Adaptive Integrated Manufacturing Enterprises: Information Technology for the Next Decade

Cheng Hsu[1], *Lester Gerhardt*[2], *David Spooner*[3],
and Alan Rubenstein[4]

Rensselaer Polytechnic Institute
Troy, New York 12180-3590

April 1992

Revised: August 1992, November, 1992

Submitted to IEEE Transactions on Systems, Man, and Cybernetics.

1. Associate Professor, Decision Sciences and Engineering Systems.
2. Associate Dean of Engineering and Professor, Electrical, Computer, and Systems Engineering.
3. Associate Professor, Computer Science.
4. Director, Adaptive Integrated Manufacturing Enterprises Program, Center for Manufacturing Productivity and Technology Transfer.

Abstract

A new vision effecting adaptiveness in integrated manufacturing enterprises for the next decade is formulated. This vision has been developed on the basis of intensive research over the past nine years in Rensselaer's industry-sponsored Computer Integrated Manufacturing Program.

Built from existing results in both the scientific community and industry, the proposed research agenda calls for new fundamental information technology to enable Adaptive Integrated Manufacturing Enterprises (AIME). It focuses on four major problems:

(1) Management of multiple systems that operate concurrently over a widely distributed network without a central controller;

(2) Achievement of an open systems architecture that can accommodate legacy systems as well as add new systems;

(3) Exploitation of object-oriented technology in production systems with the crucial ability to manage heterogeneous views and propagate changes between views; and

(4) Modeling of enterprise information requirements for inspection and the utilization of inspection information to create a feedback loop from production to design.

These problems are analyzed and approaches to their solution developed in this paper. Together, they constitute a new research agenda developing solutions that include an AIME information architecture with attendant techniques (the first two problems), an extended object-oriented technology for an AIME software environment (the third problem), and modeling of AIME information requirements for in-process verification and inspection (the fourth problem). An extended effort on AIME based on this vision is currently underway that involves significant participation from industrial corporations.

Acknowledgement

This Research is supported in part by National Science Foundation Grant DDM 9015277 and Rensselaer's CIM and AIME Programs, both of which are sponsored by ALCOA, DIGITAL,GE, General Motors and IBM and conducted under the Center for Manufacturing Productivity and Technology Transfer (CMPTT).

1. The Objective: Adaptive Integrated Manufacturing Enterprises.

Is there life after CIM (Computer Integrated Manufacturing)? Or, asking the question from a user's perspective, is there a fundamental need for a new vision beyond CIM to further address the entrenched competitiveness problem facing manufacturing enterprises in today's global marketplace? The answer is a resounding yes, as evidenced by the emerging calls for action from both government and industry [24] with such labels as agile manufacturing and flexible CIM. The key reason is simple: Previous visions, efforts, and results of integration worldwide have yet to sufficiently consider the customer's evolving needs. Solutions emphasize the achievement of synergism across enterprise functions given requirements fixed at a particular point in time. These requirements, however, rarely stay unchanged for long. In fact, due to global competition and ever heightening customer demands, the basic competitive strategy of a corporation increasingly requires that the enterprise be able to respond rapidly to market conditions with a high-degree of product differentiation and value-based customer services. These requirements cause rapid changes in the enterprise which can no longer be dealt with by using managerial savvy alone. This trend is clearly established and in all likelihood will become more pronounced as we enter the next century.

Therefore, for improved productivity and quality, we submit that manufacturing requirements for the next century must focus on achieving **adaptiveness** to consolidate and extend the substantial gains made to date in integration. The result will be an enterprise with the ability to effect shorter cycle times, with lower volumes of identical items and higher mixtures of different items on the same assembly lines. It will also lead to greater flexibility in the organization of physical facilities that potentially are distributed globally, with increased customization, greater parallel activity across business functions and processes, and closer coupling with vendors and customers. This adaptiveness must be fully characterized with established scientific principles to clearly illuminate the technological gaps between the objective and the previous results; and, hence, will be more robust than a casual notion of agility or flexibility that may not lead to the understanding of the problem necessary for guiding the search for new fundamental solutions.

A scenario might best illustrate this vision: A customer calls the manufacturer to order a product that has a personalized logo or other custom features. Some of these custom requirements might be handled easily from a standard options package, some might require changes in standard processes for the product, and others might entail a revised design or even new materials. Thus, the cost and production time varies widely depending on the specific customization. The objective is not only to be able to offer this kind of customized product at marketable prices, but also to provide a firm price and delivery date to the customer on the spot while the customer is still on the telephone line completing the order.

Accommodating these requirements challenges the ingenuity of all members of the enterprise and demands innovative techniques for designing the product and the processes needed to build it, including verification of the product and global planning and control of the enterprise. Throughout this process, the entire product life cycle must be considered, including adaptability to change after initial manufacture and the ability for obsolete products to be recycled or safely and economically destroyed. But above all, a key ingredient of this new capability is information technology. The modeling, management, utilization, storage and processing of information must be adaptive as well as integrated to support parallel functions throughout the enterprise. A new research agenda for the field is in order.

To bring the problem into better focus, we refer to the specific objective of this research agenda as the study of new information technologies for Adaptive Integrated Manufacturing Enterprises (AIME). This paper formulates a particular model for AIME to provide an in-depth effort for the problem and serve as a reference point for the research field. The new effort is built from and embodies the current results on CIM and concurrent engineering (CE), especially those achieved in Rensselaer's industry-sponsored Computer Integrated Manufacturing Program. It focuses primarily on manufacturing information processes and their interactions with product data. As such, the information requirements of engineering design with respect to manufacturing are included in the research.

The problem of AIME is first analyzed in the next section, with a solution approach presented in Section 3. The research agenda is then fully developed and justified in the following three sections. The last section concludes the paper with a discussion of the AIME prototype. To provide an overview, the rest of the paper is illustrated in Figure 1, which maps the research problems to research areas of this agenda. Succinctly, there are four major problems of AIME (Section 2); the first two are addressed in research areas discussed in Section 4, and the other two are discussed in Sections 5 and 6, respectively. As a whole, these research areas are unified by the common solution approach (Section 3) and verified through prototyping (Section 7).

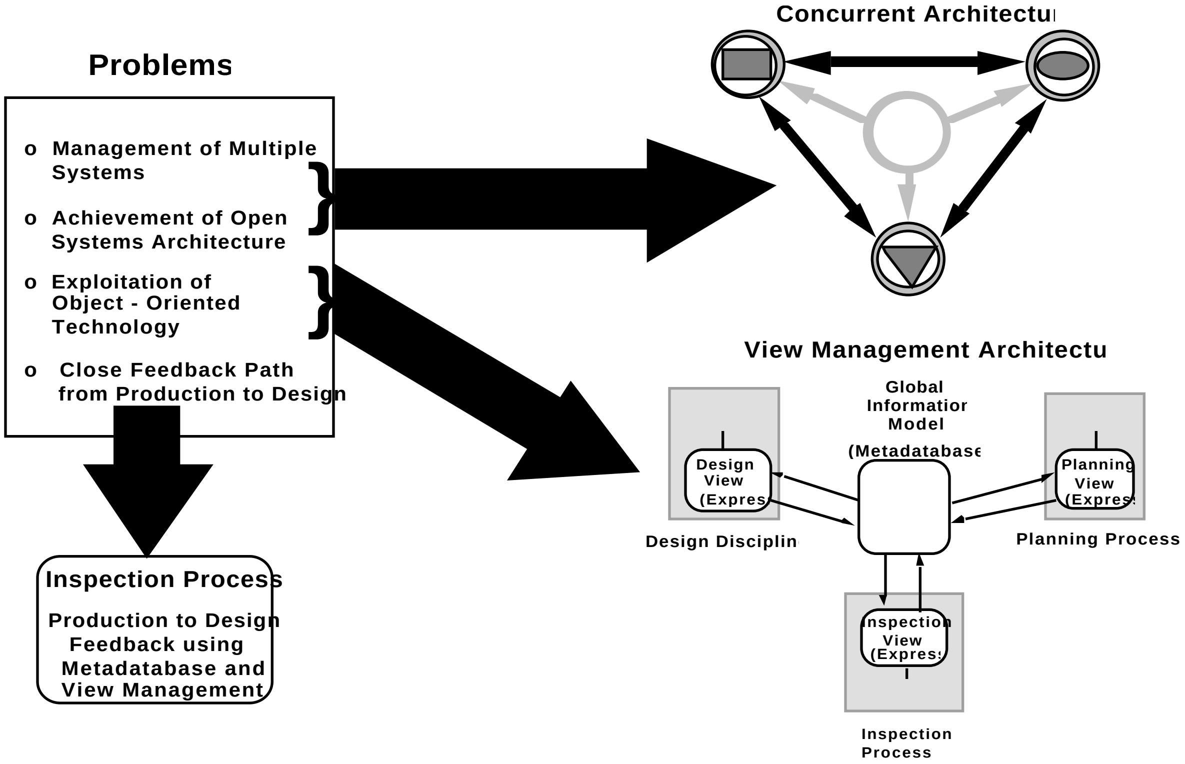

Figure 1: AIME Problems and Research Areas

We might emphasize that the problem analysis for AIME (section 2) in fact provides a working definition for the vision in such technical terms as the nature, principles and requirements of AIME. The recognition of such principles and requirements are contributions to the field. In a similar way, the specific solution approach contributes substantive concepts and methods towards accomplishing adaptiveness, thereby further solidifying the vision beyond CIM and CE. This solution approach itself, which recognizes, models, and solves AIME problems as a whole, is also a contribution. We might add that other approaches to AIME are, of course, possible. Other definitions and characterizations of the problem of AIME are reasonable also. However, the above contributions constitute a reference point serving to focus the alternative efforts in the field on the search for scientific results.

2. The Problem: Adaptiveness in Information Integration

Integrated manufacturing entails multiple information systems which have the following characteristics: (1) They are developed and maintained at local sites for particular applications and user groups according to local conditions; (2) They interact with each other in high-volume wide-area or even global networks; (3) They involve numerous different data resources (e.g., product design and manufacturing processes) and contextual knowledge (including operating rules and information flows); and (4) They or their contents are frequently revised as technology and needs evolve. These characteristics extend beyond the capabilities of available information technology. A great deal of research in CIM and CE, at many places worldwide, has been devoted to developing new integration technologies and a great deal of success has been accomplished. Prime examples include information architectures (e.g., CIMOSA [6]), software engineering capabilities (e.g., object technology -- see [28] for an example), and data exchange standards and techniques (e.g., PDES/STEP [40]). In addition, other disciplinary efforts such as in-process inspection and design for manufacturing (see e.g., a review in [50]) have resulted in better understanding of information requirements for enterprise information models.

Such progress notwithstanding, major problems remain to be solved. They range from specific technical issues such as managing updates and views in multidatabases and accommodating legacy systems and new systems in integrated environments, all the way to conceptual modeling of the requirements for real-time information flow between manufacturing and design (as opposed to only from design to manufacturing). At one level or another, these problems arise from the common limitation of past efforts that achieved integration against a fixed set of requirements without considering fully the adaptive nature of multiple manufacturing systems. The concept of AIME is established through both the vision of future competitive advantages and the need for solving present problems. This combined vision and need not only go beyond the notions of CIM and CE, but also require more substantive breakthroughs than the present visions of agile manufacturing and flexible CIM.

From the perspective of the proposed agenda, four specific problems facing AIME are considered below.

(1) Management of Multiple Systems

Prime examples of previous results on heterogeneous distributed databases include Multibase [30, 38, 46], DATAPLEX [3], IMDAS [29], Federated Databases [45], Interbase [33], among other similar efforts (see [36] for an introduction and [35] for a survey on query processing for these systems). Most of these efforts have not gone beyond laboratory research and none has completely resolved the distributed update problem. Moreover, each of these efforts employs as a cornerstone of its approach the traditional von Neumann model of synchronization which integrates schemata and serializes transactions across local systems under a central administrator. Consequently, there are fundamental limitations imposed by the architectures of these systems. These limitations place restrictions on local autonomy because of the requirements for schemata integration and standardization of system structures, global computing because of the complexity of serialization, and system evolution because of the need to re-compile or even re-design major elements of the global system when changes are made. Alternative approaches must be found to resolve these limitations, placing the solution to this problem at the heart of the adaptiveness issue. The specific initial approaches are discussed in Section 4.

(2) Achievement of an Open Systems Architecture

Another key challenge for adaptiveness is an open system architecture, which is a prevailing concern in major efforts such as CIMOSA and the industry-led Open Software Foundation. This capability is critical to a manufacturing enterprise's ability to respond rapidly to change as well as to incorporate heterogeneous systems. Despite the vast progress made in the past decade on standards, current technology still cannot support adaptiveness. These standards tend to emphasize standardization on designs and structures, rather than separating the underlying logic from systems. Thus, in a manner related to the issues in (1) above, current results do not accommodate legacy systems, revising or deleting existing systems, and adding new systems, without necessitating major re-design or re-compilation of existing systems. For example, all the systems mentioned in (1) above require major schemata restructuring at least at the global level to accommodate non-trivial changes. This often leads to unrealistic costs in practice. Standard or neutral structures alone cannot solve this problem. Techniques, such as those discussed in section 4, must be found to allow systems that do not use standards to work with systems that do and to make the global architecture and its administration independent of the local systems.

(3) Exploitation of Object-Oriented Technology in Production Systems

Object-oriented technology has been used successfully in design engineering, where product constructs are physically-based and fixed with unequivocal data semantics and object hierarchies. The technology also holds important promise to AIME since it possesses unprecedented capacity for software reusability, ease of maintenance, and user-friendliness. However, when employed for information integration, there are basic issues that must be addressed. One such issue is view management. Views, or customized and virtual interpretations of enterprise information resources, are significant in multidisciplinary design engineering. They are also critical to production systems, where the constructs involved tend to be logically-based and variable (e.g., master schedules, work orders, processes, and robotics programs). The difficulty with view management in object-oriented systems spans not only updates to views -- which are a problem even in traditional databases -- but also the creation of views due to the added complexities of the object paradigm [25, 28, 32]. To successfully implement this technology for production functions and reap its promised benefits for AIME, new view creation and management techniques are required for object-oriented systems. Section 5 discusses an approach to a solution. The mapping from a base object model to a view may be highly complex involving sophisticated algorithms to transform product and process data into a form required for a particular application. The logic of these mappings must be modeled in terms of a global interpretation of the interrelationships among the views to allow for effective management in AIME. View Management , based on object-oriented technology, is seen as an essential tool for functional concurrency which is a major requirement of adaptive manufacturing.

(4) Inspection-linked Enterprise Information Modeling and Integration

Integration entails a global view of the enterprise. Information integration with adaptiveness requires sharing, feedback, and flow of information between design engineering and production functions on a timely basis to support parallel operations across multiple systems. Therefore, a basic element of AIME is an enterprise information model that specifies these requirements and supports the utilization of these resources. Existing information models are intended for design (e.g., PDES/STEP) or production (e.g., industry-based CIM Models [6] and academic-based reference models [34]), but not for both. In a similar way, the application-oriented extensions to information requirements in CIM and CE tend to focus either on direct utilization of the design information within production (e.g., manufacturing and inspection processes development) or on general incorporation of production needs into design (e.g., design for manufacturability). What is missing is a cohesive model fusing production information requirements with those of design. This cohesive model must encompass both the modeling level (i.e., determining information requirements in product and process design mode) and the implementation level (i.e., utilizing real-time, on-line feedback of information). Albeit complex, a important step towards this end is in-process inspection, which entails the implicit inspection of a process using explicit measurement of products. As a user of data and knowledge resources, inspection, as an example of a manufacturing process, provides important information requirements for the enterprise model. More significantly, however, inspection exercises these models and closes the feedback loop from production to design. Thus, it is poised to demonstrate the new functionalities that AIME can offer. At present, inspection draws feature definitions and other product data from design systems, but does not usually tap into production planning and control knowledge or provide immediate feedback to product and process design. Effecting these new capabilities offers the opportunity for increased adaptiveness in both production and design, as well as for AIME as a whole.

We might add that standards are also a major problem facing AIME. However, there are currently many national and international efforts working on this problem, with promising results. Prime examples include the various data standards under development at ISO (International Standards Organization), protocols and models at NIST (National Institute of Standards and Technology), and software standards at OSF (Open Software Foundation). The solution of AIME will influence these efforts, but should not duplicate, nor supersede, them. Therefore, in this agenda, we shall not include new efforts in the area of standards; instead, we call for incorporating appropriate results from current efforts into the solution of the four problems above .

3. The Approach: A Core Technology for Adaptiveness

In this section, a specific solution approach is developed for the problems discussed above to achieve the vision of AIME. This solution approach synthesizes some proven results of CIM and CE research over the past several years on information technology and integration (i.e., the metadatabase model [22]), object-oriented databases (i.e., ROSE [11, 12]), and integrated inspection [7]. These results are unified by a common AIME model featuring a metadatabase-supported enterprise information system concept as shown in Figure 2. The approach uses a global information model that substantiates the metadatabase and an AIME vision interpretation prototype that serves as both a testbed for the research and a final empirical proof of the solution. Moreover, this prototype incorporates pertinent industrial standards into its design so as to ensure transferability to industry and other research centers.

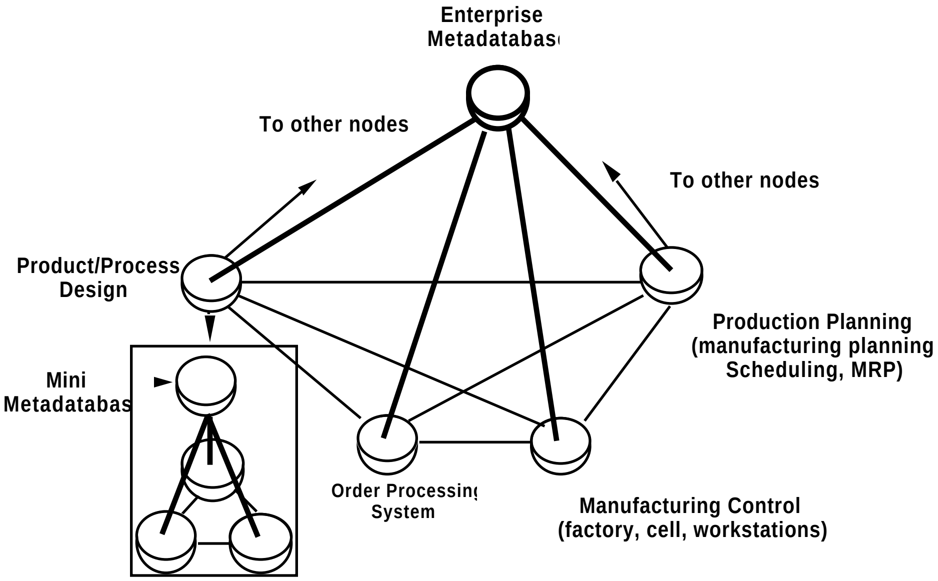

Figure 2: Metadatabase Supported Enterprise Information System

The metadatabase is an implementation of the global information model for the enterprise, containing both consolidated data resources (types, not instances) and their contextual knowledge (semantic rules, operating rules, and decision rules) from product models, production models and other pertinent data and knowledge resources. As such, the metadatabase represents the logic of the enterprise as a whole. While corresponding to the logic of individual multiple systems, the metadatabase does not supersede any schemata of local databases nor control their transactions at the instance level. The metadatabase manages only enterprise metadata, which effects synergy among concurrent but independent local operations (see the next sections for details). Therefore, the metadatabase constitutes a conceptual schema for the enterprise in the terminology of the classical three-schema architecture of databases, separating global logic from its local implementations. Its non-controlling nature so that users can utilize the global model without going through a special *metadatabasenode* distinguishes it from all other integration models. In Figure 2, the metadata flows are represented in bold lines and data instance flows are shown with thin lines. Also depicted in the figure is the possible distribution of the metadatabase itself. A distributed metadatabase can be managed in the traditional way, since the volume of transactions and the complexity of metadata are orders of magnitude less than for data instances.

Metadata will be utilized to facilitate both view management of object-oriented technology and in-process inspection and integration. Through this solution approach, basic principles for AIME information technology are formulated. These principles characterize the new functionality regardless of the specific techniques chosen to implement the vision. These principles include:

- Use of an information architecture that is based on metadatabases, repositories, or the equivalent;
- Use of an information model that unifies both production and product functions with parallel processes and closed-loop information flows; and
- Concurrent information processing in a distributed environment supporting global views, global retrievals and global changes to both information and the environment.

The technical details and research tasks are provided in the next sections.

4. AIME Information Architecture and Attendant Techniques

An information architecture is needed to implement the metadatabase-supported enterprise information system concept for the AIME model. A viable generic architecture can be built from the proven metadatabase model for CIM information integration mentioned above. This architecture, together with its attendant techniques, addresses the first two problems as shown in Figure 1.

The metadatabase concept seeks explicitly to provide an alternative to both schemata integration and global serialization, two central elements found in virtually all attempts to manage multiple systems [29, 33, 45, 46, 48]. As a result, it avoids the complexity inherent in these techniques. The enterprise metadatabase is used to assist the end user (or programs) in performing information resources management and in developing queries without the need to understand the full technical details of the hierarchy of integrated schemata. It distributes the contextual knowledge to empower the local systems to update data and communicate with each other without central database control. It incorporates legacy models and new or changed local models into its generic structure to support evolution without system redesign or recompilation.

Several new methodologies have been developed to construct the metadatabase environment. These include a definitive model for the metadatabase and its central role in enterprise information integration, a metadata representation method, and a combined data and knowledge processing methodology. Additional methods include a model-assisted global query facility, a rule-oriented programming environment (ROPE), a concurrent architecture using distributed rule-based shells, and a metadatabase management system.

The metadatabase model contributes two key results to achieving adaptiveness; namely, the **metadata independence structure** and the **concurrent architecture.** Specifically, the concept of metadata independence demands the use of a global model in place of a hierarchy of integrated schemata (or data dictionaries). This global model unifies all classes of metadata into a generic metadatabase structure that remains unchanged as local models change or are deleted or added to the system. It decouples global logic from the local system structures. It represents global metadata as instances of a metadatabase structure and processes the global metadata using metadatabase transactions (i.e., add, delete, update, retrieve) just like ordinary database applications. It incorporates changes to local models and new models as updates to metadata instances that populate the metadatabase structure. Metadata representation and management methods and model-mapping methods are key to providing this functionality.

The concurrent architecture connects local systems with a network of non-interfering, rule-based local shells as illustrated in Figure 3. These shells implement the control knowledge contained in the metadatabase in a localized, distributed manner. The concurrent architecture uses the metadatabase to create, maintain, and evolve these shells, automatically. Rule-base modeling and management methods and software engineering methods based on the ROPE concept are used.

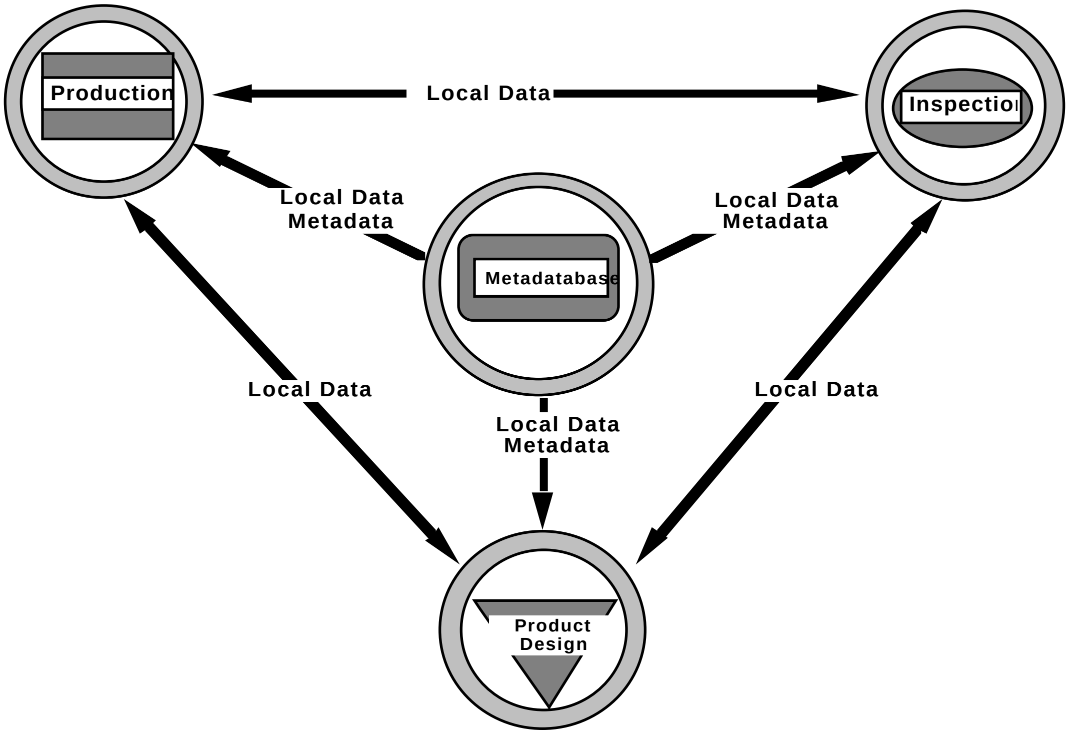

Figure 3: Concurrent Architecture using a Metadatabase

These two key elements promise to make it possible to achieve a level of adaptiveness substantially beyond that which has been possible to date. They give rise to a basic information architecture that effects the four fundamental capabilities necessary to manage multiple systems: (1) local autonomy, (2) open systems architecture accommodating new and old systems, (3) systems evolution, and (4) concurrent processing at all nodes of both local applications and global information flows.

Management of Multiple Systems

The concurrent architecture provides a solution to the problem of managing multiple systems that is not serialization-based, but still allows concurrent operations and achieves event-level consistency. Furthermore, it provides event-based control and additional functionalities for managing multiple systems beyond the traditional scope of data administration. In essence, the contextual knowledge that is contained in the metadatabase is localized and distributed, and is implemented in local rule-based shells. These shells empower the local systems to operate independently and concurrently, executing such tasks as propagating changes in persistent data resources and transmitting event-triggered information flows. The global synergism is achieved by virtue of the modeling and design of this contextual knowledge and its distribution into local shells, as opposed to utilizing centralized run-time control. However, to allow for changes to the contextual knowledge and other forms of system evolution, the metadatabase provides maintenance and synchronization of these rule-based shells. This way, the conventional requirement of a global controller for data instances is replaced by one for metadata, which is orders of magnitude simpler. In addition, because of the rule-base nature of these shells, the metadatabase can work with a traditional serialization-based controller to centrally process select classes of data instances, if this is desirable. These classes will themselves be modeled and implemented as contextual knowledge. Additional efforts are underway in the following areas of implementation:

- A distributed rule-base manager capable of deriving pertinent rules (in a neutral structure) from the global information model as it is created or evolved. This manager is also responsible for distributing the global rules into local rule-base models as well as propagating changes made to rules and executing the global rule-base contained in the metadatabase for metadata applications.

- An extended rule-oriented programming environment (ROPE) methodology to create the necessary shells in their respective software environments. This methodology generalizes the current results in the metadatabase model into an automated software engineering paradigm and is used to implement the local rule-bases created by the manager above, to maintain those shells (including inter-shell message flows and changes to rules), and to enable interfaces between the shells and their respective systems.

- An investigation of the comparative performance of alternative models for concurrent updates to facilitate system design tasks and application of this architecture.

Achievement of an Open Systems Architecture

A key task for achievement of an open systems architecture is to develop a metamodel using a metadatabase modeling system (i.e., the TSER method [17, 18]) through creation of mapping algorithms linking the system to representative models in the field (e.g., paradigm translation from TSER to EXPRESS or Relational.) With the metamodel in place, the promised openness of the concurrent architecture using a metadatabase will be achieved. Therefore, the research will extend the current metadatabase model to provide direct connections with relational (SQL) databases, systems supporting PDES/STEP (through EXPRESS), and object models such as OMG, all of which are industrial-class standards. These connections include mapping algorithms for reverse engineering to convert system (local) metadata based on these standards into the metadatabase representation models. Consolidation algorithms will then be used to incorporate the resultant metadata into the metadatabase environment. Local transaction generators that allow global queries, updates, and other transactions to be performed on these new systems will be developed (utilizing the metamodel). To illustrate this added functionality, the prototype will connect an industrial database that is based on these standards to the metadatabase system. The extended global environment that results will then operate as a whole. All of this will be accomplished with minimal effort from human experts and minimal interruption of operation at both the local and global levels. This goal serves a number of purposes simultaneously. It makes the metadatabase results usable to industry, it proves the concept of an open system architecture using a metadatabase, and it provides a model for the general effort of achieving strong open systems capabilities in the field.

5. Extended Object-Oriented Technology for AIME Applications

The third research problem (see Figure 1) is addressed in this research area. As indicated above, use of object-oriented technology provides many advantages for the definition and management of product and process models in design and manufacturing [42, 43]. Rich data structures encapsulated with semantic information in the form of constraints and operations provide the flexibility needed to accurately capture the complex interrelationships between data objects describing the structure of a product and the processes needed to manufacture it. Object technology will become even more important in the future as the emerging PDES/STEP standards [40] for exchange of product data become more widely used. EXPRESS, the data definition language of this standard, supports many of the features typically found in object technology. Major extensions, however, are needed to fully reap the benefits and promises of object-oriented technologies for AIME; namely, view capabilities. The tools to be developed for view management make it possible to simultaneously conduct concurrent activities in an adaptive environment.

View Management Models and Methods

Each engineering discipline participating in the design of a product has its own view of the product data needed to define that product. Similarly, each processing step in the manufacture of the product has its own view of the process data for the product. For example, the view for an inspection planning process might focus on geometric features, their dimensions, and their tolerances, while the view for a design analysis application might focus on geometry and material properties. These views optimize the organization and content of the product and process data from the perspective of the engineers working in each discipline or manufacturing process. In addition, these views optimize the product and process data models for the software tools used by these engineers.

Each view must be derived from a common global product and process model in order to guarantee consistency between the views and so that changes made to one view can be propagated to all other views to maintain consistency. In conjunction with the metadatabase, these models

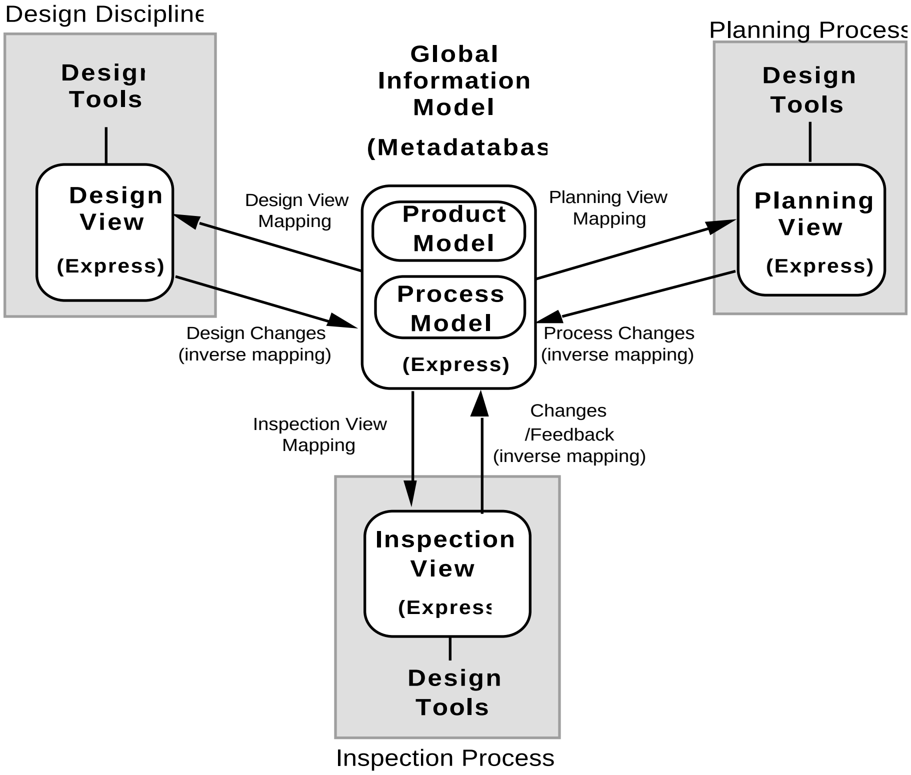

Figure 4: Architecture of a Manufacturing View Management System

form a Global Information Model for the manufacturing enterprise. Figure 4 illustrates this view architecture for three applications: design, process planning, and inspection planning. Changes made to the product and process data for a product by any one of the three applications may affect the views of the product and process data for the other two applications.

In traditional database systems, view management is a well understood problem [5]. This is not the case for object-oriented data models, however [14, 39]. The added flexibility and complexity of the object paradigm requires that new view creation and management techniques be developed. Creation of a view is no longer a simple task of selecting a subset of the rows and columns in one or more tables, but may require that networks of objects of different types be reorganized to form new networks with individual objects derived from the original objects and new links created between objects. The mapping from the base object model to a view is defined in terms of functions that construct the view. In design and manufacturing, these mapping functions may be highly complex involving sophisticated algorithms to transform geometry, topology, features, dimensions, tolerances, and other product and process data into the form required for a particular engineering discipline or manufacturing process.

Updates to views are a problem even in traditional database systems [26], and they are a more difficult problem in object-oriented systems because of the added complexity of the object paradigm. Yet, as discussed above, it is necessary for changes made to one view of a product or process model to be propagated to the global product and process models and from there to the other views currently in use. This will require careful study of the limitations of view updates in the object paradigm to prevent situations that could introduce errors due to inaccurate transmission of changes between views. Since manufacturing enterprises are often highly distributed, in many cases views must be materialized and transmitted to other sites in the manufacturing enterprise for processing. This makes propagation of changes between views (materialized views [37]) even more difficult.

One approach to addressing these problems is to base the product and process models of the global information model on the PDES/STEP standard for exchange of engineering product data [40]. The product and process models are defined using the EXPRESS language of this standard. When possible, the entities used in a product model to define a particular product will be based on the PDES/STEP standard and its associated application protocols. This approach exploits the expertise that went into defining flexible product models for the standard and simplifies integration of the system with future design tools based on the PDES/STEP standard. As discussed above, the EXPRESS language provides a flexible object-oriented data model for definition of the global product and process information.

The views of the global product and process models needed for the individual design disciplines and manufacturing processes can also be defined using EXPRESS. A suitable language must be defined to specify the mapping functions from the global models to the views. Preliminary work in this area suggests that a functional language similar to DAPLEX [38] may be appropriate for this. The mapping functions must have inverses to allow changes made to a view to propagate back to the global models. Once the global models, views, and mapping functions are defined, a **manufacturing view management system** is required that can capture the changes made to a view and manage the process of propagating these changes back to the global models.

Manufacturing View Management System

The product and process models in the global information model are extensions of the product database that is currently part of the CIM research testbed at Rensselaer [44]. This product database was developed using ROSE, a persistent object system for management of engineering data that is also under development at Rensselaer [11, 12]. The ROSE system is designed to support CE applications. It is being designed and implemented as part of the DARPA Initiative in Concurrent Engineering. It uses the EXPRESS language of the PDES/STEP standard as its data definition language and includes a tool to compile specifications of entities written in EXPRESS into C++ class definitions. These class definitions are then used to develop application programs in C++ taking full advantage of all features of the C++ programming language, including multiple inheritance. Any object created in one of these EXPRESS-generated classes is persistent. ROSE also supports multiple versions of product data and provides cooperative work tools for support of CE applications.

The manufacturing view management system for propagation of changes made to views can be designed and implemented as an extension of an object-oriented change processing system that is part of ROSE. This system captures primitive edits made to the attributes of an object and allows those edits to be applied to other versions of the same object. In this system, a change to version V_i of a product model is a list of edits to the objects in V_i, and is denoted:

$$\text{Change} = [e_1, e_2, \ldots, e_t]$$

where each edit, e_k, consists of an operation, op, applied to an object, o, and a list of parameters:

$$e_k = (\text{op}, \text{o}, p_1, p_2, \ldots)$$

Typical edit operations for a change processing system include: *create* an object, *delete* an object, *modify* the value of an instance variable in an object, *add* an object to an aggregate object, and *remove* an object from an aggregate object. Changes made to a version of a product model are recorded in a change file. For example, if engineer u makes k changes to version V_i of a product model, then the following change file is generated:

$$\text{ChangeFile}_i^u = [\text{Change}_1, \text{Change}_2, \ldots, \text{Change}_k]$$

A change file has several uses in a CE system, but one of the most important is that it provides a mechanism whereby engineers can exchange ideas about a product model. For more details on the change processing system, see [10].

To build the view management system, extensions will be necessary to the change processing system in ROSE to raise the semantic level at which updates are captured so that they can be propagated between views and the global models. It will also be necessary to handle the inverse mapping from a view to the global models when design changes are propagated.

Finally, this view management system must be joined with the metadatabase system at both the logical and physical levels. Logically, the global view model is a part of the enterprise information model that the metadatabase supports (see Figure 2). Physically, the open system architecture discussed above integrates both systems (see Figure 3).

6. Information Modeling and Integration Through Inspection

This segment of the research agenda serves as a focus for establishing and exercising a cohesive model that fuses production information requirements with those of design at both a modeling level and an implementation level, thereby addressing the fourth problem shown in Figure 1. As a user of data and knowledge resources, inspection both requires and provides important classes of information for the enterprise model. More significantly, inspection closes the feedback loop from production to design, and serves as an example to demonstrate critical new integration and adaptiveness functionalities that AIME can offer. Also, explicit inspection of a product at different stages of production permits implicit and continual evaluation of the overall process. Other processes in manufacturing (test plans, assembly plans, tool plans, process plans, etc.) can also benefit from this level of integration and adaptiveness.

Past inspection research [7, 8] has resulted in an integrated multimodal system with a product focus that concentrates on finding features, parameters and specific defects. It has been concerned with the theory and application of metrology. The system performs model and object based inspection, and minimizes the number of views needed to verify an unfixtured part by incorporating adaptive algorithms to assure the minimum number of measurement points and to optimize the tradeoff between speed and accuracy using different modalities, with an eye towards pre-, mid-, and post-process verification. On this basis, information requirements can be developed for in-process verification and design for verification/inspection. These information requirements provide a linkage for integration and adaptiveness with respect to the enterprise.

Overall, a hierarchical task decomposition technique is used which follows the architecture and philosophy of the basic metadatabase approach that permeates the underlying methodology.

Hierarchical Task Decomposition Techniques

The verification/inspection system consists of a series of high level goals to be satisfied. Goals are satisfied when specified information has been obtained regarding the part. Thus, the verification function can be represented as a decision hierarchy where each level provides logic for partially decomposing the goals into simpler goals. At the top level, the decision hierarchy starts with an abstract model of the part description. The abstract model can be a Computer-Aided Design (CAD) system model of a part. In order to simplify the complexities of the implementation, information about the external environment regarding probe selection, fixture set-up, etc., should also be available to the decision hierarchy at the top level.

As shown in Figure 5, the decision hierarchy can be implemented in 6 levels: Tolerances, Features, Surfaces, Points, Path and Interface (e.g. CMM) [16]. Using these different decision levels, the hierarchy can take a model from a CAD system or product database and directly convert it to the form needed for inspection/verification actions. This decision hierarchy entails a range of information that must be calibrated with the product database and ultimately the global information model.

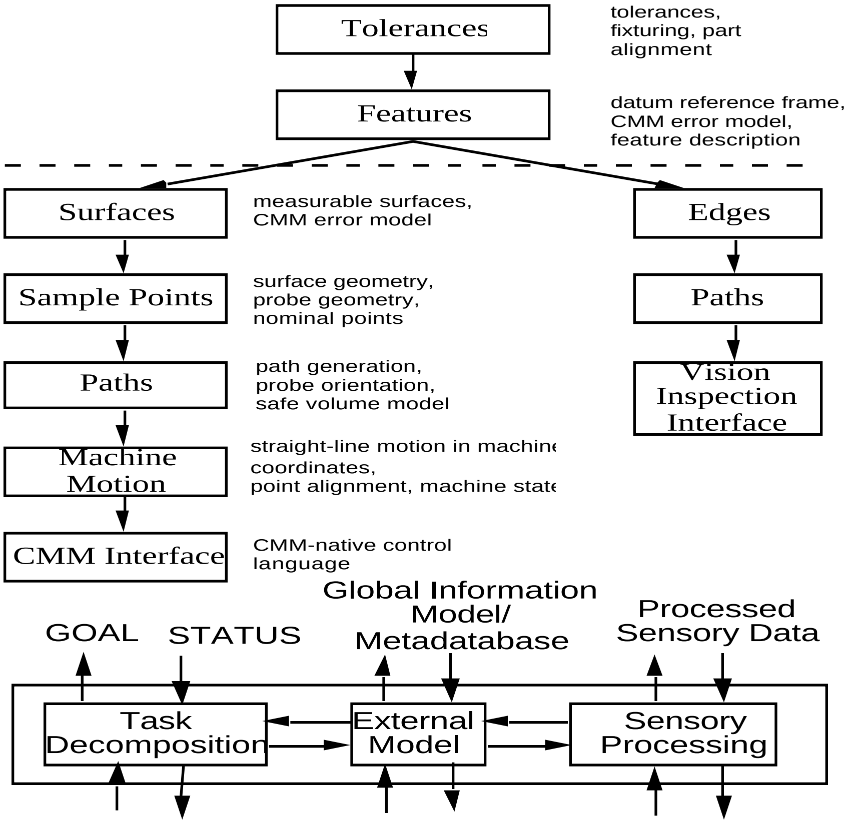

Figure 5: Hierarchical Task Decomposition for Tactile Inspection Information Requirements

Specifically, the **Tolerance** level converts the goal of verifying the part into a set of subgoals for verifying selected tolerances. Considerations to be made at this point are whether the selected tolerances can be measured in the given set-up for the part. Additional considerations on how each tolerance should be measured can render this decision hierarchy with multimodal capabilities. Such mode selection is closely related to the optimization of the process with respect to time and accuracy.

The **Feature** level first establishes a reference frame (i.e., a part coordinate system) and then generates subgoals to measure a goal feature selected at the Tolerance level. A feature definition in this level includes the definition of each tolerance and how well each tolerance is met.

It thereby provides quantitative in addition to qualitative (i.e., go/no-go) results. A possible extension to this level is inclusion of an error model to adaptively compensate for expected measurement errors. Such an error model can be used to conduct further interactive measurements of the feature in order to decrease uncertainty.

The **Surface** level decomposes a feature into surfaces and sends each surface to the Points level. The surface information should include an error model. Actual point measurements are transformed into a mathematical description of a feature or a datum reference frame.

The **Points** level has as its goal the measurement of individual surfaces. It decomposes surfaces into nominal measurement points. The nominal points on the surface are generated either by sampling the surface according to a pattern or by table look-up of external sampling guidance data. The Points level maintains the mathematical description of each surface measured. The error model in this level should incorporate the probe tip radii, offsets, and calibration data along with the each point measured.

At the **Path** level, measuring a point involves moving the probe from its current position to an offset point in preparation for measuring the nominal point. The probe then approaches the part until contact is made and the offset point is returned. The critical information needed at this level includes data for path optimization and collision avoidance.

Finally, each path from the Path level is translated to *move* and *measure* commands and is sent to the controller using the **Interface** level**.**

This overall architecture serves as the basis for verification of information requirements developed in the research for the feedback from production to design. As a result, it becomes a fundamental part of the global information model using the metadatabase structure.

Integration

The integration work capitalizes on past efforts that led to the development of an integrated inspection multimodal system. The focus will be on the integration of the verification system with the product database using a PDES/STEP framework, leading to the full integration of the system with the product database through the metadatabase AIME capability. Integration with the product database rather than with a specific 3-D modeler as done in the past enhances flexibility and increase the robustness of the system, since the neutral product model allows for compatibility with various modelers. Compatibility with PDES/STEP and other standards such as IGES [41] ensures compatibility with major commercial systems.

The thrust of the integration effort will be based on the design for a process that emphasizes verification. We will consider the design of an object so that verification/inspection in pre-, mid-, and post-process formats is easily accomplished. Consequently, the entire design process for the part will be integrally related to the area of verification from inception.

Architectural considerations must be taken into account to increase the speed of the inspection system so as to more effectively realize the trade-off between speed and accuracy for real-time inspection applications. The overall role of sensors and their optimum placement within pre-, mid-, and post-process verification, as well as the trade-offs between pre-, mid-, and post-process inspection with respect to cost, speed and re-work are important factors affecting adaptiveness of the overall architecture.

Similar to the linkage between the manufacturing view management system and the metadatabase system, the inspection system is joined with both through the metadatabase-supported open system architectures -- see Figure 3. In the figure, the Product Design and Inspection nodes represent supersets for Figures 4 and 5, respectively.

7. Conclusions: An AIME Prototype

The varied but related elements in the agenda above together lead to the development of a prototype environment for AIME. Such a prototype will facilitate researchers in the field to communicate with each other, calibrate their results, and collectively develop national or even international agendas for the next century.

From the perspective of AIME, cooperation -- including sharing of knowledge and development of standards -- is a key to success. Companies and researchers have been cooperating through government-led and industry-led consortia (e.g., OSF, PDES/STEP, CIMOSA and the various organizations under ISO, ANSI, and NIST) which involve virtually all major companies in information technology in the world. This is a fundamental testimony to the changing face of industry: No one company can dominate the market, nor the technology, and set the standards for others to follow. This need for working different vendor's technologies together is the driving force underlying the adaptive, open architecture that AIME highlights. More important to an enterprise is that cooperation among its functional areas and divisions is the key to success in competition. A company must overcome the "not-invented-here" syndrome and share knowledge to achieve its competitive goals. This cooperation must at least cover its various departments in perhaps globally distributed geographical locations; and for AIME, it should also extend to its vendors and even customers. Thus, the AIME prototype in its own right not only will verify the research results , but will also promote cooperation as necessary among companies and researchers.

Just like AIME, the notion of a prototype is generic. As a specific effort, however, three existing research systems developed for CIM and CE will serve as the starting components of the AIME prototype at Rensselaer. They include a metadatabase system [23], an object-oriented database system [11, 12], and an integrated inspection laboratory [7]. Additional efforts will then be developed using AIME results. Essentially, the results will provide extensions to existing systems and add new modules to the total environment to build a cohesive AIME prototype. The resulting prototype will include:

- The new information architecture including ROPE, the distributed rule-base manager, and modeling algorithms.

- The extended object-oriented technology with functionality for view creation and change management.

- The AIME information model including the extended information requirements for in-process verification and inspection, mini-metadatabase, and the global information model.

- AIME applications including in-process inspection (utilization of operating rules, feedback to production, and feedback to design) and other pertinent functions in both design and production.

The concept and unique promises of AIME will be verified using this prototype throughout its development. The prototyping effort will unite the parallel research tasks and give rise to a unique testbed that not only can be used for education and technology transfer, but that also represents a common ground to other potential research agendas and results in the general field.

In sum, a new vision extending beyond CIM and CE is formulated and proposed in this paper in the wake of a new era of global economy. This vision enables an integrated manufacturing enterprise to rapidly respond to customers' changing demands and other conditions in globally distributed environments. A particular model of this vision, AIME, is developed along with an approach to implementing it. The vision, including its formulation and problem analysis, is a contribution to both the research community and industry because it helps focus attention on the technical nature and requirements of future manufacturing as well as indicating a potential solution direction. This attention is expected to inspire first, awareness of the problem and second, particular approaches. The solution approach postulates basic principles as well as provides a practical model for realizing the vision. The research agenda, which is based on the model and the approach, substantiates the vision and contributes new technical results in its own right. Actual research based on the agenda is underway. In the future, the AIME vision will be developed further with new research agendas, approaches and results from all related efforts.

References

[1] M. Bouziane, *Metadata Modeling and Management,* unpublished Ph.D thesis, Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY., 12180.

[2] W. Cheung, *The Model Assisted Global Query System*, unpublished Ph.D thesis, Decision Sciences and Engineering Systems Department, Rensselaer Polytechnic Institute, Troy, NY., 12180.

[3] C.W. Chung, "DATAPLEX: An Access to Heterogeneous Distributed Databases", *Communications of the ACM,* Vol. 33, No. 1, pp. 70-80, January 1990.

[4] D.M. Dilts, and W. Wu, "Using Knowledge-Based Technology to Integrate CIM Databases" *IEEE Expert,* Vol. 3, No. 2, June 1991, pp. 237-245.

[5] R. Elmasri, and S. Navathe, *Fundamentals of Database Systems*, Benjamin Cummings Publishing Company, Redwood City, California, 1989.

[6] ESPRIT Consortium AMICE (eds.), *Open Systems Architecture for CIM,* Springer-Verlag, Berlin, Germany, 1989.

[7] L. Gerhardt, and G. Tarbox, *Design and Implementation of A Hierarchical Automated Inspection System*, International Conference on Manufacturing, Atlanta, GA, March 1990.

[8] L. Gerhardt, R. Noseworthy,, and A. Ryan,, *Camera and Laser Calibration for 3- D Visual Sensing* - Book Chapter, 1991 Clewer Publishers; Book Title "Intelligent Robotics Systems for Space Exploration".

[9] A. Goldfine, and P. Konig, *A Technical Overview of the Information Resource Dictionary System (Second Edition)* NBS Special Publication NBSIR 88-3700, National Bureau of Standards, Gaithersburg, MD, January 1988.

[10] M. Hardwick, D. Spooner, and E. Hvannberg, "Object-Oriented Change Processing for Concurrent Engineering", *Report* 91-5, Computer Science Dept., Rensselaer Polytechnic Institute, Troy, New York, (submitted for publication),

[11] M. Hardwick, D. Spooner, et. al., *ROSE 3.0: Tutorial Manual,* Technical Report, Rensselaer Design Research Center, Rensselaer Polytechnic Institute, Troy, New York, 1991.

[12] M. Hardwick, D. Spooner, et. al., *ROSE 3.0: Reference Manual,* Technical Report, Rensselaer Design Research Center, Rensselaer Polytechnic Institute, Troy, New York, 1991.

[13] M. Hardwick et al., "ROSE 3.0 Reference Manual", *Report,* Design Research Center, Rensselaer Polytechnic Institute, Troy, New York, 1992.

[14] S. Heiler, and S. Zdonik, "Object Views: Extending the Vision", *Proc. IEEE Data Engineering Conference*, IEEE Computer Society Press, 1990.

[15] S. Heiler, and S. Zdonik, "FUGUE: A Model for Engineering Information Systems and Other Baroque Applications", *Proc. of the 3rd International Conference on Data and Knowledge Bases,* Jerusalem, June, 1988.

[16] Theodore H.Hopp, *CAD-Directed Inspection*, Annals of the CIRP, Vol. 33/1,1984.

[17] C. Hsu, "Structured Database System Analysis and Design Through Two- Stage Entity-Relationship Approach", *Proc. 4th International Conference on Entity-Relationship Approach*, IEEE Computer Society, Los Alamitos, CA, 1985, pp. 56-63.

[18] C. Hsu, A. Perry, M. Bouziane, and W.Cheung, "TSER: A Data Modeling System Using the Two-Stage Entity-Relationship Approach, *Proc. of the 6th Entity-Relationship Approach Conference,* New York, NY, 1987, pp. 461-478.

[19] C.Hsu, and C. Skevington, "Integration of Data and Knowledge in Manufacturing Enterprise: A Conceptual Framework", *J. of Manufacturing Systems,* Vol. 6, No. 4, 1987, pp. 277-285.

[20] C. Hsu, and L. Rattner, "Information Modeling for Computerized Manufacturing", *IEEE Transactions on Systems, Man, and Cybernetics,* Vol. 20, No. 4, 1990, pp. 758-776.

[21] C. Hsu, M. Bouziane, L. Rattner, and L. Yee, "Information Resources Management in Heterogeneous, Distributed Environments: A Metadatabase Approach", *IEEE Transactions on Software Engineering,* Vol. SE-17, No. 6, June 1991, pp. 604-624.

[22] C.Hsu, "The Metadatabase Project at Rensselaer," ACM SIGMOD Record, Vol. 20, No. 4, December 1991, pp. 83-90.

[23] C. Hsu, G. Babin,, M. Bouziane, W. Cheung, L. Rattner, and L. Yee,, "Metadatabase Modeling for Enterprise Information Integration", *J. of Systems Integration,* Vol. 2, No. 1, Feb. 1992, pp 5-37.

[24] Industry Team and Facilitators, *21st Century Manufacturing Enterprise Strategy: An Industry-led View*, Vol. 1, Iacocca Institute, Lehigh University, Bethlehem, PA, Nov. 1991.

[25] J.Y. Joseph, S.M. Thatte, C.W. Thompson, and D.L. Wells, "Object- Oriented Databases: Design and Implementation", *Proc. of the IEEE*, Vol.

[26] A. Keller, "Updates to Relational Databases Through Views Involving Joins," in *Improving Database Usability and Responsiveness,* editor P.

[27] A. Kemper, and M. Wallrath, "An Analysis of Geometric Modeling in Database Systems", *ACM Computing Surveys,* Vol. 19, No. 1, March, 1987.

[28] W. Kim, et. al., "Features of the ORION Object-Oriented Database System", *Object-Oriented Concepts, Databases, and Applications,* edited by W. Kim and F. Lochovsky, ACM Press, Addison-Wesley, 1989.

[29] V. Krishnamurty, S.Y.W. Su, H. Lam, M. Mitchell, and E. Barkmeyer, "IMDAS, An Integrated Manufacturing Data Administration System", *Data and Knowledge Engineering*, Vol. 3, No. 2, 1988, pp 109-131.

[30] T. Landers, and R.L. Rosenberg,, "An Overview of MULTIBASE", *Second Symposium on Distributed Database,* North Holland, New York, NY, 1982, pp. 311-336.

[31] W. Litwin,, L. Mark, and N. Roussopoulos, "Interoperability of Multiple Autonomous Databases", *ACM Computing Surveys*, Vol. 22, No. 3, pp. 267-293, 1990.

[32] D. Maier, and J. Stein, "Development and Implementation of an Object- Oriented DBMS", *Research Directions in Object-Oriented Programming*, editors B. Shriver and P. Wegner, MIT Press, 1987.

[33] D. McLeod, "Interbase: An Approach to Controlled Sharing Among Autonomous, Heterogeneous Database Systems", *Data Engineering Bulletin*, Fall 1990, pp. 4-9.

[34] L. Rattner, *Information Requirements for Integrated Manufacturing*, unpublished Ph.D thesis, Decision Sciences and Engineering Systems Department, Rensselaer Polytechnic Institute, Troy, NY., 12180.

[35] M.P. Reddy, B.E. Prasad, P.G. Reddy, "Query Processing in Heterogeneous Distributed Database Management Systems", *Integration of Information Systems: Bridging Heterogeneous Database*, A. Gupta (ed.,) IEEE Press, pp. 264-277, 1989.

[36] S. Rem, "Guest Editor's Introduction: Heterogeneous Distributed Database Systems", *Computer*, Vol. 24, No. 12, December 1991, pp. 7- 11.

[37] A. Segev, and J. Park, "Maintaining Materialized Views in Distributed Databases", *Proc. 5th Data Engineering Conference*, IEEE Computer Society Press, 1989.

[38] D. Shipman, "The Functional Data Language DAPLEX", *ACM Transactions on Database Systems*, Vol. 6, No. 1, 1981.

[39] J. Shilling, and P. Sweeney, "Three Steps to Views: Extending the Object- Oriented Paradigm", *Proc. OOPSLA'89*, ACM Press, 1989.

[40] B. Smith, *Product Data Exchange Specification: First Working Draft*, NISTIR 88-4004, NTIS order number PB 89-144794.

[41] B. Smith, G. Rinaudot, T. Wright, and K. Reed, *Initial Graphics Exchange Specification (IGES)*, Version 4.0, Report NBSIR 88-38-3, National Bureau of Standards, Gaithersburg, MD., 1988.

[42] D. Spooner, M. Milicia, and D. Faatz,, "Modeling Mechanical CAD Data with Data Abstraction and Object-Oriented Techniques," *Proc. IEEE Data Engineering Conference*, IEEE Computer Society Press, 1986.

[43] D. Spooner,, M. Milicia, and D. Faatz, "Modeling Mechanical CAD Data with Abstract Data Types and Object-Oriented Database Systems", Proc. 2nd IEEE Data Engineering Conference, IEEE Computer Society Press, 1986.

[44] D. Spooner, "Creating a Product Database with ROSE", to appear in *Proc International Conference on Object Oriented Manufacturing*, Calgary, Alberta, Canada, May, 1992.

[45] A. Sheth, and J. Larson, "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Database", *ACM Computing Surveys*, Vol. 22, No. 3, 1990, pp. 203-236.

[46] J.M. Smith, et.al., "Multibase-Integrated Heterogeneous Distributed Data Systems," *AFIPS Proceedings*, Vol. 50, 1981, pp. 487-499.

[47] S.Y.W. Su, V. Krishnamurthy, and H. Lam, "An Object-Oriented Semantic Association Model (OSAM*)", in *AI in Industrial Engineering and Manufacturing: Theoretical Issues and*

Applications, edited by R. L. Kashyap, S. Kumara, and A. L. Soyster, American Institute of Industrial Engineers.

[48] U.S. Air Force, *Integrated Information Support System (IISS) Report*, Integrated Computer Aided Manufacturing (ICAM), Materials Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, OH, February 1983.

[49] R. Wang, and S. Madnick, "Facilitating Connectivity in Composite Information Systems", *ACM Data Base*, Vol. 20, No. 3, Fall 1989, pp. 38-46.

[50] L. Zeidner, and Y. Hazony, "Seamless Design to Manufacture (SDTM)", *Journal of Manufacturing Engineering*, Vol. 11, No. 4, 1992, pp 269-284.