Scholz-Reiter, B.; Hildebrandt, T.; Kolditz, J.: A modelling methodology for control processes of autonomous production systems. In: Monostori, L.; Ilie-Zudor, E. (Hrsg.): Proceedings of MITIP 2006. 8th International Conference on The Modern Information Technology in the Innovation Processes of the Industrial Enterprises, Budapest, Hungary, 2006, S. 317-322



# A MODELLING METHODOLOGY FOR CONTROL PROCESSES OF AUTONOMOUS PRODUCTION SYSTEMS

**Bernd SCHOLZ-REITER, Torsten HILDEBRANDT, Jan KOLDITZ**

Department of Planning and Control of Production Systems
University of Bremen
Hochschulring 20, 28359 Bremen
Germany
E-mail: {bsr,hil,kol}@biba.uni-bremen.de

**Abstract:**

Today enterprises are exposed to an increasingly dynamic environment. Last but not least increasing competition caused by globalisation more and more requires gaining competitive advantages by improved process control, within and beyond an enterprise. Autonomous control of logistic processes is proposed as a means to better face dynamics and complexity. Autonomous control means processes of decentralized decision making in heterarchical structures. This paper presents a concept for a modelling method for autonomous logistic processes based on the integration of approaches from business process modelling, knowledge modelling and agent modelling. The modelling method supports requirements analysis by specification of the system, its entities and their interaction.

**Keywords:**

autonomous logistics processes, process modelling, procedure model, production control, modelling method

## 1. INTRODUCTION

Today enterprises are exposed to an increasingly dynamic environment. Last but not least increasing competition caused by globalisation more and more requires gaining competitive advantages by improved process control, within and beyond an enterprise. Autonomous control of logistic processes is proposed as a means to better face dynamics and complexity.

Autonomous control means processes of decentralized decision making in heterarchical structures. It requires the ability and possibility of interacting system elements to autonomously make goal-oriented decisions. The use of autonomous control aims at achieving a higher robustness of systems and simplified processes achieved by distributed handling of dynamics and complexity due to greater flexibility and autonomy of decision making. The system elements, making their decisions autonomously, are the logistic objects within a production system themselves [12].

This paper presents a concept for a modelling method for autonomous logistic processes based on the integration of approaches from business process modelling, knowledge modelling and agent modelling. The modelling method supports requirements analysis by specification of the system, its entities and their interaction.

## 2. AUTONOMOUS CONTROL OF LOGISTIC PROCESSES

Autonomous control in the context of SFB 637, the research project this work is based on, means processes of decentralized decision making in heterarchical structures. It requires the ability and possibility of interacting system elements to autonomously make goal-oriented decisions. The use of autonomous control aims at achieving a higher robustness of systems and simplified processes achieved by distributed handling of dynamics and complexity due to greater flexibility and autonomy of decision making. Focus of the SFB lies in the areas of production and transport logistics, so the system elements, making their decisions autonomously, are the logistic objects themselves [1].

In order to enable logistic objects to be intelligent they have to be provided with smart labels. While today’s RFID (radio frequency identification)-chips have very limited capabilities in respect to energy, range, storage capacity and especially information processing [3], near future shall bring highly evolved smart labels that can provide resources alike micro computers to logistic objects. Nowadays RFID is already widely used in industry for identification matters and several visions for future applications exist [4], [5].

With respect to shades of autonomous control, different scenarios are possible, depending on which logistic objects are provided with smart labels and the functionalities they offer. This determines to what extend the logistic objects are able to make decisions. Considering the kind of decision-making by autonomous and therefore potentially intelligent logistic objects, transferring control decisions to goods, machines, storages and conveyors is obvious. Besides scenarios, where only one of the kinds of logistic objects has the ability to autonomously make decisions, arbitrary combinations are possible, depending on whether objects of the respective group are rather autonomously controlled or not.

Different logistic objectives can be assigned to the different groups of objects. For instance the objective of a high utilization can best be assigned to machines, while the objective of low due date deviation can best be assigned to a good. Concrete goal values are only achieved by the interaction of many logistic objects. Often conflicting goals of different objects have to be balanced, e.g. by negotiation. This leads to an increased coordination and communication effort compared to hierarchic forms of finding a decision. The more objects and groups of objects are involved in such a communication and make their decisions autonomously, the more important this point becomes. The number of possible communication relationships roughly grows quadratic in the number of participating objects. With 10 communicating objects there are 45 possible relationships, having 100 objects already leads to 4950. These numbers make clear that communication has to be limited to objects in the immediate spatial and/or logic neighbourhood as otherwise control strategies can only hardly be scaled to problems of a realistic size. All these points have to be considered designing a control strategy and for modelling such a system.

## 3. DEVELOPMENT OF A LOGISTICS SYSTEM BASED ON AUTONOMOUS COOPERATING PROCESSES

This chapter focuses on the development procedure of a logistic system based on autonomous cooperating processes and the integration of the proposed modelling methodology. The system development is formulated as an iterative process shown in figure 1. The steps 1 and 2 will be supported by the methodology for modelling autonomous logistic processes.

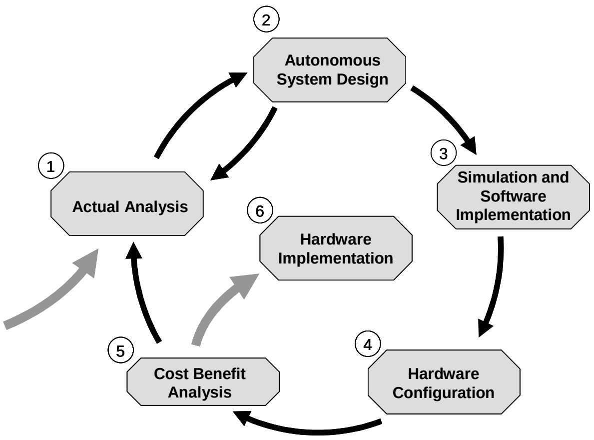


Figure 1: System development process

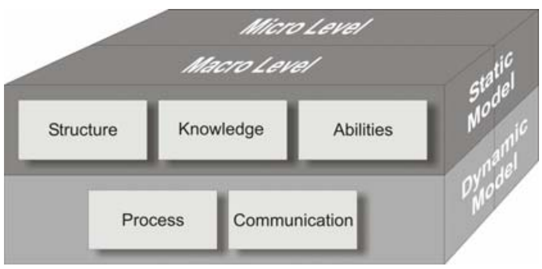


Figure 2: View Concept

Scholz-Reiter, B.; Hildebrandt, T.; Kolditz, J.: A modelling methodology for control processes of autonomous production systems. In: Monostori, L.; Ilie-Zudor, E. (Hrsg.): Proceedings of MITIP 2006. 8th International Conference on The Modern Information Technology in the Innovation Processes of the Industrial Enterprises, Budapest, Hungary, 2006, S. 317-322



*1. Actual Analysis* The starting point is an analysis of the actual state of the system. At the beginning this step is alike a feasibility study. That means very basic issues like an estimation to what extend autonomous cooperating processes are actually suited for the scenario. Associated with that the question has to be answered which objectives are pursued in the particular case by implementing autonomous logistic processes. If the application is not evaluated positively for this scenario the development procedure should be cancelled here. It has to be pointed out that this step forms a mini iteration with the following one. After the system design in step 2 concrete scenario data has to be collected and registered like machines, products or work steps. This step does not affect the degree of autonomy but is the connection between the in principle universal design of the system and its processes and concrete scenario data.

*2. Autonomous System Design* The next step consists of the design of the system. There a semi-formal specification of the proactive elements in an autonomous system as well as identification, design and allocation of decision processes are performed. It has to be clarified which elements are part of the system and which of them intelligent respectively autonomous entities are. To ensure the operability of the system all elements and processes have to be aligned with each other, making this step the basis of the development procedure.

*3. Simulation and Software Implementation* During the step of simulation and software engineering the design realised before is tested in a simulation first. Especially operability and impact on logistics performance of the whole system are focused here. A central task is the verification of required system behaviour because this is a necessary precondition for industrial application of emergent systems like autonomous logistics processes. The simulation code may already be part of the engineering process of the planned control software if the code is reusable. Otherwise the core software engineering process starts in a subsequent iteration loop.

*4. Hardware Configuration* On the basis of the ideas gained before an estimation of needed hardware equipment for the autonomous system (for example what kind of communication infrastructure) can be made, getting more detailed with every iteration loop. Conclusions may be drawn from the process model as well as from the simulation. For example from allocation of control processes and data packets to entities of the logistic system necessary memory and computing capacity can be derived. Another example is the prediction of the capacity and equipment of the communication infrastructure on the basis of the expected communication volume between logistic system entities resulting from the simulation and the physical distribution of the objects to be arranged during hardware configuration. Attention has to be paid to the fact that although several agreements have been done during the steps before, this step strongly impacts implementation costs.

*5. Cost Benefit Analysis* Now compared to the initial feasibility study a much more significant cost benefit analysis is possible getting more detailed during subsequent iteration loops. Thereafter the original process model can be adjusted in step 2 according to the new conclusions. In case of repeating negative results in this step an application of autonomous logistics processes has to be abandoned for this scenario.

*6. Hardware Implementation* The final step of the development process is the installation of the system based on autonomous logistic processes. In case of insufficient experience a prototypic setup should be tested before the actual installation.

## 4. MODELLING AUTONOMOUS CONTROL

### 4.1. Overview

Due to space limitations of the overall modelling conception only the view concept - to give a rough overview of the structure of and elements of a model -, an overview of the procedure model and an architectural sketch of the modelling tool to be developed can be presented in the following paragraphs. For the remaining parts of the modelling method, i.e. a detailed definition of general modelling principles, the definition of the meta-model, the graphical notation to be used and the reference model the reader is referred to [11].

Scholz-Reiter, B.; Hildebrandt, T.; Kolditz, J.: A modelling methodology for control processes of autonomous production systems. In: Monostori, L.; Ilie-Zudor, E. (Hrsg.): Proceedings of MITIP 2006. 8th International Conference on The Modern Information Technology in the Innovation Processes of the Industrial Enterprises, Budapest, Hungary, 2006, S. 317-322



### 4.2. Concept of Views

Creating process models usually leads to a high degree of complexity. A view concept serves as a means to reduce the complexity constructing a model [10] which is also reflected in the guideline of systematic design of a model. Our proposed view concept for modelling autonomous logistic processes is depicted in figure 2. A fundamental distinction can be made between a static and dynamic model. The static model describes the structure, the dynamic model the behaviour of the modelled system, according to the basic classification in UML [9] that is also appropriate here.

The *structure view* that shows the relevant logistic objects is the starting point. The basic elements for this view are UML class diagrams. Besides objects and classes the structure view can show relationships between them, for instance in the form of associations or inheritance relationships.

The *knowledge view* describes the knowledge, which has to be present in the logistic objects to allow a decentralized decision making. This view focuses on composition and static distribution of the knowledge while not addressing temporal aspects. For this purpose UML-class diagrams are sufficient, while for the just mentioned temporal aspects, a dedicated knowledge representation language would have to be used [13]. However it is doubtful how far the additional complexity in using it is compensated by the increased expressiveness. This is especially more important with respect to the intended use of the modelling method by a process expert.

The *ability view* depicts the abilities of the individual logistic objects. Control of a logistic system needs certain abilities, which have to be provided by the logistic objects. These abilities are supposed to be seen as collections of processes the logistic objects are able to perform.

The *process view* depicts the logic-temporal sequence of activities and states of the logistic objects. Here the objects' decision processes can be modelled. The process view plays a central role connecting the views of the static model and depicting the behaviour of logistic objects, so far only viewed statically. The notation elements used for this are activity diagrams as well as state diagrams. These two diagrams are also proposed in business process modelling using the UML [8].

The *communication view* presents the contents and temporal sequence of information exchange between logistic objects. Depicting the communication is especially necessary to depict the interaction of autonomously deciding, otherwise only loosely coupled objects to model their interaction [14]. To display the communication UML-sequence diagrams showing the interacting partners, the messages and their temporal progression as well as class diagrams to display communication contents are supposed to be used. This is inter-entity-communication is a macro-level interaction using the micro/macro-distinction that can for instance be found in agent-oriented software engineering [15].

### 4.3. Procedure Model

The procedure guiding the user during the process of model construction contains necessary steps, their results and the execution order of the steps. The procedure ensures a systematic modelling process and thus a successful and reproducible model construction and an adequate modelling result. It is a specific procedure model, giving operational recommendations using the modelling language sketched earlier. That way a user with detailed logistics knowledge is able to construct visual models to support analysis, design and improvement of systems based on autonomous logistics processes. In the following the procedure model is sketched with its main steps.

*1. Global Objectives* The first step in the design of autonomous logistics systems is the definition of the global objectives. For control processes in the production logistics domain the classic objectives of production planning and control are to be used and have to be prioritised for the concrete scenario. The objectives are short throughput times, high delivery reliability, low work-in-progress levels, and high utilization [7]. These objectives should be at least qualitatively prioritised and are the basis for the system modeller in the ongoing design process.

*2. Structure* The second step includes the collection and documentation of the relevant system entities. This step consists of two main components, namely the collection of the entities

Scholz-Reiter, B.; Hildebrandt, T.; Kolditz, J.: A modelling methodology for control processes of autonomous production systems. In: Monostori, L.; Ilie-Zudor, E. (Hrsg.): Proceedings of MITIP 2006. 8th International Conference on The Modern Information Technology in the Innovation Processes of the Industrial Enterprises, Budapest, Hungary, 2006, S. 317-322



themselves including inheritance on one hand and the definition of the associations on the other hand.

3. *Abilities* The third step broaches the issue of designing the abilities of the system entities. Abilities are connected to the logistic objects, that way defining which entities are able to perform which processes. Moreover the abilities themselves can be structured with generalisation- and uses-relationships and one or more process later on realise an ability.

4. *Top-level processes* In step four the top-level processes are modelled on a rather abstract level. First the focus lies on the more routine processes, ignoring unplanned events. After that the processes are refined to consider and handle disturbances and changes. During step four the sequence of activities and states of the single units as well as the mutual dependencies are focused. It is not a detailed explication of the decision situations intended.

5. *Decision situations and processes* This step focuses on the decision situations and the modelling of the decision processes. First the decisions have to be identified in the before modelled top-level processes. After that local objectives have to be derived for the logistics objects confronted with a decision. Now the decision process can be modelled in detail implicitly influenced by the local objective conducting the process design.

6. *Information Identification* After designing the decision processes every single situation has to be analysed in respect to identify the information that is needed. The aspect that a decision needs information to be based on was so far merely considered implicitly in support of a simplified view and has to be explicated now.

7. *Information sources* The identified information has to be allocated to the entities of the system. For the allocation of information the distribution of decision situations and control tasks can be used as an orientation. It might be reasonable to assign the information corresponding to the frequency of usage or to the location where the most up-to-date information can be assumed. Moreover limited storage capacity or other restrictions resulting from the overall system configuration might cause the necessity to adapt the information allocation.

8. *Communication processes* On the basis of the decision processes and the allocation of the information the modelling of the communication processes is done during this step. Thus the aspect is addressed that not every logistic object owns the information needed in a certain decision situation but has to acquire it from somewhere else. In a basic case the communication would be a simple request but more complex negotiations should also be widespread.

9. *Data input* In the final step of the modelling procedure the concrete scenario data is entered. The existing entities of the system belonging to the classes defined before have to be collected, to form the basis for subsequent simulation and in the end for the operability of the autonomous control system.

### 4.4. Tool Support

As already stated earlier, a software tool is part of the modelling method and its prototypic implementation is currently worked on. To achieve a high platform independence Java was chosen as a programming language. More specifically Eclipse is used as a so-called Rich-Client-Platform (RCP, [6]). Originally being a Java-development environment, it supports a large variety of platforms and its modular architecture and extendibility by a very flexible plug-in-mechanism makes it possible to easily create customized applications, either by extending existing functionality or just using necessary parts of the functionality. This makes it possible to offer multi-platform “rich” user interfaces (i.e. offering a user interface with much more possibilities e.g. a web-based application could offer), by using the relevant subset of Eclipse’s functionality.

Furthermore there are two software-frameworks making Eclipse very suitable for creating our modelling tool. These are the Eclipse Modelling Framework (EMF, [2]) and the Graphical Modelling Framework (GMF). Both together allow an easy creation of a graphical editor as is required as a user interface for our application. Both frameworks offer code-generators to automatically create large portions of the later modelling application. EMF for instance offers a modelling language similar in expressivity to class diagrams to allow the creation of meta-models (including XML-exchange formats for them) for a specific application and creates large fractions of the required source code to handle models of such a meta-model.

## 5. CONCLUSIONS

This paper addressed the topic of modelling autonomous logistic processes. Therefore after a short definition of autonomous control in the context of logistics, the overall system development process was sketched. After that the concept of our modelling method was presented, first giving a rough overview, then detailing selected aspects of it such as the view concept and the procedure model.

Further research will detail notations to be used and be concerned with the elaboration of the procedure model. Finally our work will result in the development of a software tool, specifically tailored to support our modelling method comprised of the notation and procedure model as far as possible. With the help of this tool a process expert (e.g. a logistics expert with only little background in computer science) will be supported in modelling and designing autonomous logistic processes.

## 6. ACKNOWLEDGMENTS

This research is funded by the German Research Foundation (DFG) as part of the Collaborative Research Centre 637 “Autonomous Cooperating Logistic Processes - A Paradigm Shift and its Limitations” (SFB 637).

## 7. REFERENCES

[1] Bauer, B., Odell, J., 2005: UML 2.0 and Agents: How to Build Agent-based Systems with the new UML Standard, Journal of Engineering Applications of Artificial Intelligence 18/2: 141-157.

[2] Budinsky, F. et al., 2003: Eclipse Modeling Framework: a developer’s guide, Addison Wesley, Boston, USA.

[3] Finkenzeller, K., 2006: RFID-Handbook - Fundamentals and Applications in Contactless Smart Cards Identification, Wiley & Sons LTD, 2. edition, Swadlincote, UK.

[4] Fleisch, E., Mattern, F. (eds.), 2005: Das Internet der Dinge - Ubiquitous Computing und RFID in der Praxis: Visionen, Technologien, Anwendungen, Handlungsanleitungen. Springer, Berlin, Germany.

[5] Heinrich, C., 2005: RFID and Beyond: Growing Your Business through Real World Awareness, Wiley Publishing, Indianapolis, Indiana.

[6] McAffer, J.; Lemieux, J.-M. (2005): Eclipse Rich Client Platform: designing, coding, and packaging Java applications, Addison-Wesley, Upper Saddle River, NJ, USA.

[7] Nyhuis, P., von Cieminski, G., Fischer, A., 2005: Applying Simulation and Analytical Models for Logistic Performance Prediction, Annals of the CIRP, 54/1:417-422.

[8] Oestereich, B., 2003: Objektorientierte Geschäftsprozessmodellierung mit der UML, dpunkt Verlag, Heidelberg, Germany.

[9] OMG: Unified Modelling Language Specification (version 2.0). http://www.uml.org 2005.

[10] Scheer, A.-W., 1998, Wirtschaftsinformatik – Referenzmodelle für industrielle Geschäftsprozesse, Springer Verlag, Berlin Heidelberg.

[11] Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T., (2006): Specifying Adaptive Business Processes within the Production Logistics Domain – A new Modelling Concept and its Challenges. In: Hülsmann, M.; Windt, K. (eds.), 2006: Understanding Autonomous Cooperation in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer (to appear)

[12] Scholz-Reiter, B.; Windt, K.; Freitag, M., 2004: Autonomous Logistic Processes - New Demands and First Approaches, Proceedings of the 37th CIRP-International Seminar on Manufacturing Systems 2004, 19.-21.5.2004, Budapest, Hungary, 357-362.

[13] Sowa, J., 2000: Knowledge representation: logical, philosophical, and computational foundations, Brooks Cole Publishing Co., Pacific Grove, USA.

[14] Weiß, G. (ed.), 2000: Multiagent Systems – A modern approach to distributed artificial intelligence, The MIT Press, Cambridge, USA.

[15] Weiß, G., and Jakob, 2005: R. Agentenorientierte Softwareentwicklung, Springer Verlag, Berlin Heidelberg, Germany.