Autonomic Machine Control - A Case Study (Short Paper)

Christian Dannegger
Whitestein Technologies GmbH
Spitalstrasse 15
78166 Donaueschingen, Germany
Phone +49 771 8987 0
chd@whitestein.com

Dominic Greenwood
Whitestein Technologies AG
Pestalozzistrasse 24
8032 Zürich, Switzerland
Phone +41 44-256-5000
dgr@whitestein.com

ABSTRACT
This paper describes an autonomic machine control system applied to the adaptive control of a modular soldering machine. The particular case concerns the creation of a novel modular production machine with an integrated distributed agent control system, which will be sold worldwide from mid 2008. The agent model is described in terms of the specific customer requirements and specific advantages of the approach discussed.

Categories and Subject Descriptors
I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems.

Keywords
Production machines, soldering, software agents, autonomic, control system, optimization.

1. INTRODUCTION
Efforts to increase the flexibility of production lines are a pivotal trend in manufacturing. Whole assembly lines as well as individual machines are increasingly subdivided into modules in order to adapt precisely and just in time to constantly changing specifications (quasi make-to-order). An instance of this trend can be also found in micro production.

However, the centralized, 'hardwired' design of traditional control software imposes limits on successfully dealing with unpredictability. It is thus necessary to choose a novel approach to the development of control software, such that it is capable of managing modular machines dynamically and with minimal manual intervention while automatically maximizing the throughput and by that optimizing the investment into production resources.

This allows a production line to continuously adapt to changing boundary conditions and order specifications. Such a control system stands out due to its superior flexibility and adaptivity and drives the automation and optimization of modern production lines further – while at the same time embracing the rising complexity and dynamics of their environment.

Cite as: Autonomic Machine Control – A Case Study (Short Paper), Christian Dannegger, Dominic Greenwood, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008) – Industry and Applications Track*, Berger, Burg, Nishiyama (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 19-22.

An innovative offering in this area is a key differentiator for all vendors and users of modular production lines. Whitestein's product the LS/AMC – Autonomic Machine Control – makes use of these principles and applies them in the modular machine control market.

The particular case discussed is a concrete industrial application that will enter live production in mid 2008 and be further offered as a solution to the general market.

2. LS/AMC PRODUCT KERNEL
The foundation for the customer and machine specific solution is the generic LS/AMC product kernel providing the following features and functionality:

The agent platform LS/TS is used as runtime environment for the agent based solution.

The agent principle implies distributed autonomic control for each resource or entity in the system.

The agent type framework allows a jump-start for the solution building as it provides all general agents and templates for the application specific agents.

User, roles and rights management is needed in every multi-user environment. New functions can easily be put under the generic access control.

The built-in standard *CANopen*[1] interface allows fast integration of every *CANopen* compliant controller devices. LS/AMC has implemented a generic interface to *CANopen* to give each agent transparent access to its sensors and actuators.

3. CASE ENVIRONMENT
The particular application case of the LS/AMC control system is a modular soldering machine wherein each module is governed by an independent local agent controller. Coordination of the individual module operational parameters and the transition of boards from one module to another are the key control aspects.

[1] CAN = Controller Area Network; CANopen is a communication protocol and device profile specification for embedded systems based on CAN.

3.1 Machine Setup

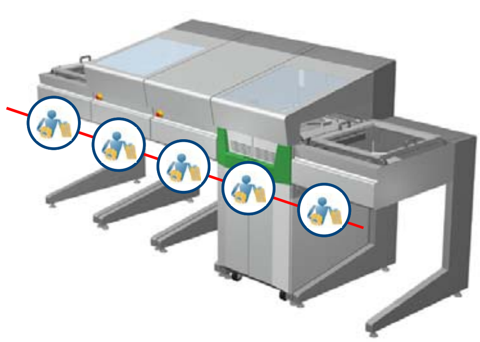
After many years of successful soldering using a conventional monolithic machine, the project team decided to prepare for the future by initiating a redesign of the centrally controlled machine (Figure 1) as a novel modular approach employing distributed control (Figure 2).

Figure 1. Old machine design: The processing units (modules) are contained within one static monolithic block and centrally controlled.

The modular setup of the new design not only allows configuration of the machine according to the customer's needs, but also has a separate local control within each module. The single drive for the one and only conveyor belt has also been replaced by a conveyor and a drive per module. This gives broad processing flexibility, as the target market for this machine typically requires changing production programs in real time.

A typical machine setup is composed of a feeder, a fluxer, 1-3 heaters, a soldering wave module and a cooler module.

Figure 2. New machine design: Each module of the machine is controlled by its own agent.



3.2 Sensors and Actuators

Each machine module has several sensors and actuators connected to the local controller board. There are digital switch sensors such as "end-of-belt", "zero-position", "emergency-stop" and "liquid level" as well as linear sensors including "temperature" and "encoder" of step motors. Actuators comprise motors, pumps and heaters as well as fans and signal lights. Overall a small standard configuration with 5 modules already contains around 40 sensors and 50 actuators, which must be managed and coordinated.

3.3 Customer Requirements

The goal of using agent technology in this project is to minimize the complexity of development, operation and maintenance of machines, without reducing the degrees of freedom for future application scenarios. Specifically, this implies:

3.3.1 Autonomic equipment adaptation

The control software of a modern production line must autonomically adapt to the ideal equipment configuration for each order. This effectively eliminates the need for manual reconfiguration. It also ensures that future enhancements of the system remain possible with only minor outlay.

3.3.2 Dynamically varying solder programs

Typically this machine is used for batch-size-1 tasks, which means that each and every board is processed with different soldering parameters and the boards are processed in parallel, i.e., pipelined.

3.3.3 Dynamic performance optimization

The capability to dynamically optimize capacities with changing configurations and target values is a top priority. This ensures maximum throughput and minimizes idle capacities and quality failures.

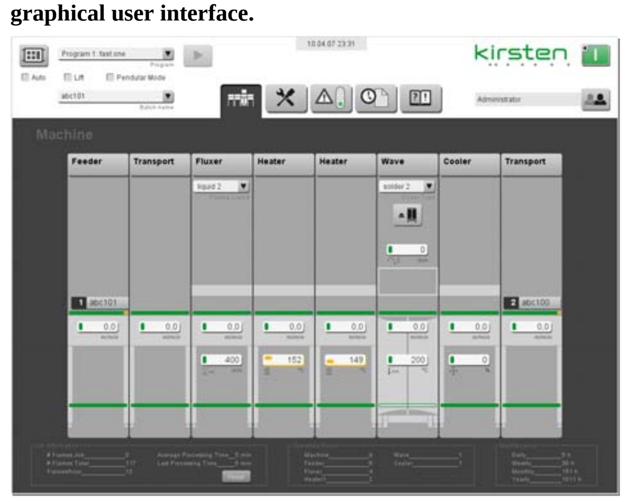
3.3.4 Seamless integration capability

At the macro level it is required that the control software for modular production lines such as this offer standard interfaces to integrate into a total production control system.

3.3.5 Intuitive user interface

Not least, such an advanced solution also needs to provide an intuitive user interface, which automatically adapts to the actual machine setup (Figure 3). It offers simple controls for the machine operator, extended functionalities for specialists and technicians, and comprehensive remote maintenance capabilities via the Web.

Figure 3. The modularity of the machine is reflected on the graphical user interface.



4. SOLUTION DESIGN

4.1 Existing solutions for agent based control

Whitestein Technologies has applied agent based distributed control in many related domains throughout recent years. The following examples all make use of multi-lateral negotiation algorithms to continuously seek optimal solutions:

4.1.1 Production Scheduling

The resources in a production environment including personnel, machines and materials are represented by software agents that use negotiation algorithms (e.g. auctions) to offer and “sell” their capacity to bidding orders, which are also represented by agents.

4.1.2 Road Logistics

To automate the creation of dispatching plans for transportation logistics systems each resource (vehicle) is represented by an agent, which coordinates and exchanges loads with others by making use of bilateral negotiations.

4.1.3 Supply Networks

All the players in a supply network continuously need to coordinate their demand forecasts and capacity availability. Agents can assist in this time consuming and time sensitive task perfectly. Monitoring agents along the supply chain fire an alarm and trigger activities if the reality deviates too much from the plan.

4.2 From monolithic to modular control

As in the previous examples, the LS/AMC soldering machine solution uses modular control principles because each module not only needs coordination with neighbor modules but also needs local – autonomic – control to optimize the overall process. For example, the heater module must maintain the temperature within tolerance limits irrespective of environmental changes caused by a board running through the module or a user opening a lid. Each module must thus combine its local control tasks with overall process coordination.

Modular control also means that each module holds its own production schedule and is able to give a production forecast in a backward-chain manner to enable the feeder module to estimate when best to start a new board. The module agents combine this production planning part with the real-time control when a board physically appears and when target temperatures are reached in reality.

4.3 Agent Model

Besides an agent type per physically available module type, the agent model (Figure 4) comprises one agent per order (board to be soldered) and some administrative agents for user management, configuration management and client communication.

The following are the core features of the implemented agent model:

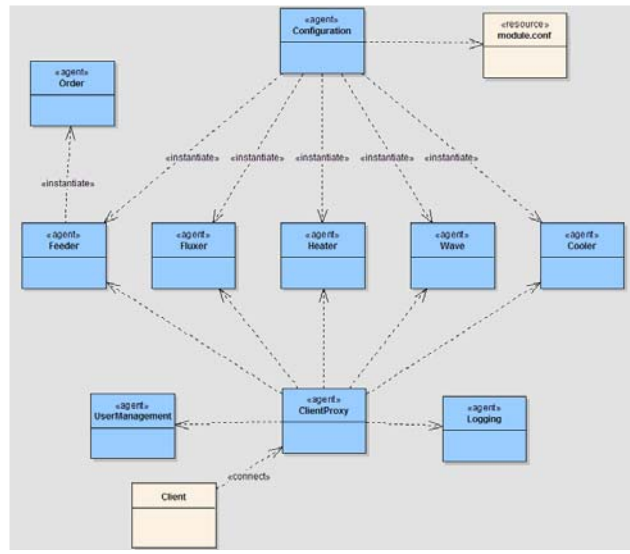
4.3.1 Autonomic module control

Every machine module is represented by a specifically adapted software agent that optimizes the module’s operations and capacity utilization.

4.3.2 Superordinate coordination

Through permanent bilateral negotiation and coordination between neighboring modules (i.e., of their software agents) the system constantly reaches a state of superordinate coordination. This eliminates the need for a central control instance.

Figure 4. The agent model



4.3.3 Self-managing orders

As for every module (resource), software agents also are responsible for the control of each production unit (order). They self-manage the order’s progress through the machine(s) autonomously and ensure that all requirements relating to (cost-) efficiency, speed and quality are optimally satisfied.

4.3.4 Distributed communication

The decentralized approach based on bilateral communication allows for virtually unlimited scaling possibilities, while at the same time increasing robustness against malfunctions and various external influences.

4.3.5 Standards compliance

At the controller level we provide full support for the *CANopen* industry standard machine control and communication interface.

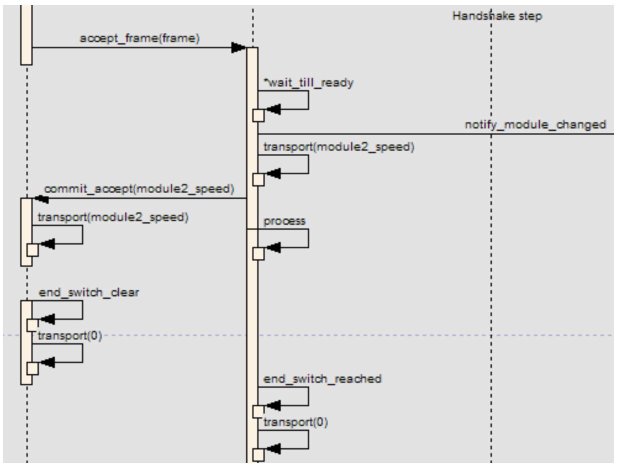
4.4 Interaction Model

One of the core principles of this solution is to dynamically create one agent per module detected on the *CANbus* and establish a communication link to the two neighbor agents. Consequently there is no global communication among the agents but only the one on the “left” and the “right” side.

This bi-lateral communication model is very lean but still powerful enough to drive the backward scheduling and the real-time synchronization between the modules. Here is one example for this synchronization task:

As each board (production job) and each module has different processing parameters the conveyor belts typically run at different speeds. To ensure a clean handover from one to the next module, LS/AMC has implemented a communication protocol (Figure 5) following a notify-and-pull principle, where the “sender” stops and notifies the “receiver” and, as soon as ready, the “receiver” sets the receiving speed and then grants the “sender” to send at this speed.

Figure 5. Extract of the agent communication protocol: handshake step



5. ADVANTAGES AND BENEFITS

5.1 Flexibility

The modular and distributed architecture of LS/AMC’s agent system allows for the easy addition of new modules, without causing fundamental changes in the existing system architecture.

The introduction of new kinds of machine modules only requires the development of a new module agent, which can be integrated into the current system with minimal effort.

5.2 Autonomic Adaptivity

New modules or modules that are failing or in need of maintenance can be exchanged while the system is running. Moreover thanks to LS/AMC’s distributed system architecture and intrinsic feedback-based adaptivity, machine control is updated autonomically at run-time without requiring restart of the control software.

5.3 Maintainability

Compared to traditional procedural or purely object-oriented approaches, the agent-oriented design of LS/AMC offers the advantage of intuitively mapping the real-world production line and order structure 1-to-1. This makes the system better to understand and use, increases its durability, and improves its maintainability.

An agent system also supports the easy and targeted customization of logging routines at the process level. This ensures the availability of more helpful and efficient methods of error monitoring and analysis.

5.4 Simulation

Complex simulation scenarios are easy to develop with LS/AMC, since a realistic mirror of a production line is more straightforward to simulate than an abstract model. Many different machine states and process flows can be recreated quickly and realistically. This significantly reduces the cost of quality control and improves personnel training and product demonstrations.

5.5 Goal-orientation

The software agents employed in this solution explicitly represent their behaviour using partially conflicting logical goals. Order agents, for example, pursue the minimization of throughput time, and module agents have the goal of optimizing the modules’ resource consumption. With LS/AMC these goals do not block one another but rather dynamically coordinate toward achieving optimal overall performance.

5.6 Dynamic optimization

A production optimization program is coupled to each work item (board) and transitions together with it through the machine modules. Each module adapts to the particular program and dynamically anticipates parameter and control adjustments when appropriate. The program is linked to the individual order or batch and not tied to a central, fixed setting for the entire machine.

6. FUTURE DEVELOPMENT

- Integration into preceding and successive processing machines, e.g. AOI (Automated Optical Inspection), cleaning or packaging.
- Machine-controlled board loaded through new lift modules. This allows throughput improvement by allowing agents to influence the sequence of production, which is not the case when boards are loaded manually.
- Making use of optional surface temperature sensors to improve the control of the temperature curve directly on the processed board.

7. ACKNOWLEDGMENTS

Our thanks go to Kirsten Soldering AG, Switzerland, having given us the opportunity to prove the agent concept in another real-world environment.

8. REFERENCES

[1] Rimassa, G., Greenwood, D., Kernland, M. E., 2006. The Living Systems Technology Suite: An Autonomous Middleware for Autonomic Computing. Presented at the International Conference on Autonomic and Autonomous Systems (ICAS) July 19-21, 2006, Silicon Valley, USA

[2] Greenwood, D., Dannegger, C. 2007. An Industry-Proven Multi-Agent Systems Approach to Real-Time Plan Optimization. Presented at the Fifth Workshop on Logistics and Supply Chain Management October 3-5, 2007, Berkeley, CA, USA

[3] Zimmermann, R. 2006. Agent-based Supply Network Event Management. Whitestein Series in Software Agent Technologies and Autonomic Computing. A Birkhäuser book. ISBN: 978-3-7643-7486-0