The AARIA Agent Architecture:
An Example of Requirements-Driven Agent-Based System Design

H. Van Dyke Parunak*, Albert D. Baker**, and Steven J. Clark*

* Center for Electronic Commerce
Industrial Technology Institute
PO Box 1485
Ann Arbor, MI 48106
{van, sjc}@iti.org

** Dept. of Electrical and Computer Engineering
University of Cincinnati
PO Box 210030
Cincinnati, OH 45221-0030
bert.baker@uc.edu

Abstract

Designs for real-world agent-based systems must reflect domain requirements as well as the technical capabilities of agents. This needs-driven approach is being applied in AARIA (Autonomous Agents for Rock Island Arsenal), an industrial-strength agent-based factory scheduling and simulation system being developed for an Army manufacturing facility. A review of the operations of Rock Island in the light of broader industrial needs yields seven requirements. After introducing the AARIA agent community, we summarize each of these requirements and how AARIA supports it. More information is available at http://www.aaria.uc.edu.

The AARIA Agent Community

In discrete manufacturing, Parts move through a network of Unit Processes (UP’s) and Buffers. Each UP acquires one or more input Parts from Buffers of the needed types, and engages certain Resources to produce one or more output Parts into appropriate Buffers. In AARIA, we use linguistic case theory [Parunak 95] over a set of declarative sentences describing the domain to identify candidate agents, then refine this population using the requirements. Figure 1 shows the resulting community, with its flows of Parts and Engagements for Resources along orthogonal axes that intersect at UP’s.

Previous research on agent-based factory control and scheduling (including our own) differs widely on what is represented as an agent: levels in a hierarchical decomposition of the factory (Butler & Ohtsubo 92, Tilley & Williams 92, Parunak 87), Resources (Shaw & Whinston 85, Baker 96, Parunak et al. 87, Heaton 94), or Parts (Maley 88, Duffie et al. 88). In AARIA, Separate full-fledged agents represent parts, resources and unit processes with substantially equal intelligence and responsibility in each type of agent.

The AARIA system has a number of other important enhancements from past work in multi-agent manufacturing. The system incorporates new features for schedule optimization and fault recovery. Unlike many past systems, these agents are being implemented in an agent infrastructure that allows true agent behavior by supporting true broadcast and multicast communications, subject based mail handling, multithreaded agent activity, multi-platform instantiation, agent migration, and the implementation of multiple scheduled activities within an agent. Also, the system is being implemented with dual functionality so it can run a real factory or run in simulation mode. The sections below outline AARIA's seven design requirements and highlight other enhancements to past work.

Meeting the AARIA Requirements

Uniformity (An operation at the boundary of AARIA interacts with external suppliers or customers in the same way that it does with internal ones.)—Both an external customer and an internal UP draw from Buffers, and both an external supplier and a UP feed Buffers. Thus UP’s represent both customers and suppliers. This result offers a novel approach to supply-chain integration. Traditionally, each firm in a supply chain is viewed as a monolithic entity with its own internal mechanisms, requiring a special set of mechanisms to interact with other firms. AARIA views each firm as an internal supply chain. As a result, the interfaces between AARIA agents within a firm are the same as those between one firm and another, and integrating multiple AARIA-based firms into a supply chain becomes immediate and transparent.

Metamorphosis (The system maintains continuity between different entities that represent different stages in a common life cycle, for example, an order for a part, the part itself, and its production history.)—AARIA distinguishes persistent agents, whose behavior does not change over the time scale involved in daily shop operation, from transient agents, which represent interactions among persistent agents and which go through a
well-defined life cycle of behaviors as the shop operates. Buffers, Resources, and UP’s are persistent agents, and generate transient agents by their interactions. The transient agents (Engagements, Operations, and Parts) model interactions between a UP and some other agent.

Least-Commitment (The customer’s statement of demand (e.g., product specification, quantity, price, and delivery time) develops interactively, rather than being specified in detail at the outset.)—Resource agents support least-commitment scheduling through a novel density-based scheduling kernel (DESK) that schedules operations within dynamically adjusted windows rather than to fixed points in time and rewards customers for requesting less stringently timed deliveries.

Modality Emergence (An entity’s factory control modality emerges dynamically from its operation in the system, rather than being hard-coded or inferred explicitly.)—Traditional factory control modalities are points in a two-dimensional space (Figure 2), reflecting the degree to which the system is constrained by external commitments that it has made to others, and by the degree of demand that others place on it. Traditional approaches, and much past agent-based work as well, choose a point in this space in which the system as a whole will operate. In AARIA, (1) points in the space intermediate between the classical three points are accessible; (2) different agents can be at different places in this space, and (3) agents can migrate through this space and thus change modality as circumstances change.

Frequent Change (System behavior adapts dynamically as the environment changes.)—Multi-agent systems can potentially reconfigure and adjust to change, in apparent violation of the Second Law of Thermodynamics. To obtain self-configuration at a macro level, many naturally occurring agent-based systems couple agents’ macro behavior to a low-level dissipative mechanism, such as pheromone evaporation [Kugler and Turvey 87]. This micro mechanism both generates a flow field to which agents can orient themselves and provides an “entropy leak” from the macro to the micro level. One such dissipative mechanism is currency flow in a market. AARIA draws on this dynamic by instituting markets among the agents, extending the methods of [Baker 96]. To interface with the outside world (as Uniformity requires), AARIA’s currency is denominated in dollars.

Empowerment (Human stakeholders, including operators, manufacturing engineers, and managers, receive the information they need, with interfaces to let them control the system rather than being controlled by it.)—Humans interact with the system through two types of agent: Manager and Resource. Each Manager is a watchdog for some aspect of system performance (such as cash flow, or chaotic behavior, or equipment utilization), and interacts with other agents by cash grants and taxes that adjust the underlying dynamics of the system and coax it into new behaviors. Operators are a subclass of Resource. These Resource agents represent the operators’ needs and wishes.

MRP Functionality (The aggregate behavior of the agent community subsumes functionality currently provided by MRP II systems.)—MRP functionality emerges naturally from the architecture that we have constructed to satisfy the other requirements. The flow of bids along the part axis mirrors the bill-of-materials decomposition of the product that is the core of MRP. The different operating modalities support planning and finite-capacity scheduling. Uniformity means that order entry and purchasing are modeled at every operation, not only at the limits of the firm. The use of currency flow for self-organization makes each agent a profit center and naturally supports integrated financial functions. Demonstration of these features provides a final enhancement to past work.

Acknowledgments.—AARIA is funded by DARPA Agile contract F33615-95-C-5524 to Intelligent Automation, Inc. (Len Haynes, Kutluhan Erol, and Renato Levy), and is managed by USAF ManTech. Contractors are the Industrial Technology Institute (Van Parunak, Steve Clark, Jorge Goic), the University of Cincinnati (Albert Baker, Bradley Matthews, Ben Moore, Brian Birtle, Veena Pandiri), Flavors Technology (Howell Mitchell), and Rock Island Army Arsenal (Greg Peters, Don Tice).

References


