A Hybrid Agent-Oriented Infrastructure for Modeling Manufacturing Enterprises

Weiming Shen and Douglas H. Norrie
Division of Manufacturing Engineering, The University of Calgary
2500 University Dr. NW, Calgary, AB, Canada T2N 1N4
E-mail: [wshen | norrie]@enme.ucalgary.ca

ABSTRACT

Manufacturing enterprises are now moving towards open architectures for integrating their activities with those of their suppliers and customers within wide supply chain networks. Traditional knowledge engineering approaches with large scale or very large scale knowledge bases are not suited for such widely distributed systems. Agent-based technology provides a natural way for resolving this problem. This paper presents a hybrid agent-oriented infrastructure for modeling manufacturing enterprises so as to integrate design, planning, scheduling, simulation, execution, material supply, and marketing services into a distributed intelligent open environment. In this paper, we discuss the requirements for next generation of manufacturing enterprises, and describe the main features of the proposed general infrastructure and the functions of its components. A machine-centered dynamic scheduling and rescheduling mechanism is then detailed and a prototype implementation is presented.

Keywords: Enterprise integration, knowledge engineering, distributed manufacturing systems, manufacturing scheduling, agent, mediator.

1 INTRODUCTION

Manufacturing enterprises are now moving towards open architectures for integrating their activities with those of their suppliers and customers within wide supply chain networks. To compete effectively in today's markets, manufacturers must be able to interact with customers, suppliers, and services rapidly and inexpensively. Traditional knowledge engineering approaches with large scale or very large scale knowledge bases are inappropriate because of the highly distributed nature of the systems. Agent-based technology derived from Distributed Artificial Intelligence provides a natural way for resolving this problem.

At The University of Calgary, we are now working on the MetaMorph II project whose enhanced capabilities will embody lessons learned from our previous research work. This paper describes our ongoing MetaMorph II project and presents its prototype implementation. The rest of this paper is organized as follows: Section 2 discusses the requirements for next generation of manufacturing enterprises; Section 3 introduces agent-based technology for modeling manufacturing enterprises; Section 4 presents our MetaMorph project; Section 5 describes the dynamic scheduling and rescheduling mechanisms developed for MetaMorph II; Section 6 presents the prototype implementation; Section 7 gives concluding remarks and perspectives.

2 REQUIREMENTS FOR NEXT GENERATION OF MANUFACTURING ENTERPRISES

Manufacturing strategy has shifted rapidly over the past ten years to support global competitiveness, new product innovation and introduction, and rapid market responsiveness. The next generation of manufacturing systems will be time oriented versus cost or even quality based. Such manufacturing systems should meet following fundamental requirements:

Enterprise Integration

In order to support its global competitiveness and rapid market responsiveness, an individual manufacturing enterprise has to be integrated with its related management systems (e.g., purchasing, orders, design, production, planning, control, transport, resources, personnel, materials, quality, etc.) which are, in general, heterogeneous software and hardware environments. Such integration may be realized via tactical planning systems that rely heavily on distributed knowledge-based systems to link demand management directly to resource and capacity planning.

Cooperation

Manufacturing enterprises have to fully cooperate with their suppliers and customers for material supply, parts fabrication, final product commercialization, and so on. Such cooperation should be in an efficient and quick-response manner.

In a cooperative system, dynamic chains of events are embedded in concurrent information processes. Requirements imposed by customer orders, managerial decisions, and design stages are integrated with the production planning and resource allocation tasks in a complex framework that incorporates high-level decisions into the planning activities. This is essentially a cooperative, concurrent information-processing environment. Cooperation is an imperative requirement for any complete functional model for advanced manufacturing systems.

Integration of humans with software and hardware

People and computers need to be integrated to work collectively at various stages of the product development, with access to required knowledge and information. Heterogeneous sources of information must be integrated to support these needs and enhance the decision capabilities of the system. Bi-directional communication environments are required to allow effective, quick communication between human and computers to facilitate their cooperation.

Agility

Economic globalization and expanding market expectations are rapidly transforming the environment for manufacturing. Considerable attention must be given to reducing product cycles to be able to respond more quickly to customer desires. In this new scale of economic transformation, corporations are progressively reorienting their strategies to expand their share of the market and to integrate "Agile" manufacturing into their production facilities.

Agile manufacturing is the ability to adapt in a manufacturing environment of continuous and unanticipated change and thus is a key component in manufacturing strategies for global competition. To achieve agility, manufacturing facilities must be able to establish convenient associations with heterogeneous partners. Ideally, partners are contracted with "on the fly" only for the time required to complete specific tasks. This type of interaction can also be used to plan long-term strategies. Agility will bring greater flexibility to the manufacturing organization without incurring large or diverted industrial investments.

Scalability

Scalability is an important property for advanced manufacturing systems. Scalability means that more resources can be incorporated into the organization as required. This property should be available at any working node in the system and at any level within the nodes. Expansion of resources should be possible without disrupting organizational links previously established.

To identify and incorporate new components into the system, organizational knowledge registries are required. When new physical components arrive in the system, representative entities are created to act as counterparts to the components throughout their life cycles. The ability to add new components incrementally allows the system to respond flexibly to a wide variety of requests. For example, the system might dynamically add increased intelligence and manufacturing capacity to supply a rapidly expanding market or reduce capacity to adjust downwards during low demand periods. When physical components are removed from the system for maintenance or other reasons, the listing of these components is removed from the system registry. Robust registration mechanisms are needed to provide ongoing integration of new components or the removal of existing ones

Dynamic reconfiguration

Both human beings and artificial entities in manufacturing systems need to be more alert to environmental changes. Every stage in manufacturing planning is affected by dynamic variations coming from either internal or external sources. In conventional manufacturing systems, the input from customers triggers a sequence of events, starting with planning operations. At this level, requests are processed according to preestablished stages (which include specification of product design, material management, manufacturing capacity planning and availability, and preparation of production costs). The planning process also triggers requirements for subcontracting external services.

Conventionally, the planning process progressively advances through a series of sequential evaluations that correspond to system conditions from an earlier time. Therefore, any subsequent variations in the state of the environment can make these plans invalid. This type of sequential system thus becomes expensive, since there is a tendency to reuse computing resources for redundant and repetitive evaluations.

Eventually, there is a transition from the planning process to a second major area of manufacturing control activity in which manufacturing plans and tasks are allocated execution times. More variations are introduced at this stage, which affect the stability of the system and its ability to execute plans according to schedule. This complex stage restricts the ability of the system to reconfigure to cope with dynamic and unforeseen changes.

Expandability is possible in such conventional manufacturing systems, but it requires major reconfiguration of the system. Tightly coupled interconnections among the system’s existing components make adjusting each processing module to new component availability very tedious. Similarly, the removal of components implies a considerable readjustment of the system.

Knowledge capitalization and distribution

The efficient capture and distribution of knowledge pertaining to each aspect of the organization - finance, marketing, design, and manufacturing - coupled with its effective use, will result in startling advances in market research, product and process development, production planning and scheduling, and ultimately customer responsiveness.

The more obvious problems in information processing observed in conventional manufacturing systems are highly centralized information management and production control, which corresponds to the need to maintain an overall system view in order to minimize costs (and so to win over more of the market). Centralized databases are commonly used to accumulate system information for establishing production plans and forecasting future requirements. Powerful centralized computers process large amounts of data to create production plans and schedules. Transactions among various resources are also forced to pass through a centralized control unit.

All activities in conventional manufacturing systems are limited by the accuracy and stability of the centralized processing components, yielding a fragile infrastructure.

Although centralized and sequential information-processing systems have in the past minimized hardware and software costs, their central structure is not suited to the inherent distributed nature of concurrent information flow in agile manufacturing.

Distribution of production knowledge will enhance system modularity and facilitate both integration and reconfiguration. The increased modularity reduces the complexity of organizing knowledge by maintaining knowledge locally. Information held locally can be processed concurrently, thus avoiding the limitations of sequential information processing

Concurrent Engineering

Ensuring the manufacturability of the product constitutes the first step in implementing concurrent engineering. Geometric and functional specifications, availability of raw materials, and the capability and availability of shop-floor resources each has a major influence on manufacturability. A design may be manufacturable under one combination of product requirements and shop-floor resources, but not under another. The selection and availability of stock material from which the part will be manufactured influences the number of intermediate steps required, and hence the production cost. The capability and availability of shop-floor resources impact the process plan to be used, and again the production cost. Thus, all of these aspects need to be considered simultaneously for effective concurrent engineering.

3 AGENT-ORIENTED APPROACHES FOR MODELING MANUFACTURING ENTERPRISES

3.1 Knowledge Sharing and Reuse

From the perspective of knowledge intensive engineering, we can view all relevant aspects of an organization domain in terms of 'knowledge'. This applies to the structure and nature of the organization itself, the data used within different components of the organization and the flow of this data through the organization along with the value added to it during the execution of the organizational tasks. This knowledge exists in the form of data (factual assertions about the organization and its tasks) which may reside in existing information systems (such as databases), and in the form of specific 'business rules' applied to this data in order to carry out some function within the organization, either relating directly or indirectly to the tasks at hand. The focus of knowledge systems techniques is the explicit representation of this organizational knowledge in a form that is optimal for effective reasoning about the tasks in the organization, as well as for the representation of this information to those assigned the execution of line tasks (Davis and Oliff 1988).

Previously, large scale or very large scale knowledge bases have been often advocated for engineering applications including design, manufacturing, operations, and maintenance, because these activities require an extremely huge amounts of and various kinds of knowledge (Forbus 1988). But, according to Tomiyama et al (1995), not only the quantity of knowledge but also the quality of knowledge in terms of sharability and reusability of knowledge is crucial. Knowledge intensive engineering aims at both the amount and flexibility of knowledge. A single knowledge base can make inferences in a particular circumstance but it may hard-fail. Therefore just having a large scale or very large scale knowledge base alone is not enough for modern manufacturing whose robustness and reliability are actually important.

Knowledge in modern manufacturing must be well organized and should be able to be flexibly applied to different kinds of applications. Figure 1 compares three possible different types of knowledge sharing architectures (Tomiyama et al 1995). Figure 1(a) depicts a situation with independent knowledge bases. In this case, the 'strength' of knowledge is just a sum of each of independent knowledge bases. Integrated knowledge bases can be represented as in Figure 1(b). Here, the knowledge bases can be applied to various situations and the 'strength' of knowledge is near maximum. However, this requires having a platform with a uniform language. The Cyc project (Lenat and Guha 1989) is an example of this approach. In Figure 1(c), independent knowledge bases can communicate and form an interoperable situation, although the 'strength' of knowledge might be weaker than that in Figure 1(b). The entire knowledge base is a federation or a set of loosely coupled intelligent agents. This approach has recently been used by projects like SHADE (McGuire et al 1993), PACT (Cutkosky et al 1993), DESIRE (Brazier and Treur 1996), and DIDE (Shen and Barthès 1997).

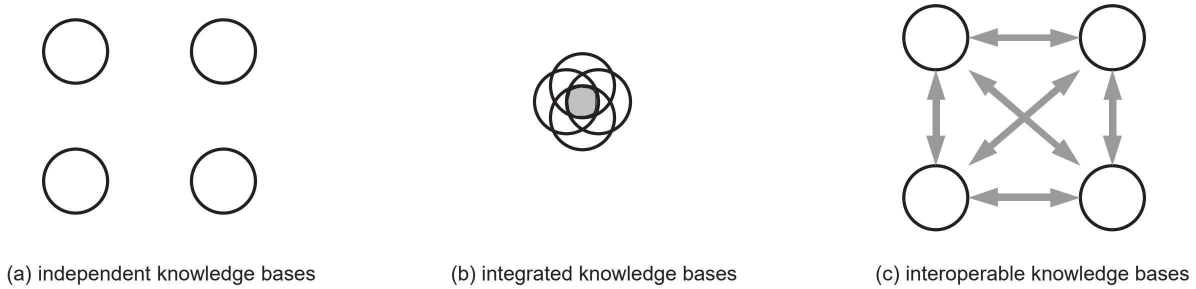

Figure 1. Knowledge sharing architectures (Tomiyama et al 1995)

The second architecture (Figure 1(b)) may suffer from the lack of uniform knowledge representation. Unless carefully designed, the platform language cannot cover everything. The third architecture (Figure 1(c)) may overcome this problem, if the framework is abstract enough to incorporate the different types of ontology each agent uses. However, it does not completely avoid the problem, because communication among agents requires at least understanding what other agents are talking about.

The same communication requirements justify the representation model used in SHADE (McGuire et al 1993). The model, called KIF (Knowledge Interchange Format) (Genesereth & Fikes 1992), is a machine-readable version of first order predicate calculus, with extensions to enhance expressiveness. KIF specifications define syntax and semantics; ontology defines the problem-specific vocabulary. Agents exchange sentences in KIF using the shared vocabulary.

To support the sharing and reuse of formally represented knowledge among AI systems, it is useful to define the common vocabulary in which shared knowledge is represented (Patil et al 1992). A specification of a representational vocabulary for a shared domain of discourse definitions of classes, relations, functions, and other objects is called an *ontology* (Gruber 1993). The need for a shared ontology is a direct result of the multidisciplinary nature of engineering. There are many different views of a design (function, performance, manufacturing), each with a different language. However, the various perspectives typically overlap, necessitating the sharing of information if design is to proceed concurrently and cooperatively. For information to be shared, there must be a commonly understood vocabulary. A detailed discussion on ontology can be found in (Gruber 1993). An application of ontology in enterprise modeling was proposed by Fox et al (1996).

In design applications, it is necessary to represent knowledge at several levels: domain knowledge associated with the particular vocabulary used in the design domain, but also general

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**