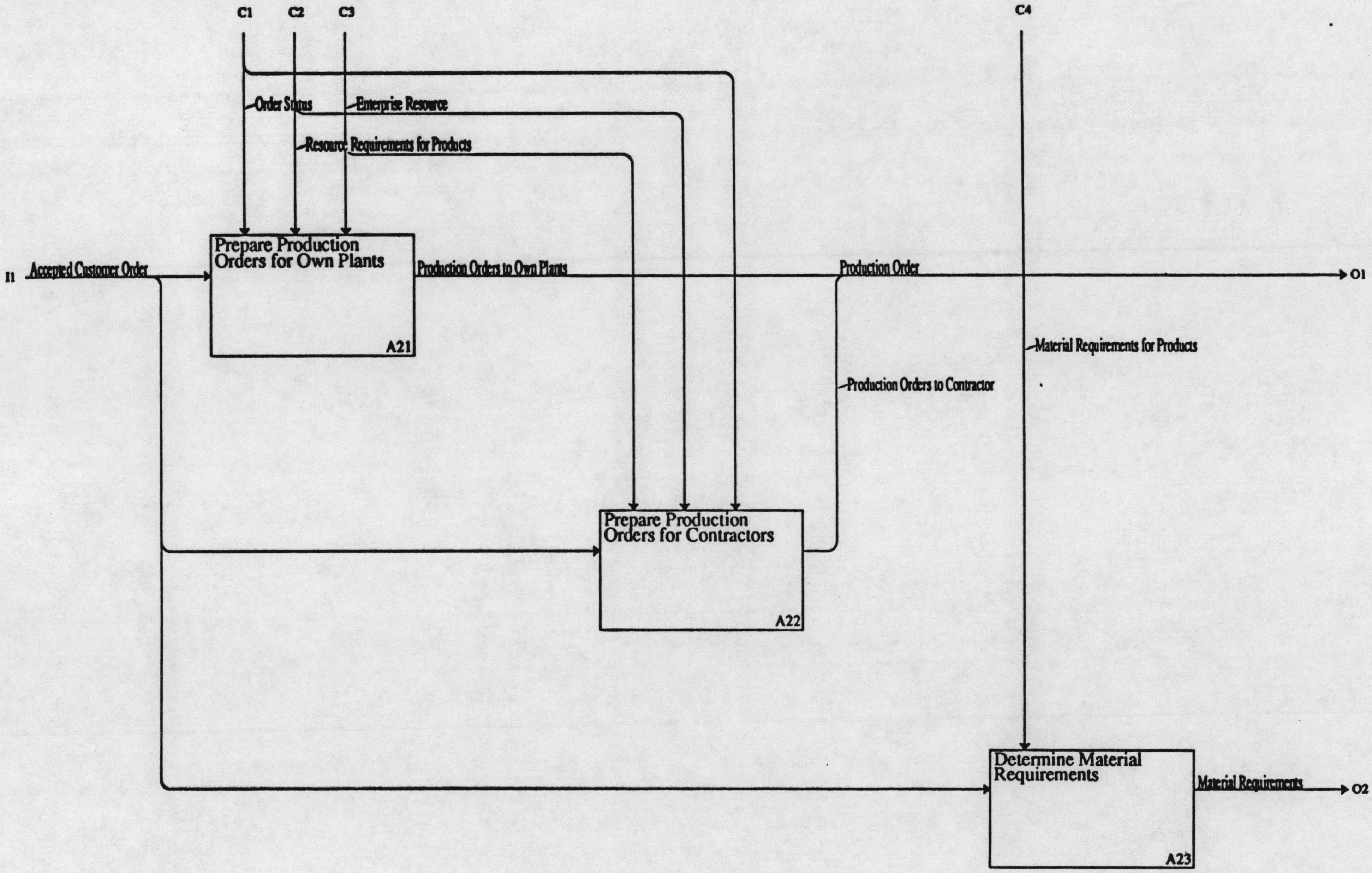


Figure 3b. Details of the Activity Prepare Production Orders

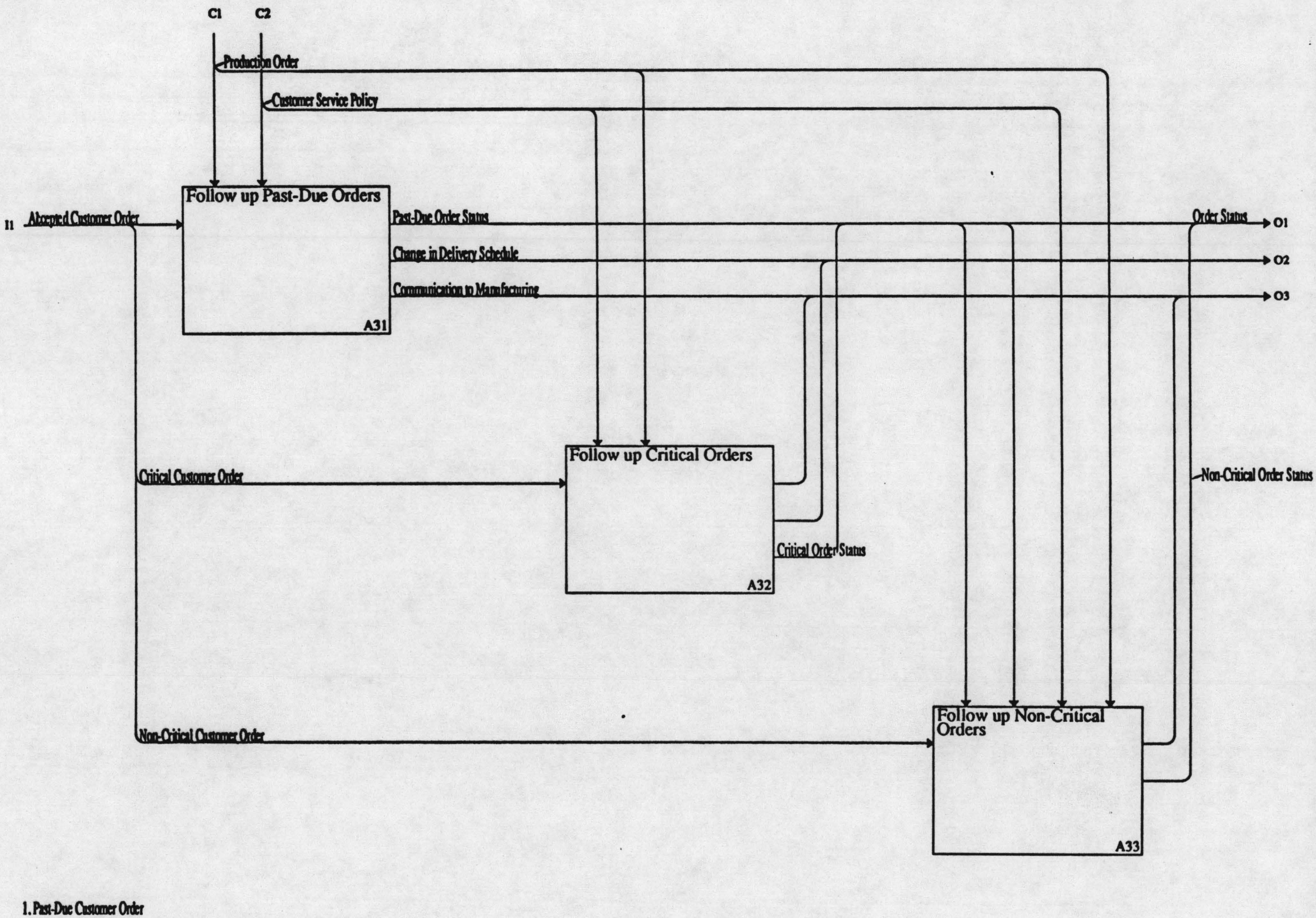


Figure 3c. Details of the Activity Follow up on Customer Orders

Where A is the activity for which the preceding activities are to be determined.

There are two features that make deriving precedence relations between activities using this definition directly difficult:

• *Precedence relationships cannot be propagated top-down.* An activity X precedes Y only implies that there is at least one sub-activity of X that precedes Y. For example, Prepare Production Order needs Order Status as one of its controls and hence should be preceded by Follow up on Customer Orders which generates Order Status (Figure 2); however, one of Prepare Production Order's children, *viz.*, Determine Material Requirements, can be carried out independently of any activity under Follow up on Customer Orders (Figure 3b).

• *Entities merge (into more generic or composite classes) and split (into more specific and component classes) as they flow between functions.* This makes tracing the origin of entities, particularly across levels in the aggregation hierarchy in the activity model, complex. There are three different types of junctions in entity flow between activities:

Type 1. The outputs of several activities merge into a generic class[1] before serving as an input or as a control to another activity. For example, Accepted Stock Order (output of Accept Stock Orders) and Accepted Special Order (output of Accept Special Orders) merge into their generic class Accepted Customer Order (Figure 3a) and serve as input to Prepare Production Order and its sub-activities (Figure 3b).

Type 2. A generic entity splits into several specific classes and serve as inputs or controls to several activities. For example, Customer Order is split into Stock Order and Special Order and the specific entity classes serve as inputs to the activities Accept Stock Orders and Accept Special Orders, respectively (Figure 3a).

Type 3. The outputs of several activities are specific entity classes (classified on a particular basis) and merge into a generic class; they again split into specific classes (classified on a *different* basis) before serving as an input or as a control to other activities. For example, Accepted Stock Order (output of Accept Stock Orders) and Accepted Special Order (output of Accept Special Orders) are sub-classes of Accepted Customer Order, classified based on the regularity with which particular products are ordered (Figure 3a); after merging, they are re-split into Past-Due Customer Order, Critical Customer Order and Non-Critical Customer Order based on deadline and serve as inputs to the children of Follow up on Customer Orders (Figure 3c).

1. Only the specialization hierarchy of Entities is discussed here, although everything will be applicable to aggregation hierarchy of Entities also.

Description of the Procedure

The procedure for determining the precedence relationships is illustrated by finding the activities that precede Follow up Non-Critical Orders.

Step 1. *For each Input and Control to the current activity, find the lowest level activities that generate them* (usually, the precedence relationship is sought to be established only among the lowest level activities as only they are executed in a real world setting). All these activities should precede the currently considered activity. This step identifies the precedence relationships determined by the simplest form of entity flows -- no splitting or merging of entities and corresponds to the first cut definition proposed above. Follow up Past-Due Orders and Follow up Critical Orders are identified using this step (Figure 3c).

Step 2. *For each Input and Control find the lowest level activities which generate all their specific classes*. It can be stated in FOPL as follows:

$\forall$(u, v, x, y) [ENTITY (u) $\wedge$ ENTITY (v) $\wedge$ ACTIVITY (x) $\wedge$ ACTIVITY (y) $\wedge$
(input (u, A) $\vee$ control (u, A)) $\wedge$ generic-entity (u, v)
((output-from (v, x) $\wedge \neg \exists$y[consists-of (x, y)]) $\Rightarrow$ preceded-by (A, x)]

This step identifies the precedence relationships determined by Type 1 entity flows. In our example, Production Order is the only entity class with specific classes, *viz.*, Production Orders to Own Plants and Production Orders to Contractors. These are generated by Prepare Production Orders for Own Plants and Prepare Production Orders for Contractors (Figure 3b).

Step 3. *For each Input and Control, find the lowest level activities which generate their generic classes (except the most generic class in EMF, viz., Entity) of the Inputs and Control entities*[1]. This step identifies the precedence relationships determined by Type 2 entity flows. In the current case, Non-Critical Customer Order and its generic class Accepted Customer Order, Past-Due Order Status and Critical Order Status and their generic class Order Status are identified. Since neither of the two generic classes is generated by any lowest level activity, no precedence relationships are derived during this step.

Step 4. *For each Input and Control, find out all the sibling classes with a different basis of classification* (Sibling classes with the same basis of classification have mutually exclusive instance sets and hence need not be considered in determining the precedence relationship). It can be stated in FOPL as follows:

$\forall$(u, v, x, y) [ENTITY (u) $\wedge$ ENTITY (v) $\wedge$ ACTIVITY (x) $\wedge$ ACTIVITY (y) $\wedge$
(input (u, A) $\vee$ control (u, A)) $\wedge$ sibling-entity (u, v)
((output-from (v, x) $\wedge \neg \exists$v[classification-basis (u, w) $\wedge$ classification-basis (v, w)]
$\wedge \neg \exists$v[parent-of (BASIC-ENTITY, v)] $\wedge \neg \exists$y[consists-of (x, y)]) $\Rightarrow$ preceded-by (A, x)]

1. Since the FOPL representation of this step is very similar to that of Step 2, it is not shown.

Non-Critical Customer Order has the following sibling classes: Past-Due Customer Order, Critical Customer Order, Accepted Stock Order and Accepted Special Order (Figures 3a & 3c). Of these only the last two have a different basis of classification. The activities that generate them are Accept Stock Orders and Accept Special Orders (Figure 3a).

Step 5. Apply Step 4 recursively to the parent classes of Inputs and Controls, till the most generic class (except Entity, the base class in EMF) is reached. Steps 4 and 5 together identify the precedence relationships determined by Type 3 entity flows.

Step 5. To avoid clutter, replace sub-activities by the parent activity whenever possible, i.e., if all the sub-activities of a particular activity are in the *Preceded by* slot of an activity class, they can be replaced by the parent activity. In the current example, the activities Accept Stock Orders and Accept Special Orders can be replaced by Accept Customer Order (Figure 3a). However, when the precedence information needs to be used for decision making, the parent activity has to be replaced by its children.

4. Using the Precedence Relationships

The precedence relationships between activities can be used to analyze information flow between the activities, and to make management decisions employing network analysis. Contrary to the classical network models, the precedence graph derived by EMF will have the activities represented as nodes. However, algorithms for the analysis of such networks are available and widely used in the construction industry [Lawrence 77].

It can be seen from Figures 2 and 3a-c that cycles will be common in the precedence graph. To apply deterministic network models to such graphs, the cycles have to be broken where possible by further decomposition or aggregation of functions. Otherwise the cycles can be linearized by using stochastic models as shown in Figure 4.

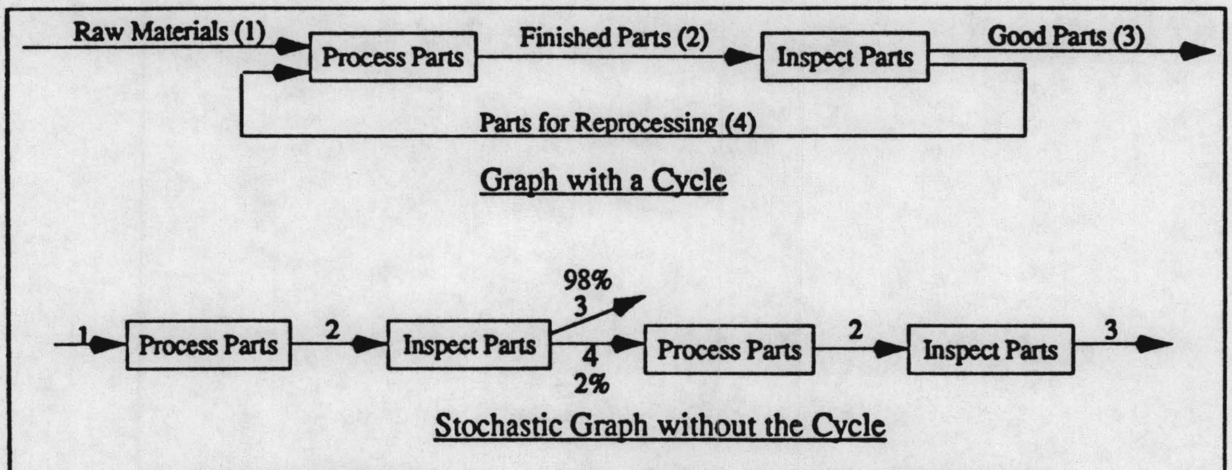


Figure 4. Breaking Cycle in the Precedence Relationship Graph

5. Conclusions

Aggregation hierarchy is the most common way of representing enterprise activities. The lack of precedence relationships between activities in such hierarchies is a serious shortcoming in them. A bottom-up procedure has been proposed for generating precedence graphs for an aggregation hierarchy of activities with the entities flowing between the activities represented as aggregation and specialization hierarchies. The procedure exploits the redundant representation of entity flows in both the entity and activity definitions. Some well-known methods have been suggested for utilizing the resulting relationship graphs in management decision making.

Acknowledgments

The authors wish to thank the National Science Foundation for funding this research in part through Presidential Young Investigator Research Award No. DDM-8957861.

References

Caselli, S., Papaconstantinou, C., Doty, K.L., and Navathe, S.B., "A Structure-Function-Control Paradigm for Knowledge-Based Modeling and Design of Manufacturing Workcells," *Journal of Intelligent Manufacturing*, 3, 1992, pp. 11-30.

Integrated Computer-Aided manufacturing (ICAM), *Function Modeling Manual* $IDEF_0$, Materials Laboratory, Air Force Wright Aeronautical Laboratories, AFSC, Wright-Patterson AFB, OH, 1981a.

Jorysz, H.R., and Vernadat, F.B., "CIM-OSA Part 1: Total Enterprise Modeling and Function View," *Int. J. Computer Integrated Manufacturing*, 3(3,4), 1990, pp. 144-156.

Lawrence, B.F., *Critical Path Precedence Networks: a Handbook on Activity-on-Node Networking for the Construction Industry,* Von Nostrand Reinhold Co., New York, 1977.

Srinivasan, K., and Jayaraman, S., "MEA: An Object-oriented Framework for Modeling A Manufacturing Enterprise," *Workshop on Object-Oriented Programming in AI, AAAI-91*, Anaheim, CA, July, 1991

Srinivasan, K., and Jayaraman, S., "Design and Development of an Enterprise Modeling Methodology," *Workshop on AI in Enterprise Integration, AAAI-92*, San Jose, CA, July, 1992.

E-27-651
#5,6

June 28, 1996

The Reports Coordinator
NSF-DDM
National Science Foundation
4201 Wilson Blvd.,
Arlington, VA 22203-9966

Dear Sir/Madam:

Enclosed please find the original and four copies of the Final Report for the Project entitled Presidential Young Investigator Award: Design and Development of a Manufacturing Enterprise Architecture (DDM-8957861). The original volume contains Part IV, Summary Data on Project Personnel.

If there are any questions on the report, I will be glad to respond. Thank you again for funding the effort.

Sincerely,

COPY

Sundaresan Jayaraman, PhD
Professor

Final Technical Report for Project Entitled

Presidential Young Investigator Award: Design and Development of a Manufacturing Enterprise Architecture

Submitted to

National Science Foundation
4201 Wilson Blvd.
Arlington, VA 22230

PYI Award Number: DDM-8957861

Dr. Sundaresan Jayaraman
Principal Investigator

Georgia Institute of Technology
School of Textile & Fiber Engineering
Atlanta, Georgia 30332-0295

E-mail: sundaresan.jayaraman@tfe.gatech.edu
Tel: 404/894-2490
Fax: 404/894-8780

Georgia Tech Project #: E-27-651

June 28, 1996

SJ-TR-NSFPYI-9606

OMB Number 345-0058

NATIONAL SCIENCE FOUNDATION
4201 Wilson Blvd.,
Arlington, VA 22230

BULK RATE
POSTAGE & FEES PAID
National Science Foundation
Permit No. G-69

PI/PD Name and Address

Sundaresan Jayaraman
Textile Engineering
GA Tech Res Corp - GIT
Atlanta GA 30332

NATIONAL SCIENCE FOUNDATION
FINAL PROJECT REPORT

PART I - PROJECT IDENTIFICATION INFORMATION

1. Program Official/Org.	Donald Gross - DMI
2. Program Name	OPERATIONS RESEARCH & PRODUCTION SYSTEMS
3. Award Dates (MM/YY)	From: 10/89 To: 03/96
4. Institution and Address	GA Tech Res Corp - GIT Administration Building Atlanta GA 30332
5. Award Number	8957861
6. Project Title	Presidential Young Investigator Award: Design and Development of a Manufacturing Enterprise Architecture

This Packet Contains
NSF Form 98A
And 1 Return Envelope

NSF Grant Conditions (Article 17, GC-1, and Article 9, FDP-11) require submission of a Final Project Report (NSF Form 98A) to the NSF program officer no later than 90 days alter the expiration of the award. Final Project Reports for expired awards must be received before new awards can be made (NSF Grants Policy Manual Section 677).

Below, or on a separate page attached to this form, provide a summary of the completed projects and technical information. Be sure to include your name and award number on each separate page. See below for more instructions.

PART II - SUMMARY OF COMPLETED PROJECT (for public use)

The summary (about 200 words) must be self-contained and intelligible to a scientifically literate reader. Without restating the project title, it should begin with a topic sentence stating the project's major thesis. The summary should include, if pertinent to the project being described, the following items:

- The primary objectives and scope of the project
- The techniques or approaches used only to the degree necessary for comprehension
- The findings and implications stated as concisely and informatively as possible

PLEASE SEE ATTACHED. THANKS.

PART III - TECHNICAL INFORMATION (for program management use)

List references to publications resulting from this award and briefly describe primary data, samples, physical collections, inventions, software, etc. created or gathered in the course of the research and, if appropriate, how they are being made available to the research community. Provide the NSF Invention Disclosure number for any invention.

PLEASE SEE ATTACHED. THANKS.

I certify to the best of my knowledge (1) the statements herein (excluding scientific hypotheses and scientific opinion) are true and complete, and (2) the text and graphics in this report as well as any accompanying publications or other documents, unless otherwise indicated, are the original work of the signatories or of individuals working under their supervision. I understand that willfully making a false statement or concealing a material fact in this report or any other communication submitted to NSF is a criminal offense (U.S. Code, Title 18, Section 1001).

	June 28, 1996
Principal Investigator/Project Director Signature	Date

IMPORTANT:
MAILING INSTRUCTIONS
Return this *entire* packet plus all attachments in the envelope attached to the back of this form. Please copy the information from Part I, Block I to the *Attention block* on the envelope.

Table of Contents

Part II Summary of Completed Project

An important prerequisite for the successful implementation of Computer-Integrated Manufacturing (CIM) in an enterprise is a detailed knowledge and understanding of the functions and information associated with the enterprise. Such a definition of the manufacturing enterprise is known as the *architecture* of manufacturing. A standard architecture would reduce the overall system complexity and enable users to build systems in increments. The Manufacturing Enterprise Architecture (MEA), developed in this research, is the framework that captures, represents and integrates the three major facets of an enterprise, viz., *function*, *information* and *dynamics*.

The overall objective of this research effort has been to design and develop MEA which will serve as a blueprint for the creation of a Computer-Integrated Enterprise (CIE). The research encompassed several complementary activities which led to the following significant accomplishments: creation of domain-specific models for yarn, fabric, apparel and carpet manufacturing enterprises; demonstration of the domain-independence of the manufacturing architecture through application of the architecture to healthcare delivery; design and development of a new methodology, viz., integrated framework for enterprise modeling methodology (IFEM); implementation of the IFEM methodology using object-oriented programming techniques resulting in the Enterprise Modeling Framework (EMF); design and development of specialty fabrics for ballet costumes using innovative CAD/CAM techniques for The Atlanta Ballet and the Centennial Cultural Olympiad; and transfer of technology and knowledge gained during the research to companies through graduates and collaborative case studies.

Part III Technical Information

1. INTRODUCTION

To be successful, competitive, and achieve excellence in today's global economy, a manufacturing enterprise must reengineer its operations and deploy the most advanced concepts and methods including Computer-Integrated Manufacturing (CIM). The scope of CIM transcends the traditional boundaries of the factory floor and encompasses the whole enterprise, giving rise to a Computer-Integrated Enterprise (CIE). A CIE can be defined as an enterprise that utilizes computers for the engineering, planning, manufacturing, marketing and business functions of the enterprise, and for the integration of all these functions into a cohesive enterprise system through a common information/knowledge base.

2. NEED FOR AN ENTERPRISE ARCHITECTURE

An important prerequisite for the successful realization of a CIE is a detailed knowledge and understanding of the functions and information associated with the enterprise. Such a definition of the manufacturing enterprise is known as the *architecture* of manufacturing. A standard architecture would reduce the overall system complexity and enable users to build systems in increments. The Manufacturing Enterprise Architecture (MEA), developed in this research, is the framework that captures, represents and integrates the three major facets of an enterprise, viz., *function*, *information* and *dynamics*.

3. RESEARCH OBJECTIVE AND SCOPE

The overall objective of this research effort has been to design and develop MEA which will serve as a blueprint for the creation of a CIE. The research has encompassed the following complementary activities aimed at realizing the overall objective:

- Design and development of domain-specific architectures;
- Illustration of the domain-independence of the manufacturing architecture through its application to *healthcare* systems;
- Development of enterprise modeling methodologies using major software engineering techniques and tools including object-oriented programming and databases;
- Investigation of the role of concurrent engineering in the textile/apparel complex;
- Investigation of methodologies for justification of investments in information technologies and systems;

- o Systematization of domain-specific knowledge and its harnessing for instructional purposes using multimedia technology; and

- o Design and development of specialty yarns and fabrics using computer-aided design and manufacturing techniques for ballet costumes in collaboration with The Atlanta Ballet.

4. SUMMARIES OF RESEARCH ACCOMPLISHMENTS

The major highlights of the various research activities carried out as part of the PYI Award are presented here; the references cited for the various activities provide details of the accomplishments.

4.1 Domain-Specific Architectures

The first step toward developing a *generic* manufacturing enterprise architecture (MEA) was to develop domain-*specific* architectures; these domain-specific models could then serve as the basis for the necessary generalization. Among the many sectors of the American industry facing intense foreign competition accompanied by the erosion of the manufacturing base and loss of employment opportunities is the textile/apparel sector. For this reason, the textile/apparel environment was chosen as the initial test bed for the development of the domain-specific manufacturing architectures.

A set of criteria was devised for evaluating and selecting modeling methodologies [5] for developing the architecture. Based on these criteria, several methodologies were evaluated; the IDEF methodology -- developed under the US Air Force's ICAM Program -- was selected. The details on the Yarn and Fabric Manufacturing Architectures can be found in [16]; the details of the Function model of the Carpet Manufacturing Architecture can be found in [3]. The Apparel Manufacturing Architecture (AMA) is discussed in [10, 13, 14]. The research on AMA received funding from the US Defense Logistics Agency under DLA-900-87-D-0018.

4.2 Domain-Independent Architecture

As the domain-specific architectures were being developed, the possibility of creating a domain- or industry-*independent* architecture was explored. The models were generic enough at higher levels (e.g., product development, production planning, and distribution) to represent other sectors of manufacturing besides textiles and apparel. At the lower levels, however, information specific to the domain needs to be modeled. The details of these concepts vis-a-vis the product continuum that ranges from commodity-type items to specialized items can be found in [4]. The role of knowledge and experience in operating an enterprise are also discussed in [4]. Several key issues germane to research in the area of integrated architectures for manufacturing were also identified.

4.3 Healthcare Delivery Systems Architecture

To further explore the concept of a domain-independent architecture and the application of the manufacturing architecture to non-manufacturing domains, research was carried out in collaboration with pediatricians in a group practice. The healthcare delivery system was viewed as a manufacturing enterprise and the research resulted in the healthcare delivery system architecture [2, 12, 27]. This effort conclusively demonstrated the concept of the domain-independent modeling methodology and architecture.

4.4 Enterprise Modeling Methodologies

During the course of this research, several major shortcomings in the IDEF methodology were identified and a new methodology termed IFEM (integrated framework for enterprise modeling) was proposed [14, 15]. The proposed schema can serve as the foundation for the development of manufacturing systems modeling software. Such an enterprise modeling methodology is essential in the context of implementing advanced concepts of Quick Response and Just-in-Time manufacturing in an enterprise.

MEA consists of three models, viz., *entity* model, *activity* model and *knowledge & beliefs* model to encompass the function, information and dynamics facets of an enterprise. A detailed discussion of the three models can be found in [28]. MEA overcomes the shortcomings of IDEF and other modeling methodologies and has the following salient features [29, 31, 34]:

- An Entity model based on a Semantic (Object-oriented) Data Model.
- An Activity model with both *IS-A* and *PART-OF* hierarchies of manufacturing functions.
- Seamless integration of the Entity and Activity models: the interface between activities is defined as Views on Entities; the editing and browsing tools for MEA have been designed for working concurrently on both the models; automatic consistency maintenance between the two models.

The *conceptual* schema proposed in [28] for MEA was implemented in software using CLOS, an object oriented programming (OOP) language [29, 30]. LispView was used to build the user interface conforming to OpenLook standards. Graphical tools for browsing through the Activity and Entity models were developed; the dynamics script has been integrated into the Activity model. The resulting Enterprise Modeling Framework (EMF) represents a significant contribution to the domain of enterprise modeling methodologies [17, 32, 33, 34, 35].

4.5 Concurrent Engineering in the Textile/Apparel Complex

The textile/apparel industry is probably one of the most dynamic manufacturing industries; this is because textiles and clothing are seasonal and the consumer is increasingly fashion, value and quality conscious. The discerning consumer is seeking unique styles and the ability to choose from a wide variety of fabrics (woven, knitted) made from a range of yarns (staple, filament), which in turn are made from an array of fibers (natural, man-made). Moreover, when the consumer doesn't find the specific item in the retail store, the consumer seeks alternatives and the potential sale may not materialize. Therefore, the ability to respond quickly to market trends is yet another important operating requirement for the textile-apparel complex [9]. This means the product and the associated manufacturing processes should be engineered to facilitate rapid production. The role of *concurrent engineering* in realizing these goals was explored [8].

4.6 Justification of Investments in Information Systems and Technologies

Information is the lifeblood of an enterprise, especially when a manufacturing enterprise needs to rapidly reconfigure itself -- change designs, materials, styles, production techniques, etc. -- in response to consumer demands and market trends [11]. The ability to successfully harness this valuable resource in a timely and well-coordinated fashion calls for investments in information systems and technologies. Investments in such technologies, however, cannot always be justified using traditional techniques such as net present value, return on investment and payback period. This is because such investments may provide competitive, strategic and tactical advantages that may not be as tangible as operational benefits (e.g., savings in personnel costs and improved operator productivity). The primary objective of this research effort was to explore the various issues related to the development of a methodology for justifying investments in information systems and technology.

The life-cycle view of an information system/technology (IS/IT) project was used to explore issues related to the development of a methodology for justifying investments in information systems and technology [1]. Preliminary findings indicated that there were no specific or well-defined methodologies used by organizations to justify investments in IS/IT and suggested the need for additional research in this area.

4.7 Systematization of Domain-specific Knowledge

The proliferation of powerful and inexpensive hardware/software systems has paved the way for innovative applications of information technology in the classroom. The first step towards building such multimedia-based intelligent tutoring systems is the *systematization* of domain knowledge and the development of a taxonomy for knowledge representation. Therefore, research was carried out to develop the knowledge representation schema for the domain of textile engineering; the schema was subsequently used to implement a tutoring system, TEESS, under MS-Windows using VisualBasic [25]. The system is expected to serve as a tool for imparting textile engineering knowledge to freshman students and new hires in the textile industry. Additional work is currently in progress to take advantage of the recent advancements in Web Browser technology and the World Wide Web.

4.8 Computer-Aided Design and Manufacturing of Specialty Textiles

The success of an enterprise depends, among other things, on its ability to effectively utilize advanced modeling methods and technologies in its operations, especially in the design and manufacturing facets. Therefore, to demonstrate the concepts of an integrated approach to the design and manufacturing of textiles, research was carried out on two major fronts: The first was aimed at modeling the structure-property relationships of yarns produced under various manufacturing conditions. The second was the design and development of a specialty fabric to meet the functional and aesthetic requirements of ballet costumes.

As part of the structure-property relationships research, the use of artificial neural networks for the prediction of yarn tensile properties was explored [26]; this effort turned out to be the first of its kind in textile research literature. The details of the studies and models can be found in [20, 21, 22, 23, 24].

The research on producing the specialty fabric was carried out in collaboration with The Atlanta Ballet [19]. Working with the ballet dancers, the characteristics (*functional* and *aesthetic*) required of the fabric were developed; these were then used to *engineer* the required yarns and fabrics using CAD/CAM systems. The fabric was used in the costume worn by the ballerinas during performances of the Atlanta Ballet and led one of them to remark "*this is the best costume I have ever worn in my career; I felt like I had nothing on me during the program ....*". Considering the fact that the human *skin* is the *ultimate garment* for the human body, this remark testifies to the research accomplishments and also demonstrates the true fusion of art and technology. The fabric has been continuously improved over the past three years and the most recent version will be featured as part of The Atlanta Ballet's performances during the upcoming Cultural Olympiad in Atlanta [18].

5. EDUCATION AND TECHNOLOGY TRANSFER

Several graduate students (4 PhD and 9 MS), 3 post-doctoral fellows, 2 research associates and 3 additional faculty members participated in the various research efforts during the course of the Award, and thus were able to pursue their research interests. In addition, the program afforded the PI the opportunity to write a textbook and Instructor's Manual for an introductory computing course for engineers [6, 7]. However, the Award funds were *not* used to support this activity.

The developed technologies have been transferred to the industry through two main channels: (i) collaborative efforts on case studies with industry; and (ii) the subsequent employment of graduates by major companies such as UPS (United Parcel Service), Intel, US Sprint and FedEx. The research results have also been transferred to the students in classroom settings in Senior and Graduate level courses at Georgia Tech, and at a *NATO Advanced Study Institute* (ASI) on Mechatronics held in Turkey.

In summary, during the course of the PYI Award, considerable progress was made towards realizing the complementary goals of:

- o advancing knowledge and the state-of-the-art in manufacturing systems;
- o transferring technology to the industry; and
- o educating the future generation of scientists and engineers.

Acknowledgements

Thanks are due the National Science Foundation for the PYI Award (DDM-8957861) which enabled the researchers to carry out various exciting projects summarized in this report. Thanks are also due Hewlett-Packard Company for providing the matching funds for the PYI Award through an equipment and software grant. An additional equipment grant from Sun Microsystems in support of the research is also thankfully acknowledged. The industry partners who served as test beds also deserve thanks and appreciation. Finally, the contributions of the PI's current and former graduate students, and colleagues towards accomplishing the research objectives are reflected in this report and various publications; these individuals also deserve sincere thanks and appreciation for their participation.

* * *

List of Publications

[1] Etheridge, S., Gilley, D., Narasimhan, L., Narasimhan, S., and Jayaraman, S., "Justification of Information Technology Projects", SJ-WP-IT-9209, Working Paper, Georgia Institute of Technology, Atlanta, Georgia 30332-0295.

[2] Govindaraj, T., and Jayaraman, S., "Healthcare Delivery Systems: An Enterprise Modeling & Simulation Perspective", AMIA 1994 Spring Congress, San Francisco, CA, May 4-7, 1994.

[3] Hamilton, G., "Design and Development of a Carpet Manufacturing Architecture: The Function Model", MS Thesis, Georgia Institute of Technology, Atlanta, Georgia, June 1996.

[4] Jayaraman, S., "On a Manufacturing Enterprise Architecture", IJCAI '89 (International Joint Conference on Artificial Intelligence) Workshop on Manufacturing, Detroit, MI, August 21-25, 1989.

[5] Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part I: Basic Concepts and Methodology Selection", Textile Research Journal, vol. 60, No. 5, pp. 248-254, 1990.

[6] Jayaraman, S., Computer-Aided Problem Solving for Scientists and Engineers, New York, NY: McGraw-Hill, Inc., February 1991, 432 pages.

[7] Jayaraman, S., Instructor's Manual to Accompany Computer-Aided Problem Solving for Scientists and Engineers, New York, NY: McGraw-Hill, Inc., September 1991, 210 pages.

[8] Jayaraman, S., "On Concurrent Engineering in the Textile/Apparel Complex", Proceedings of the 9th International Conference on Engineering Design, The Hague, The Netherlands, pp. 723-726, August 17-19, 1993.

[9] Jayaraman, S., "On Information, Technology, Competitiveness and the Integrated Textile Complex", The 1994 Fiber Society Technical Conference, Atlanta, Georgia, November 17-19, 1994.

[10] Jayaraman, S., and Cidambi, A., "Apparel Manufacturing Architecture [Version 1.5]", Volumes I, II and III, Technical Report SJ-TR-ARCH-9412, Georgia Institute of Technology, Atlanta, Georgia, December 1994.

[11] Jayaraman, S., "Computer-Aided Design and Manufacturing: A Textile-Apparel Perspective", in Advancements and Applications of Mechatronics Design in Textile Engineering, (Acar, M., Ed.), Kluwer Academic Publishers, The Netherlands, pp. 239-269, 1995.

[12] Jayaraman, S., and Ramesh, M.C., "Healthcare Delivery System Architecture: The As Is Function Model of a Pediatric Group Practice", Technical Report SJ-TR-NSFPYI-9501, Georgia Institute of Technology, Atlanta, Georgia, January 1995.

[13] Malhotra, R., and Jayaraman, S., "Design and Development of an Architecture for Computer-Integrated Manufacturing in the Apparel Industry, Part II: The Function Model", Textile Research Journal, vol. 60, no. 6, pp. 351-359, 1990.

[14] Malhotra, R., "An Architecture for an Apparel Manufacturing Enterprise", PhD Dissertation, Georgia Institute of Technology, Atlanta, Georgia, March 1991.

[15] Malhotra, R., and Jayaraman, S., "An Integrated Framework for Enterprise Modeling", Journal of Manufacturing Systems, vol. 11, no. 6, pp. 426-441, 1992.

[16] Narasimhan, B., "Design and Development of an Architecture for Information Management in a Manufacturing Enterprise", MS Thesis, Georgia Institute of Technology, Atlanta, Georgia, March 1994.

[17] Narasimhan, B., Navathe, S., and Jayaraman, S., "On Mapping ER and Relational Models Into OO Schemas", Lecture Notes in Computer Science #823 Entity-Relationship Approach - ER'93, (eds: Elmasri, R.A., Kouramajian, V., and Thalheim, B.,), Springer-Verlag, Berlin, Germany, pp. 402-413, 1994.

[18] Novina, T., "Textiles in the Olympics", America's Textiles International, pp. 62-66, June 1996.

[19] Park, S., "Studies on Effect of Fiber Properties and Fabric Structure on Fabric Hand", MS Thesis, Georgia Institute of Technology, Atlanta, Georgia, March 1995.

[20] Rajamanickam, R., "Studies on Fiber-Process-Structure Property Relationships in Air-jet Spun Yarns", PhD Dissertation, Georgia Institute of Technology, Atlanta, Georgia, December 1995.

[21] Rajamanickam, R., and Jayaraman, S., "A Computer Simulation Approach for Engineering Air-Jet Spun Yarns", to appear in Textile Research Journal.

[22] Rajamanickam, R., and Jayaraman, S., "An Analysis of the Modeling Methodologies for Predicting the Strength of Air-Jet Spun Yarns" to appear in Textile Research Journal.

[23] Rajamanickam, R., and Jayaraman, S., "Studies on Fiber-Process-Structure-Property Relationships in Air-Jet Spinning, Part I: Effect of Process and Material Parameters on the Structure of Microdenier Polyester/Cotton Blended Yarns", submitted to Journal of the Textile Institute, April 1996.

[24] Rajamanickam, R., and Jayaraman, S., Studies on Fiber-Process-Structure-Property Relationships in Air-Jet Spinning, Part II: Effect of Process and Material Parameters on the Properties of Microdenier Polyester/Cotton Blended Yarns, submitted to Journal of the Textile Institute, April 1996.

[25] Ramanan, P.S., "A Multimedia-based Learning Environment for Textile Engineering Education", MS Thesis, Georgia Institute of Technology, Atlanta, Georgia, June 1995.

[26] Ramesh, M.C., Rajamanickam, R., and Jayaraman, S., "The Prediction of Yarn Tensile Properties by Using Artificial Neural Networks", Journal of the Textile Institute, vol. 86, No. 3, pp. 459-469, 1995.

[27] Ramesh, M.C., Jayaraman, S., Hightower, P., Trawick, S.P., and Rosenfeld, J.M., Design and Development of An Architecture for Implementation of Medical Information Systems in Group Practices, Part 1: The As-Is Function Model, Technical Report SJ-TR-NSFPYI-9505, Georgia Institute of Technology, Atlanta, Georgia, May 1995.

[28] Srinivasan, K., and Jayaraman, S., "MEA: An Object-oriented Framework for Enterprise Modeling", Proceedings of the AAAI '91 Workshop on OOP in AI, Anaheim, CA, July, 1991.

[29] Srinivasan, K., and Jayaraman, S., "Design and Development of an Enterprise Modeling Methodology", Proceedings of the AAAI '92 Workshop on AI in Enterprise Integration, San Diego, CA, July, 1992.

[30] Srinivasan, K., and Jayaraman, S., "On Precedence Relationships in Enterprise Modeling Framework", Proceedings of the AAAI '93 Workshop on Modeling at Large, Washington, D.C., July, 1993.

[31] Srinivasan, K., and Jayaraman, S., "A Review of Enterprise Modeling Methodologies", Technical Report SJ-TR-NSFPYI-9303, Georgia Institute of Technology, Atlanta, Georgia, March 1993.

[32] Srinivasan, K., and Jayaraman, S., "Integration of Constraints in an Object-Oriented Enterprise Model at the Conceptual Level", Technical Report SJ-TR-NSFPYI-9307, Georgia Institute of Technology, Atlanta, Georgia, July 1993.

[33] Srinivasan, K., and Jayaraman, S., "EMF: A Modeling Environment for Hierarchical Specification and Simulation of Enterprise Dynamics", Technical Report SJ-TR-NSFPYI-9310, Georgia Institute of Technology, Atlanta, Georgia, October 1993.

[34] Srinivasan, K., "Design and Development of an Enterprise Modeling Framework", PhD Dissertation, Georgia Institute of Technology, School of Textile & Fiber Engineering, Atlanta, Georgia, December 1993.

[35] Srinivasan, K., and Jayaraman, S., "Comparison of Object-Oriented Programming Languages: The Enterprise Modeling Framework Perspective", Journal of Object-Oriented Programming, pp. 27-32, June, 1994.

Note: The regular reports in the Proceedings of the NSF DDM Grantees Annual Conference are not listed here.

* * *

E-27-651

PART IV -- FINAL PROJECT REPORT -- SUMMARY DATA ON PROJECT PERSONNEL

(To be submitted to cognizant Program Officer upon completion of project)

#6

The data requested below are important for the development of a statistical profile on the personnel supported by Federal grants. The information on this part is solicited in resonse to Public Law 99-383 and 42 USC 1885C. All information provided will be treated as confidential and will be safeguarded in accordance with the provisions of the Privacy Act of 1974. You should submit a single copy of this part with each final project report. However, submission of the requested information is not mandatory and is not a precondition of future award(s). Check the "Decline to Provide Information" box below if you do not wish to provide the nformation.

Please enter the numbers of individuals supported under this grant.
Do not enter information for individuals working less than 40 hours in any calendar year.

	Senior Staff		Post-Doctorals		Graduate Students		Under-Graduates		Other Participants[1]	
	Male	Fem.	Male	Fem.	Male	Fem.	Male	Fem.	Male	Fem.
A. Total, U.S. Citizens	1					3				
B. Total, Permanent Residents	3		2							
U.S. Citizens or Permanent Residents[2]:										
American Indian or Alaskan Native										
Asian........................										
Black, Not of Hispanic Origin.........										
Hispanic........................										
Pacific Islander..................										
White, Not of Hispanic Origin....... ...										
C. Total, Other Non-U.S. Citizens										
Specify Country 1. INDIA			1		7	1			1	
2. KOREA						2				1
3.										
D. Total, All participants (A + B + C)	4		3		7	6			1	1
Disabled[3]										

☐ Decline to Provide Information: Check box if you do not wish to provide this information (you are still required to return this page along with Parts I-III).

[1] Category includes, for example, college and precollege teachers, conference and workshop participants.

[2] Use the category that best describes the ethnic/racial status fo all U.S. Citizens and Non-citizens with Permanent Residency. *(If more than one category applies, use the one category that most closely reflects the person's recognition in the community.)*

[3] A person having a physical or mental impairment that substantially limits one or more major life activities; who has a record of such impairment; or who is regarded as having such impairment. *(Disabled individuals also should be counted under the appropriate ethnic/racial group unless they are classified as "Other Non-U.S. Citizens.")*

AMERICAN INDIAN OR ALASKAN NATIVE: A person having origins in any of the original peoples of North America and who maintains cultural identification through tribal affiliation or community recognition.

ASIAN: A person having origins in any of the original peoples of East Asia, Southeast Asia or the Indian subcontinent. This area includes, for example, China, India, Indonesia, Japan, Korea and Vietnam.

BLACK, NOT OF HISPANIC ORIGIN: A person having origins in any of the black racial groups of Africa.

HISPANIC: A person of Mexican, Puerto Rican, Cuban, Central or South American or other Spanish culture or origin, regardless of race.

PACIFIC ISLANDER: A person having origins in any of the original peoples of Hawaii; the U.S. Pacific territories of Guam, American Samoa, and the Northern Marinas; the U.S. Trust Territory of Palau; the islands of Micronesia and Melanesia; or the Philippines.

WHITE, NOT OF HISPANIC ORIGIN: A person having origins in any of the original peoples of Europe, North Africa, or the Middle East.