ExPlanTech: Multiagent Support for Manufacturing Decision Making

Michal Pěchouček and Jiří Vokřinek, *Gerstner Laboratory, Czech Technical University*

Petr Bečvář, *CertiCon*

ExPlanTech's multiagent approach offers a unified framework for decision-making support and provides a proven alternative to known mathematical and system science-modeling technologies for simulating the manufacturing process.

ExPlanTech is a consolidated technological framework resulting from a series of European Union Research & Technological Development and Trial projects in agent-based production planning. ExPlanTech provides technological support for various manufacturing problems and comprises different components, which you can assemble to develop a customized system that supports a user's decision making in different aspects of production planning. The system should help human users size resources and time requirements for a particular order, creating production plans, optimizing material resources manipulation, managing and optimizing supply chain relationships, visualizing and analyzing medium- and long-term manufacturing processes, and accessing external data.

Using ExPlanTech's multiagent architecture as a foundation for a software system, you can create a component-based, flexible, and reconfigurable system that allows distributed computation and flexible data management. In Gersntner Laboratory, we integrated each ExPlanTech component in an agent wrapper that complies with the FIPA (Foundation for Intelligent Physical Agents, www.fipa.org) standard for heterogeneous software agents. You can use these components in various configurations or independently as standalone applications. System configurations can contain various planning, data-management, or visualization agents. ExPlanTech also offers an *agentification* process that integrates an enterprise's existing software and hardware into an FIPA-compliant agent.

Deploying agent technologies in manufacturing problems (see the related sidebar) lets you process relevant production data distributed across the enterprise. A classical approach that collects and processes data centrally has difficulty dealing with situations with voluminous and frequently changing production-planning data. An agent approach lets you process data proactively at its place of origin and exchange only necessary results. ExPlanTech, or any agent-based technology, certainly doesn't provide an uncomplicated solution of NP-hard planning problems. However, agent technology lets you integrate heavy-duty AI problem solvers (such as constraint satisfaction systems, linear programming tools, genetic algorithms, and so on). Such technology works well for integrating manufacturing enterprises into a supply chain. In terms of planning, it's irrelevant whether a system reasons about an in-house manufacturing workshop or about a subcontracted company. Additionally, production managers are often interested in production process modeling and simulation. Using ExPlanTech in a simulation environment can simplify experimenting with changes to production lines and help show how they affect the manufacturing process. ExPlanTech doesn't offer agents and components for control and real-time diagnostics. Even though ExPlanTech technology is in principle open to providing support for control and real-time diagnostics, the current implementation doesn't feature any such agents or components.

ExPlanTech

Given our long experience working with the manufacturing industry, we identified several key requirements that industrial users in manufacturing frequently request. They want a system that is

- *Open, extensible, and general.* Allows customization for several different domains and lets you expand functionality (such as extending production planning to supply chain management).

1094-7167/05/$20.00 © 2005 IEEE
Published by the IEEE Computer Society

Agent Technologies and Multiagent Systems

Agent technologies and the concept of *multiagent systems* (MASs) originated in artificial intelligence and computer science, drawing from principles of component-based software engineering, distributed decision making, parallel and distributed computing, autonomous computing, and advanced methods of interoperability and software integration.[1] Agent-based system operations are based on collaborative (or sometimes self-interested) interactions of autonomous and loosely coupled software or hardware entities—*agents*. An agent can integrate existing software systems required for operating the manufacturing enterprise, hardware modules such as computer numeric control machines, and various programmable logic controllers with advanced planning systems, simulation environments, diagnostic algorithms, or sophisticated control mechanisms.

Agents technologies suit domains that have one of the following properties:

- Requires solving highly complex problems or controlling highly complex systems
- Has distributed, not centrally available information required for solving problems or controlling systems
- Has a dynamically changing environment and problem specification
- Must integrate a high number of heterogeneous software (and possibly hardware) systems

Application areas

Several agent technology application areas typically relate to manufacturing. In production, we solve highly complex planning problems, so we must control dynamic, unpredictable, and unstable processes. We might also need agent-based diagnostics, repair, reconfiguration, and replanning.

For virtual enterprises and supply chain management, we have requirements for forming business alliances, planning long-term and short-term cooperation deals, and managing (including reconfiguration and dissolving) supply chains. So, we also can use various agent technologies for agents' private knowledge maintenance, specification of various ontologies, and ensuring service interoperability across the supply chain. For Internet-based business, we can use agent technologies for intelligent shopping and auctioning, information retrieval and searching, remote access to information, and remote system control.

Additionally, we can use MASs to manage transportation and material handling and for optimal planning and scheduling—especially in cargo transportation, public transport, peacekeeping missions, military maneuvers, and so on. Also, agent technologies nicely pair with managing utility networks such as energy distribution, mobile operators, and cable providers. We might use distributed autonomous computation for simulation and predication of alarm situations, prevention of blackouts and overload, and intrusion detection.

Available systems

Classical planning systems (using scheduling algorithms with various heuristics, constraint logic programming, genetic algorithms, and simulated annealing[2,3]) work centrally and allocate resources usually in one run for every product order in the system. These methods use mostly stochastic algorithms and generate near-optimal solutions to minimize the defined criteria (for example, sum of weighted tardiness and inventory costs). Such solutions are fully sufficient for planning in stable environments. However, in an environment with requirements to continually revise the plan, these approaches would breach the *calculative rationality requirement* (the minimal time required for two relevant changes in the environment is larger than the maximal time needed to process the change). When replanning is required, the plan is usually completely rebuilt and the algorithms' random aspect can cause major, unwanted changes, which makes this approach unsuitable for many manufacturing areas. For physically distributed production units, it's advantageous to break down and distribute the planning problem.[4,5]

Multiagent technology can address a wide range of manufacturing decision-making support problems, but few MAS implementations cover more than a single type of a problem. Solutions exist for low-level scheduling or control systems as well as product-configuration and quotation phases for short- and long-term production planning and supply chain management.[6–8]

Prosa (products-resource-order-staff architecture) is a reference architecture for manufacturing control.[9] It's mainly oriented to interholon architecture and identifies kinds of *holons* (agents)—their responsibilities, functionality, structure, and interaction protocols. Prosa defines three main classes of holons: *product, resource,* and *order*. Product agents manage production procedures and process techniques—for instance, which operations to perform to achieve the product. Resource agents represent resources such as machines. Order agents represent manufacturing orders and are responsible for following deadlines. The Prosa architecture's authors also designed *staff* agents to give the previous basic agents sophisticated knowledge support.

The Gerstner Laboratory's ProPlanT (*pro*duction *plan*ning *t*echnology) is a hierarchical technology; it comprises several basic agent classes that let you model the enterprise structure by level (which depends on the chosen granularity). From this point of view, Prosa's basic structure seems rather flat. ProPlanT's architecture seems quite static and consists of a relatively reasonable number of agents. Contrary to Prosa's architecture, the number of products, tasks, or orders doesn't affect the size of the ProPlanT agent community. On the other hand, exchanged messages in ProPlanT are more complex compared to messages in Prosa. The designers of the Prosa architecture admit that using the basic structure as the reference architecture has serious drawbacks: Because of the scalability problem, a community could end up with numerous agents (according to the size of the factory and number of products). It could provide unpredictable behavior. Additionally, Prosa's optimization has not been adequately addressed.

These drawbacks led Prosa designers to aggregate related holons to create the staff holon, a bigger holon with its own identity. Staff holons empower basic holons but play only an advisory role to avoid conflicts with the system's hierarchical rigidity. We can say that holons in Prosa are loosely coupled while in ProPlanT-like architectures a strong link exists among agents on different hierarchical levels.

More recent solutions, such as the Micro-Boss system (created at the Robotics Institute at Carnegie Mellon University), let scheduling systems constantly revise their scheduling strategies while constructing or repairing a schedule. Micro-Boss can meet new demands on the planning system and uses deterministic algorithms. Unfortunately, despite many decentralized aspects, it's a centralized approach.

The Robotics Institute also successfully integrated Micro-Boss into the MASCOT (*m*ulti *a*gent *s*upply chain *co*ordination *t*ool) system. They developed MASCOT for dynamic supply chain creation and coordination, and it doesn't affect the intra-enterprise level. It solves the problem of dynamic reconfiguration and supply chain creation and adequately covers demands on integration and enterprise cooperation. At the University of Calgary, they developed MetaMorph[10] as a generic multiagent architecture. It tries to cover every phase of the manufacturing process using mediator-centric federation architecture but provides only a simple research prototype.

Additionally, many other initiatives have investigated agent-based organization of manufacturing processes. The MADEFAST project demonstrated collaborative engineering possibilities.[11] The AARIA system integrates manufacturing capabilities (for example people, machines, and parts) in a MAS so that each agent interoperates with other agents in and outside its own factory.[12] AARIA uses a mixture of heuristic scheduling techniques: forward and backward, simulation and intelligent scheduling. A proposed *production reservation* approach (an alternative planning strategy, in which tasks are scheduled in the order they arrive at the system) has adopted the classical contract-net bidding technology. The IMS (Intelligent Manufacturing Systems) consortium has previously studied the role of multiagent systems and holons in manufacturing.[13] Table A shows the state of development and properties of selected systems and architectures.

References

1. V. Mařík and D. McFarlane, "Industrial Adoption of the Agent-based Technologies," to be published in *IEEE Intelligent Systems,* 2005.
2. N. Sadeh and M.S. Fox., "Variable and Value Ordering Heuristics for the Job Shop Constraint Satisfaction Problem," *Artificial Intelligence,* vol. 86, no. 1, 1996, pp. 1–41.
3. J. Kubalik, L.J.M. Rothkrantz, and J. Lazansky, "Genetic Algorithm with Limited Convergence," *Information Processing with Evolutionary Algorithms: From Industrial Applications to Academic Speculations*, M. Grana et al., eds., JCIS/Assoc. for Intelligent Machinery, 2005, pp. 216–235.
4. K.S. Decker and V.R. Lesser, "Generalizing the Partial Global Planning Algorithm," *Int'l J. Intelligent Cooperative Information Systems*, June 1992, pp. 319–346.
5. E.H. Durfee, "Distributed Problem Solving and Planning," Multiagent Systems: *A Modern Introduction to Distributed Artificial Intelligence*, MIT Press, 1999.
6. J.M. Swaminathan, S.F. Smith, and N.M. Sadeh, "Modeling Supply Chain Dynamics: A Multiagent Approach," *Decision Sciences*, vol. 29, no. 3, 1998, pp. 607–632.
7. P. Bečvář et al., "ExPlanTech/ExtraPLANT: Production Planning and Supply-Chain Management MultiAgent Solution," *EXP: In Search of Innovation*, vol. 3, no. 3, 2003, pp. 116–125.
8. N.M. Sadeh et al., "MASCOT: An Agent-Based Architecture for Dynamic Supply Chain Creation and Coordination," *Production Planning & Control*, vol. 12, no. 3, 2001, pp. 212–223
9. H. Van Brussel et al., "Reference Architecture for Holonic Manufacturing Systems: PROSA," *Computers in Industry*, vol. 37, no. 1, 1998, pp. 255–274.
10. F.P. Maturana, W. Shen, and D.H. Norrie, "MetaMorph: An Adaptive Agent-Based Architecture for Intelligent Manufacturing," *Int. J. Production Research*, vol. 37, no. 10, 1999, pp. 2159–2173.
11. M. Cutkosky, J. Tenenbaum, J. Glicksman, "Madefast: Collaborative Engineering over the Internet," *Comm. ACM*, vol. 39, no. 9, 1996, pp. 78–87.
12. H. Parunak, A. Baker, and S. Clark, "The AARIA Agent Architecture: An Example of Requirements-Driven Agent-Based System Design," *Proc. 1st Int'l Conf. Autonomous Agents*, ACM Press, 1997, pp. 482–483.
13. S. Bussmann, M. Jennings, and M. Wooldridge, *Multiagent Systems for Manufacturing Control: A Design Methodology*, Springer-Verlag, 2004.

Table A. Selected systems and architectures' state of development and properties.

System	Domain	Open	Lightweight	Standard	Transparency	Simulation	Development
Centralized	Planning, scheduling	No	No	No	No	No	Commercial systems
PROSA	Planning, scheduling	—	—	No	Yes	Yes	Architecture
ProPlanT	Planning, scheduling	Yes	Yes	No	Yes	No	Prototype
MASCOT	Supply chain management	Yes	—	No	Yes	No	Pilot
MetaMorph	Planning and control	—	No	KQML (Knowledge Query and Manipulation Language)	Yes	Yes	Pilot
AARIA	Control, scheduling	No	No	No	Yes	No	Architecture

Table 1. ExPlanTech components' state of development and properties.

Domain	Development	Open	Lightweight	Low cost	Standard	Transparency
Control	No	—	—	—	—	—
Planning, scheduling	System	Yes	No	Yes	Yes	Yes
Supply chain management	System	Yes	Yes	Yes	Yes	Yes
Simulation	Prototype	Yes	No	Yes	No	Yes

- *Lightweight and low cost.* Isn't computationally demanding, can run on various hardware, and optimizes reuse of the existing computational infrastructure.
- *Standard.* Uses standard interfaces to integrate new modules and functionalities (possible at runtime) with the existing infrastructure.
- *Transparent and tractable.* Provides transparent and tractable decisions.
- *Simulation and integration.* Links to both physical production machinery and simulation and planning algorithms using the same mechanisms.
- *Replanning and reconfiguration.* Facilitates local, efficient replanning and minimizes required manufacturing reconfiguration due to physical malfunctions.

Our multiagent technology answers the listed challenges to different extents. See the sidebar and Table A for a brief analysis of the available multiagent manufacturing systems and an explanation of how they fit the requirements. Table 1 illustrates the ExPlanTech multiagent system's relevant properties.

Architecture

The ExPlanTech framework adopts the ProPlanT multiagent architecture. It contains an approximately fixed number of nontrivial agents, each providing different system functionality—for example, planning, simulation, and user access. We built ExPlanTech on top of the Java Agent Development Environment (JADE, http://jade.cselt.it).

It was easy to use JADE, which allows rapid development of sophisticated and reliable *multiagent systems*. A predefined agent core with already-implemented control and message transport protocols frees the author of MASs from low-level programming and resource management. The designer can focus on high-level functions and easily build user-targeted application. Any application built on the JADE platform complies with FIPA interoperability standards (www.fipa.org) for implementing independent software agents. This feature facilitates easy integration of new and third-party components. We implemented JADE (and thus the whole ExPlanTech system) in Java 2, which gives it platform independence and openness. Agents can run on different platforms (MS Windows, Windows CE, Linux, and even programma-

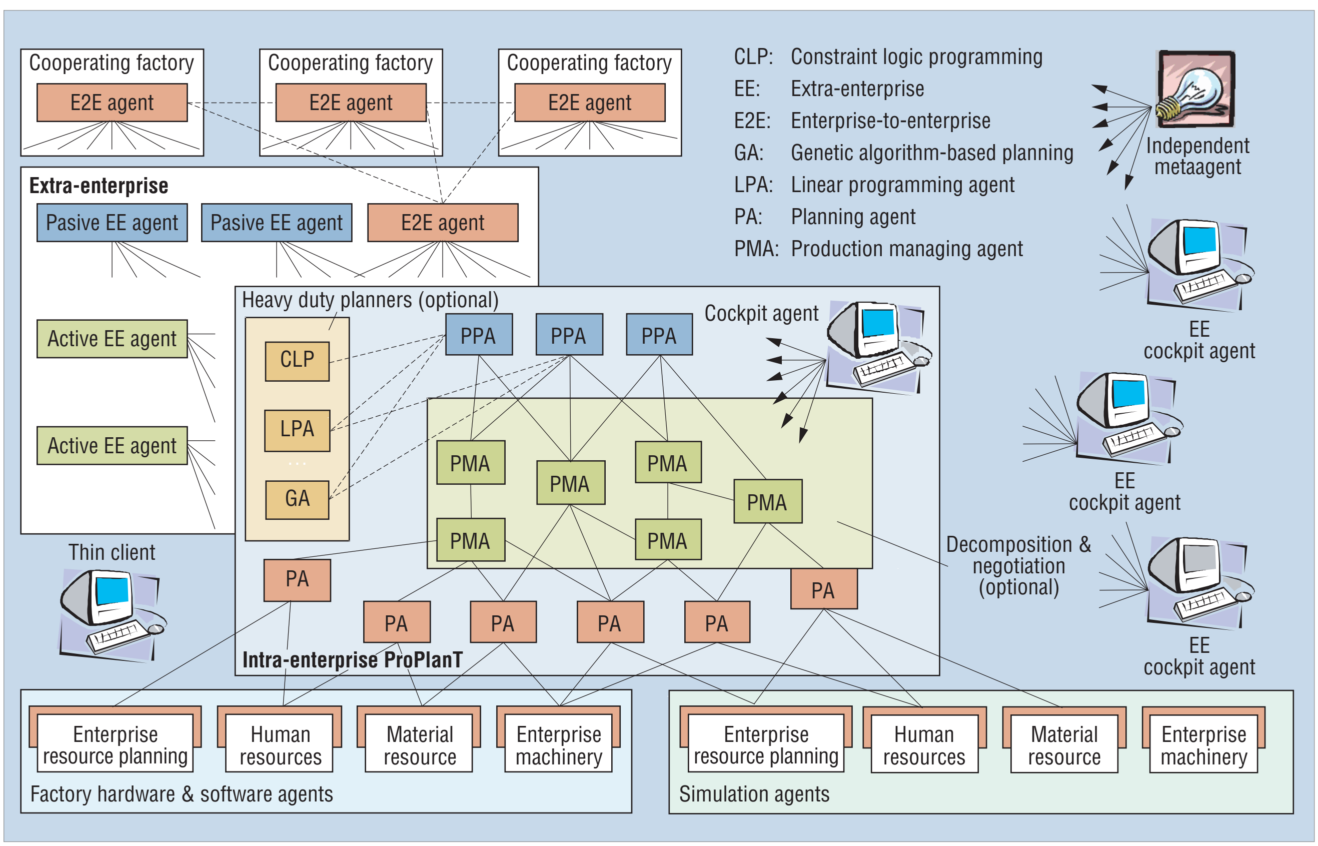


Figure 1. The ExPlanTech architecture.

ble logic controllers) and cooperate without worrying about low-level, platform-specific problems. We've developed an appropriate ontology for semantic interoperability in the manufacturing domain in ExPlanTech.

Planning agents

The core of the any ExPlanTech-based system is a community of appropriate planning agents (see Figure 1). A planning agent makes production plans for individual orders, taking care of conflicts and managing replanning and plan reconfiguration. Planning is implemented by the production planning agent, which primarily focuses on product configuration and quotation using one or a community of PMAs (production managing agents). PMAs plan production by task decomposition and partial-order planning. Additionally, you can develop various existing AI planning engines to handle different types of production—for example, linear programming, constraint logic programming, or genetic algorithm-based planning.

Resource agents

Typically, many resource agents running in the system directly interact with a planning agent and carry out data gathering and specific data preprocessing. ExPlanTech features two types of agents for integrating or representing manufacturing resources (see Figure 1). These agents

- Integrate a factory's hardware & software systems (for example, creating a bridge to a material-resources-provision (MRP) system, or integrating PLC controllers)
- Simulate a specific machine, workshop, or department (for example, a computer numeric control machine or a computer-aided design department).

Cockpit agents

Several different users could want to interact simultaneously with the planning agent. To allow this and control possible conflicts, we developed the cockpit agent (see Figure 1). Cockpit agents offer a user-friendly way to view the state of production processes, plans, given resource loads, and so on. Cockpit agents also let users interact with the system and, according to access rights, change plans or resource parameters (see Figure 2).

Extra-enterprise and enterprise-to-enterprise agents

Although we designed cockpit agents for use inside the factory (and its security firewalls), extra-enterprise agents let authorized users access the system from outside using a thin-client technology (see Figure 1). An EE agent has made the ExPlanTech system accessible (through a secure connection protocol) via a Web browser, PDA device, or WAP (Wireless Application Protocol)-enabled phone to remote users. An enterprise-to-enterprise agent makes the system accessible to the external software systems, such as remote cockpit agents or E2E agents at cooperating factories or material resources suppliers.

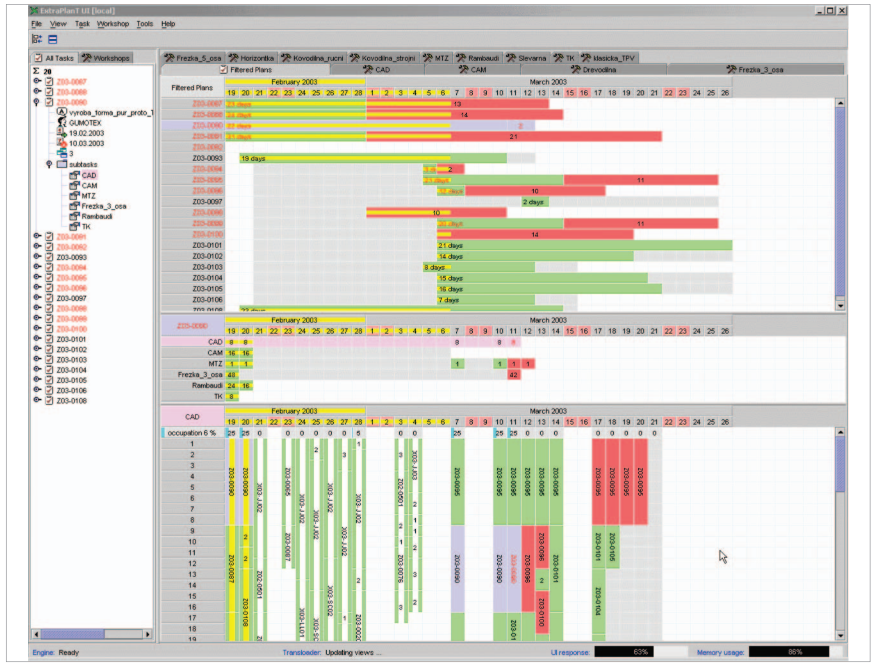


Figure 2. The cockpit agent-based graphical user interface.

Metaagent

We deployed the metaagent at the intra-enterprise and extra-enterprise levels. It carries out sophisticated methods of metareasoning to independently monitor information flow among the agents and to suggest possible operation improvements (such as workflow bottlenecks, inefficient or unused production components, and long-term performance measurements.)

Agent coordination and negotiation

ExPlanTech's key concept is the agentification of existing and new software components. Our system has two levels of software integration: *interaction* and *social*. Interaction integration builds the interaction wrapper (provided by JADE's special class `agent`) that acts as an interface between the agent's body and other agents. Interaction integration also translates messages between the FIPA ACL (Agent Communication Language) and the agent's internal language that invokes its behavior.

More interesting, however, is ExPlanTech's social integration. To efficiently collaborate, the agents need to collect knowledge and data about the other agents with which they may collaborate—we refer to these sets of agents as an agent's *monitoring neighborhoods.*[1] This type of knowledge, often referred to as *social knowledge*, is located in the agents' *acquaintance models*. We developed different acquaintance models for each type of agent. The cockpit agent, which only visualizes the information provided by the planning agents, doesn't need a rich acquaintance model, while the planning agents need sophisticated acquaintance models containing rich social knowledge to provide efficient distributed plans in a timely way.

Distributed planning aims mainly to divide the task into several relevant subtasks (often selecting one of many options) and then subcontract these subtasks to collaborating agents. This is a very complex activity that can't necessarily guarantee a global optimum. Without a precompiled social knowledge, a planning agent must initiate a *contract-net-protocol* (CNP) for every admissible decomposition. In complex situations this is almost impossible. Social knowledge stored in the acquaintance models circumscribes the space of possible decompositions and contracts.

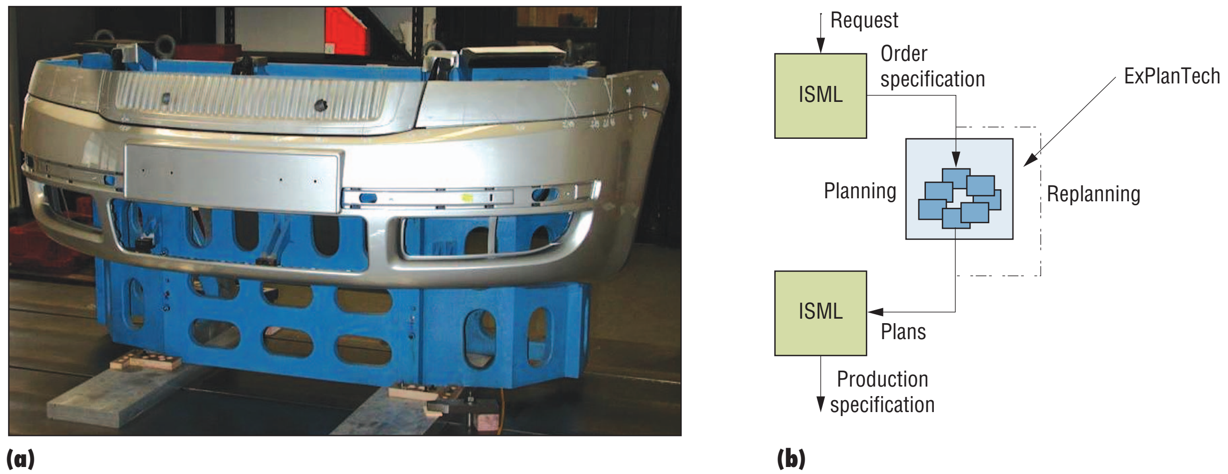


Figure 3. Example of (a) the Modelárna Liaz product and (b) ExPlanTech's role in the production-planning process.

In practical applications, the concept of the *tri-base acquaintance model* developed for project-driven production planning proved to be useful. This model collects social knowledge in three separate bases:

- *Cooperator-base.* Contains static information (a *white page list* comprising physical information about locations, IP address, and ACL encoding and a *yellow page list* with information about provided services) about the other agents.
- *State-base.* Administers nonpermanent information about the agents in the monitoring neighborhood (operational load, implementation state of various tasks, and trust in competing communities).
- *Task-base.* Contains a list of all currently planned tasks and the *decomposition templates* for new task decomposition if a specific requirement arrives.

When planning a task, the planning agent selects the most optimal task decomposition template in the task base and instantiates the template with the cheapest (or fastest) cooperators in the state base. It further contracts parts of the constructed plan to the cooperators, using the cooperator base's information.

Maintaining the social knowledge represent the principal bottleneck of deploying acquaintance models in real applications. The more useful the acquaintance model the more its data must be kept up-to-date. This becomes very costly in complex agent systems. Several different approaches exist to maintenance, ranging from *periodical revisions*, in which the planning agent periodically asks cooperators about the information's validity in the state base, to *subscription-based interaction*, in which the planning agent subscribes to cooperators for the relevant data.

In *decomposition-based planning*, a permanent or semipermanent hierarchy of agents exists, in which each agent decomposes a task into subtasks and coordinates its completion. By contrast, *fully autonomous planning* relies on agents working together on the same planning problem. They form their local plans, which are later merged (for example, by negotiation and voting), and replanning (for example, Partial Global Planning[2]) resolves conflicts. With *backward-chaining planning*—a compromise between decomposition-based planning and fully autonomous planning—the request backpropagates through the manufacturing flow. This doesn't have a command-and-control hierarchy or a central component, but the agents autonomously push requests for their prerequisites

In ExPlanTech, we used decomposition-based planning mainly for production planning and supply chain management. We used backward-chaining planning primarily for simulation purposes. We haven't widely adopted the concept of fully autonomous planning.

In both the fully cooperative and self-interested (that is, competing) agent communities, their negotiation methods let agents reach an agreement. Although these methods focus primarily on supply-chain management and virtual enterprise organization, we've used negotiation based on a classical CNP also in the intra-enterprise environment. Negotiation is also used for autonomous replanning, especially in domains in which the planning specification changes frequently.

Possible use cases

These use cases represent the most usual ways to use an ExPlanTech-based system.

Production planning, dynamic replanning

The most obvious use case is intra-enterprise production planning. ExPlanTech provides sets of linear and nonlinear plans and schedules in-house manufacturing activities so that the requested orders and tasks are achieved while optimizing enterprise resources. Given fixed deadlines, the system gives the user resource requirements and an appropriate manufacturing schedule. If the available resources are insufficient to meet the deadline, the system notifies users and initiates a supervised process of replanning and rescheduling. Replanning in ExPlanTech often occurs when the planning problem dynamically changes (for example, if the manufacturing machinery malfunctions). So, replanning solves existing or potential conflicts in production plans. ExPlanTech provides sophisticated tracking of interdependence among particular tasks, which makes replanning very fast and avoids planning again from scratch.

ExPlanTech continually analyses production data to give feedback to the project planner and keep plans up to date. Users can change task specifications and resources capacities at any time, and it recomputes and displays new plans in real time. *Variant planning* lets users examine several possible orders, test their feasibility, and choose the best one.

Supply chain management

Solving the complicated task of automated supply chain management requires overcom-

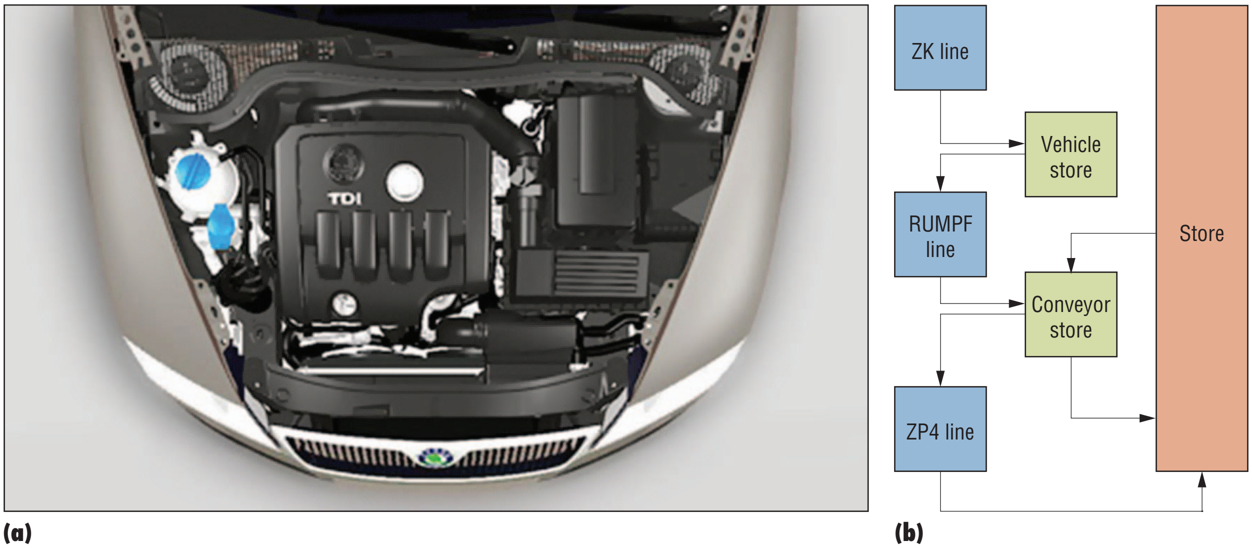


Figure 4. Example of the SkodaAUTO (a) product and (b) motor manufacturing process.

ing many technical and commercial difficulties. Unlike in intra-enterprise planning, ExPlanTech lacks complete knowledge about a supplier's parameters and capacities. This is why the simplest interaction approach (master-slave) doesn't suffice and ExPlanTech offers classical auctioning techniques (such as CNPs). Additionally, ExPlanTech handles secure and authenticated communication through the X-Security component and uses acquaintance models to handle temporary supplier inaccessibility. For supply chain integration and management, ExPlanTech provides EE agents, E2E agents, and MRP agents.

Simulation

Simulation can support decision-making in two ways. First, users can simulate a new factory or an overhauled or upgraded existing factory. The simulation tool supports a high-fidelity analysis of what an alternative's performance will be. Second, simulation can aid decision support in factory control to test how key machines' performance changes would affect the manufacturing process. ExPlanTech's integrated simulation environment lets users simulate different manufacturing scenarios to make technology changes and control safer.

EE access

Users can implement EE access either by a thin-client technology that requires an appropriate browser on the client side or by a thick client technology that assumes installation of software based on Java and JADE technology on a user's computer. Remote users (according to access rights) can exploit the functionality, which ranges from a passive observation of the system to active interventions (for example, planning custom orders).

ExPlanTech deployed

We've collaborated with several industrial partners to deploy ExPlanTech. They didn't all use an identical collection of the software system, so we customized solutions for each.

Modelárna Liaz

Modelárna Liaz is a mid sized pattern shop enterprise in the Czech Republic. The enterprise's customers are mainly from the automotive industry in the Czech Republic, Germany, and Belgium. The pattern shop specializes in single-part production of pattern equipment; permanent molds and dies; measuring and gauging devices; and welding, cooling, positioning, and machining fixtures and cubings (see Figure 3a).

The enterprise adapted ExPlanTech on the planning level. It aimed to improve medium- and long-term horizon efficiency. The important criterion was the load of strategic departments (machines) and delivery times. It implemented multiagent decomposition-based planning within the ExPlanTech. ExPlanTech agentified the factory information system and updated resource agents with real-time production feedback. One planning agent is responsible for the whole planning course. ExPlanTech implemented several cockpit agents for parallel connection to the system.

Once the enterprise resource planning system (denoted in Figure 3b as ISML, Information System in Modelárna Liaz) receives the order specification, ExPlanTech produces a complete set of production plans that are reshipped to the ERP system. Planning a new order or a change in the factory shop floor (represented by the resource agents) triggers a replanning process of all precommitted plans in ExPlanTech.

Besides production planning, ExPlanTech supports factory management with EE access to the planning data and automation of its supply chain management. The complete solution helps to find more efficient intra-enterprise plans and improve EE activities. The faster and more precise cooperation with suppliers and selling free capacities can shorten the production lead-time and create higher use of the factory. After several months of testing, the system proved its potential by improving machine use by 30 percent and reduced the finished product's due time by 5.3 percent.

SkodaAUTO

The SkodaAUTO motor factory, in collaboration with Gedas and CertiCon software companies, has successfully applied the ExPlanTech technology to design a robust planning system for car engines manufacturing. This exemplifies high-volume production, in which a few thousand engines (see Figure 4a) are manufactured daily. A high variability exists in the types of motors to be manufactured. The planning system needed to provide us with hourly plans for a period of six weeks. The production process (see Figure 4b) involves three production lines (ZK, Rumpf, ZP4) and two different parts buffer stores (vehicle and conveyor) and the main store for the finished products.

The agent technology provided a great help in solving the highly complex problem of planning assembly line production. We designed planning to occur on two independent levels:

- On a higher level, ExPlanTech produces a rough plan. This plan specifies an approximate amount of engines to be produced each day so that all the requested constraints are met. We have used a linear programming based heavy-duty agent for elaborating this higher level plan.
- On a lower level, the agents (each representing a line or a buffer store) analyze the provided higher-level plan and check for conflicts. In an ideal situation, the amount of conflicts is reasonable so that agents can negotiate and solve conflicts by swapping the tasks within days. The lower-level planning algorithm primarily performs conflict resolution by negotiation. The lower-level planning also provides daily ordering of the tasks.

Behr

The use case for Behr, an automotive supplier in the field of cooling and air-conditioning systems, employs mainly ExPlanTech's production simulation, supported by simplified planning and special cockpit agents and metaagents. The simulation aims primarily to compare the long-term effectiveness of several shop floor layouts. The simulation also lets Behr find production bottlenecks and optimal product buffer positions and evaluate the impact of important machine failures. These results are very important in decision support during the design of new or reconfiguration of an existing factory or even during important control decisions. Behr carried out adoption of ExPlanTech within the MPA (Modular Plant Architecture) project funded in part by the European Commision.

ExPlanTech's—and that of agent-based technologies in general—main virtue is in its high level of integration openness. When implementing an ExPlanTech-based, production-planning system, it's possible to reuse most of the previously existing IT infrastructure and software equipment as well as to integrate new decision-making support modules. At the same time, users have noted the system's high level of decision-making transparency and ability to involve human experts in the planning process.

In any of the listed deployments, the agent-based solution doesn't guarantee to provide optimal solutions with fewer computational resources (such as time and memory) than classical AI planning systems. This is one of the most common disappointments that agent-based technology adopters experience. However, the multiagent paradigm's ability to combine distributed AI algorithms with heavy-duty problem solvers and heuristic knowledge of the planning problem provides sophisticated solutions in specific cases.

JADE-based systems such as ExPlanTech require substantial amounts of computational resources to run promptly. As we have always developed a limited number of agents (up to 30), with any of the deployed applications, this hasn't been a problem. Scalability of the JADE-based software system has proven to be a bottleneck (especially in situations where several agent-based systems are working with the same hardware or software resources). A server-based version of the ExPlanTech system was developed to overcome this obstacle. Users access the system via a community of lightweight cockpit agents or from Web browsers by means of EE agents.

Currently, at Gerstner Laboratory, we're investigating in the concept of lightweight agent platforms that would allow massive scalability in the number of agents. We developed a novel lightweight platform—A-globe (http://agents.felk.cvut.cz/aglobe)—which recently received the System Innovation Award at a prestigious Cooperative Information Agent (CIA) workshop at the NetObject Days event in Erfurt. ■

Acknowledgment

We prepared this article under the support of the European Union Networking's Innovative Production Machines and System project and the EU Integrated Project, European Collaborative Networked Organisations Leadership (ECOLEAD).

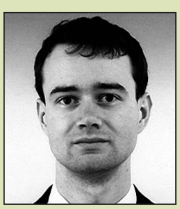
References

1. M. Pěchouček, V. Mařík, and O. Štěpánková, "Role of Acquaintance Models in Agent-Based Production Planning System," *Cooperative Information Agents* IV, vol. 1, Springer-Verlag, 2000, p. 179–190.
2. K.S. Decker and V.R. Lesser, "Generalizing the Partial Global Planning Algorithm," *Int'l J. Intelligent Cooperative Information Systems,* June 1992, pp. 319–346.

The Authors

Michal Pěchouček is the head of the Agent Technology Group at Gerstner Laboratory, a senior lecturer in artificial intelligence at Czech Technical University, and a part-time senior consultant for CertiCon. His research interests include agent-based computing, industrial deployment of agents, and social reasoning. He's the industry action coordinator for the AgentLink management committee, head of the EUMAS advisory board, and cochair of several international conferences, such as the industry track of the International Joint Conference on Autonomous Agents and Multiagent Systems 2005. He is a principal investigator on several research projects with direct defense and manufacturing industry involvement. He received his PhD in AI from CTU. Contact him at Gerstner Lab., Czech Technical Univ. in Prague, Technická 2, 166 27 Prague 6, Czech Republic; pechouce@labe.felk.cvut.cz.

Jiří Vokřinek is a researcher in the Agent Technology Group at Gerstner Laboratory. His research interests include artificial intelligence, multiagent systems, manufacturing planning and replanning, and virtual organizations. He's involved researching advanced agent-based technologies for supporting virtual organizations (IST Integrated Project ECOLEAD) and agent-based production processes simulations (CONCEERN, *Con*ex *C*entral *E*uropean *E*lectronics *R*ecycling *N*etwork). He received a university degree in technical cybernetics from CTU. Contact him at Gerstner Lab., Czech Technical Univ. in Prague, Technická 2, 166 27 Prague 6, Czech Republic; vokrinek@labe.felk.cvut.cz.



Petr Bečvář is a researcher at the Centre of Applied Cybernetics and a project manager at CertiCon. His research interests include process classification, expert systems, technical diagnostics, and intelligent human-computer interfaces. He received his PhD in technical cybernetics from the University of West Bohemia. Contact him at CertiCon, Václavská 12, 120 00 Praha 2, Czech Republic.