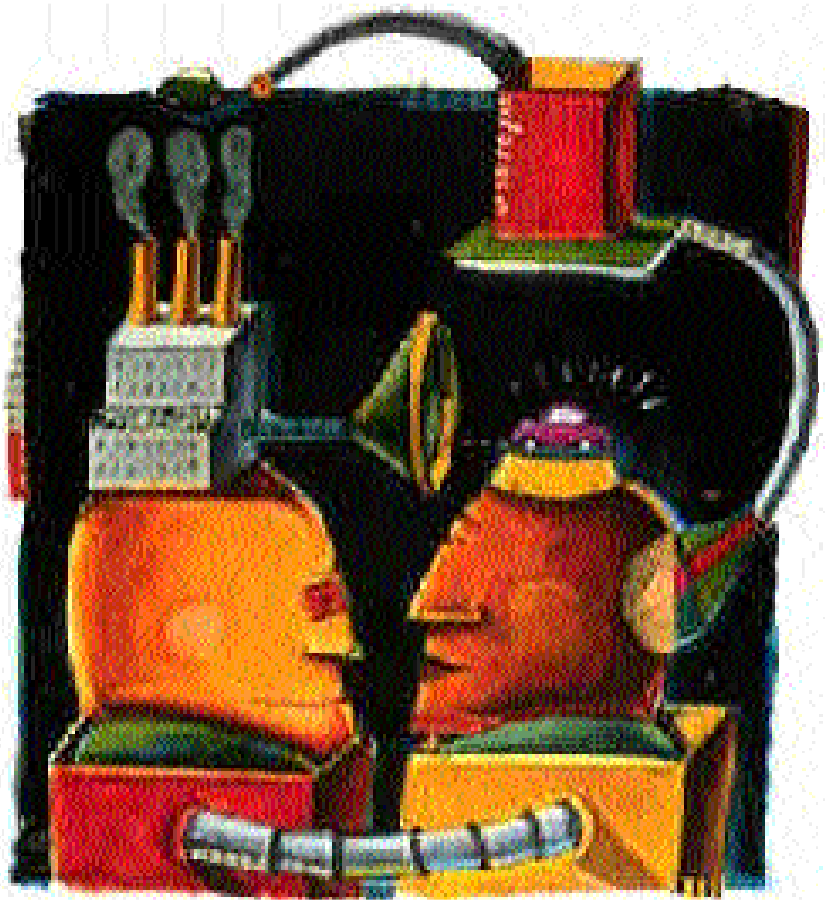
Agent-based Manufacturing

Part 1 of 2

Putting agents to work in the real world

James J. Odell & David Greenstein
jodell@compuserve.com / dgreenst@tir.com



Centralizing a corporation was once considered an efficient way to run a manufacturing enterprise. Decisions and information processing occurred in an orderly, top-down, hierarchical manner. However, it is now clear that this type of system only works in a reasonably stable market. Globalization and changes in technology are causing today's market to be in a state of constant change. Companies that cannot adapt fast enough and thrive in new markets will be left behind.

In response, a major automotive company is building an agent-based manufacturing system.[1] Here, the agents not only adapt to their environment but can also evolve by learning *from* the environment. This new architecture—the Agile Manufacturing Information System (AMIS)—prepares manufacturing enterprises for the increasingly complex marketplace and enables them to respond rapidly to change.

The Traditional Manufacturing Approach

In traditional manufacturing, information systems mimic organizational structures, using a top-down, command-and-control structure. Communicating decisions and information down through the organization is time consuming—making it impossible to respond and adapt quickly to external forces.

Furthermore, traditional manufacturing relies on schedules as a means of forecasting what needs to be produced. Schedulers sequence jobs based on the assumption that the environment will not change significantly during the schedule's time span. This approach works adequately in a predictable market. However, in a turbulent marketplace, a schedule is impractical. Any small, unanticipated change in demand or factory floor conditions can render the schedule obsolete.

Another problem with traditional schedulers is that they try to anticipate and plan for every possible change. Unfortunately, the range of scenarios and the possible combinations of parameters are infinite because manufacturing is so complex. Even if it were possible to precode all possible scenarios, the cost of considering and programming all possible combinations is prohibitive. An unanticipated scenario could cause the system to fail.

In short, traditional manufacturing facilities have shortcomings that affect their ability to compete in today's constantly changing marketplace.

- They do not have mechanisms in place to accommodate rapid changes in business conditions caused by global competition and changing market demands.
- They do not have mechanisms in place to implement technology refresh smoothly.
- They are rigid and slow to make significant organizational or functional changes.
- They do not have a mechanism to recover gracefully from partial failures on the factory floor.
- They are unable to form or to participate in virtual enterprises.
- They are not scalable for changes in the market.
- The business model and the operational philosophy are not customer driven.

These shortcomings cause problems such as reduced productivity, increased costs, and missed market opportunities. To remain competitive in today's marketplace, manufacturing must change its approach.

An Agent-Based Solution

AMIS is a new operational model that addresses the problems of traditional manufacturing practices. Because today's dynamic marketplace is similar to ecosystems, AMIS is modeled after the behavior of the natural world, an approach which is agile, adaptive, and dynamic. It can adapt to changes in the marketplace and in technology, making it effective and competitive.

Traditional manufacturing systems rely on a rather rigid, top-down structure to represent a manufacturing enterprise. AMIS uses a loose aggregation of software agents to represent a manufacturing entity. For example, *resource agents* represent the capabilities and capacity of the various resources available, such as machines, tools, people, and computers. The work performed within a facility is represented by *job agents*. In a small system, the interaction of the resource agents with job agents manages the manufacturing process.

However, in systems involving many jobs and resources, the interaction could tax even modern information systems. Here, resource agents can be grouped into *cells*. Because cells are agents in their own right, they can form virtual organizations—able to adapt constantly to the changing global marketplace. This dynamic structure enables each cell, as well as the overall business, to remain agile. Rather than being constrained by a fixed hierarchy, it can thrive in a continuously changing and unpredictable environment.

Each cell, then, can be treated as a manufacturing business unit. Being responsible for its own bottom line, each cell must be profitable over time. When a cell is consistently unprofitable, it is dissolved and other cells absorb its resources. Similarly, each resource in a cell is responsible for maintaining a positive bottom line and contributing to the cell's overall profit. This distributed profit responsibility allows the cell to maintain the right mix of resources for the current workload and the appropriate size, while maintaining the flexibility to address future needs.

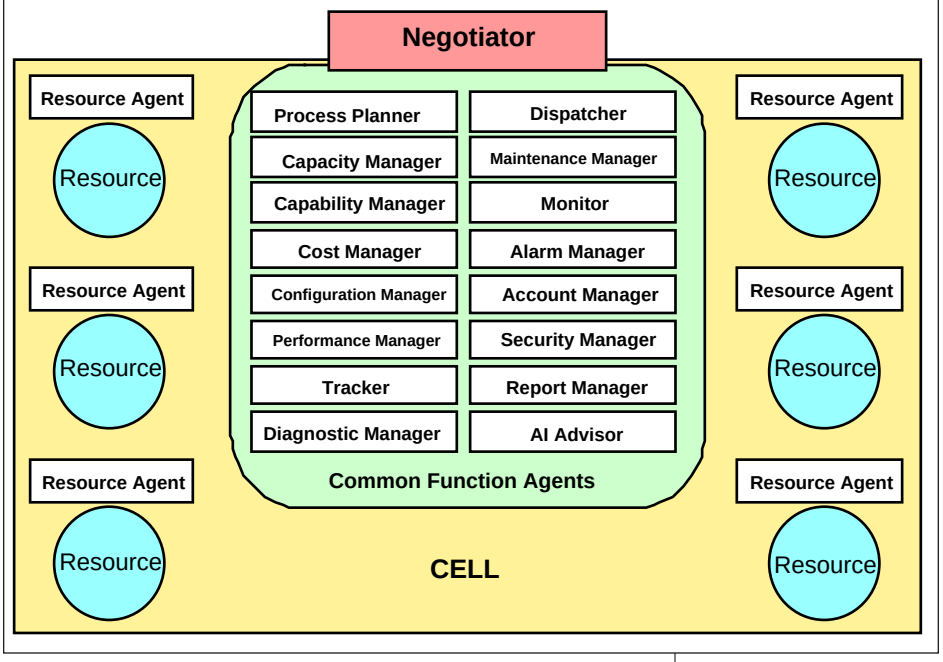
For a manufacturer to succeed in today's competitive world, it must have the optimal mix of people, equipment, and knowledge to make the product. The AMIS architecture provides the flexibility and agility in a software system that can enable a manufacturer to monitor, evaluate, and adjust the mix of resources, people, and tools required as market demands change.

The AMIS Agents in More Detail

Cell Agents

In living systems, a cell is a self-contained unit that has its own structure and behavior. It consists of other self-contained structures that interact to support the cell. How well the cell and its components work together determines whether the cell lives or dies. In a manufacturing system, each cell agent is a business unit representing a collection of physical resources, including machines, tools, and people. The cell operates as a self-contained business unit and only continues to exist if it meets its profit and production goals and responsibilities. The cell also controls its own size—it changes the mix and number of resources over time to maintain its profitability and competitiveness in the marketplace. The architecture of a cell is summarized in Fig. 1.

Figure 1. Cell architecture.



Common Function Agents The common function agents interact with each other and with the resource agents. They provide the complete set of business functionality required to operate the cell as an independent business unit. Each common function agent is responsible for a different business or manufacturing function within the cell. Some of these agents contain information about the resources within the cell, such as the capabilities of the resources. Other agents provide interfaces to the people working in the cell, such as process planners and machine operators. For example, the maintenance manager schedules and directs maintenance activities whether they are scheduled, opportunistic, or reactive. It also keeps track of the maintenance history.

The process planner determines whether the cell will bid on RFQs (requests for quote) received by the cell. A broad analysis is made of the cell to judge its ability and desire to produce a quote for this RFQ. The analysis uses criteria associated with the type of product being requested (automotive, pharmaceuticals, electronics); the processes needed (welding, casting, packing); and the resources needed (five-axis CNC, drill, sheet metal press). If the cell either lacks the appropriate abilities or cannot subcontract them, the cell will not bid on the RFQ. If the cell does bid on the RFQ, the process planner generates the process workflow (e.g., a UML activity diagram) that will be used to execute the quote, if selected.

The capability manager uses the process workflow to verify that the cell has the resources needed to carry out the job. It verifies each step in the process (job agent) with the available resources in the cell. The verification of the capability is based on the information contained in the workflow (time, quality, special characteristics, and cost criteria). If the capability is not present in the cell, the capability manager initiates the subcontracting process through the process planner.

Similarly, the capacity manager uses the information from the workflow to provide capacity to the job agents. The jobs currently accepted by the cell are taken into consideration when deciding if the cell has the capacity to take on this new job. If sufficient capacity is not present in the cell, the capacity manager initiates the subcontracting process through the process planner.

Negotiator Agents *Negotiator agents* (at the top of the cell architecture diagram in Figure 1) communicate with the outside world on behalf of the cell. The negotiator provides an interface between the cell and the outside world. It routes messages received from the outside world to the appropriate common function agent. When preparing quotes for new jobs, the negotiator assembles the quote information provided by the other agents and summarizes the final quote information for the customer. Similarly, when the cell receives quotes from subcontractors, the negotiator works with other agents to select the winning quote.

Resource Agent Each *resource agent* represents a physical resource within the cell: a machine, a tool, a computer or a person. Each physical resource provides a specialized utility or function to add value to the order completion process. The resource agent captures the attributes of the physical resource, allowing the agent to represent it in the cell and to coordinate the cell's use of the resource.

Each resource agent contains a list of capabilities that define the processes that the physical resource can perform. For example, a resource might be able to perform several types of milling operations. The capability list allows the resource agent to determine whether to bid on the various jobs in the cell.

Resources keep track of their assigned jobs by maintaining a prioritized list of jobs that the resource wins. Each job defines its job type, the earliest start time for the job, the expected job duration, the latest finish time, and the estimated cost.

The resource agent also maintains profit and loss figures for the resource. The best interest of a resource is to maximize profit by working on as many jobs as possible. If the resource does not maintain a profit over time, the cell may sell the resource to another cell. The resource agent is responsible both for ensuring the resource is optimally utilized and for representing the resource when bidding for new jobs.

Job Agent The *job agent* represents the customer through the order placed into the system. The job agent defines the processes needed to complete the final product specified in a

customer order. Each node in the process workflow is a subjob and is handled by an agent. Each subjob agent contains information about that specific process, including the type of process, set-up time, run time, and cost.

The job agent is responsible for monitoring its current status and due dates. As due dates approach for the overall job or for individual subjobs, the subjob agent will raise alarms to initiate corrective action. The subjob also communicates with its neighboring subjobs, passing state information and alarms to allow the previous and following subjobs to monitor more accurately their status and take appropriate action. The job and subjob agents are active agents responsible for making sure that they are completed by the expected due dates and at the lowest cost possible.

Broker Agent The *broker agent* helps customers find providers of services and products. In the AMIS environment, each provider is a cell that registers with the broker, specifying the types of products and services it provides. For example, car buyers do not have to contact each automobile manufacturer. Instead, they send the attributes of the desired model (including such criteria as price, delivery date, color, and accessories) to the broker. The broker forwards the request to each automobile-producing cell that has registered with the broker.

The customer specifies the date by which all cells must provide quotes. The broker waits until this date and then collects all the cell bids and returns them to the customer. When the customer selects a winning quote, the broker forwards the award notification to the winning cell. Losing cells can view the attributes of the winning bid and compare it to their bid, in order to improve their bids in the future.

AMIS organizes brokers in a hierarchy based on geographical regions. First, the local broker forwards the customer request to its registered local cells. If no local cell can meet the customer request, the broker forwards the request to the regional broker, who in turn forwards the request to each local broker within that region. In turn, these local brokers forward the request to every cell within their local area that manufactures the requested product.

The bids from each cell pass back through this hierarchy, going from the local information brokers to the regional broker. The regional broker returns the bids to customer via the local information broker that originally received the customer request.

If no cells within the region can meet the customer requirements, the regional broker forwards the request to the global broker. The process is repeated with the global broker forwarding the request to each regional broker, down through the local information brokers, and to each cell worldwide that produces that product.

In some cases, the customer may wish to solicit quotes from cells worldwide without initially limiting the scope to cells registered with the local broker. In this case, the customer sends the requirements directly to the global broker, bypassing the local and regional brokers.

Conclusion AMIS provides a means for a manufacturer to be more productive and adaptive in responding to changing market demand. Specifically, it will allow a manufacturer to:

- increase machine (resource) utilization by better matching capacity to workload
- increase throughput by making the right products at the right time
- reduce the number of late jobs by better capacity planning and monitoring
- utilize/tune the correct resource types and mix by monitoring resource efficiency
- create a flexible and dynamic architecture that responds rapidly to a continuously changing market
- enable an activity-based costing (ABC) approach to collect and calculate actual production costs
- reduce "single points of failure" in production systems

In Part 3 of this article, we will discuss how these agents work together to produce a working, agent-based, manufacturing system.

Reference

1. Greenstein, D. and Thomas, K., Intelligent agents for an emergent industrial ecology, *Intelligent Manufacturing Systems*, Proceedings of IJCAI, 1995.

James Odell <<www.jamesodell.com>> is a consultant, educator, and author. David Greenstein is a project manager in the Manufacturing Information Systems department at General Motors.