Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

Specifying Adaptive Business Processes within the Production Logistics Domain – A new Modelling Concept and its Challenges

Bernd Scholz-Reiter, Jan Kolditz, Torsten Hildebrandt

Department of Planning and Control of Production Systems, University of Bremen, Bremen, Germany

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 the borders of producing enterprises. One possibility to face increasing dynamics is autonomous control of logistic processes. This shall allow more robust processes in spite of growing environmental as well as internal complexity.

This paper presents the idea of autonomous logistic processes and focuses on a concept for modelling such processes. It is structure as follows: the next section gives a short overview of the concept of autonomous logistic processes. Subsequently section *Development of a logistics system based on autonomous cooperating processes* presents the overall system development cycle. The main section *Modelling autonomous control* discusses process modelling under the paradigm of autonomy and starting with requirements introduces our modelling method, important aspects of which are thereafter presented in more detail. The paper is concluded by a short summary and an outlook of future work.

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

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

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 (Scholz-Reiter et al. 2004).

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 (Finkenzeller 2003), 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 (Fleisch and Mattern 2005; Heinrich 2005).

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 neighborhood 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.

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

Development of a logistics system based on autonomous cooperating processes

The development of an autonomous logistics system can be described on the basis of the Systems Engineering (Haberfellner et al. 2002) procedure model, as shown in figure 1. The methodical core is the iteration loop during main study and detail study phase. In the following the single phases of the procedure guiding through the development of an autonomous system and the connection with the more general procedure model of Systems Engineering are described.

1. Initiation

The rather unstructured initiation phase is triggered by sensing a problem and is completed by the decision to start a preliminary study. In our context this might be a problem associated with production planning and control or the assumption of a chance to improve the system's performance by adoption of autonomous cooperating logistic processes.

2. Preliminary Study

During preliminary study the objectives of adopting autonomous cooperating logistic processes have to be defined. Usually you will aim at concrete improvements in the fulfilment of logistic goals. In this regard the considered system and the scope of work have to be stated, for example a certain area of a production system.

Part of the preliminary study is also a situational analysis, which provides an overall understanding of the scope of work, of the existing problems and of the control processes. If required this system analysis can be detailed for certain aspects in later phases.

An important basis for the decision whether to continue or to abandon the project is an estimation of the impact of the solution principle. Therefore it is estimated to what extend an application of autonomous cooperating logistic processes is reasonable and promising. This shall allow a decision whether to start the main study phase or to cancel the project in respect to the development of an autonomous system.

The comparison of autonomous control and alternative methods and therewith the rating of such a solution principle under certain conditions is an issue of ongoing research that will not be discussed here any further (Scholz-Reiter et al. 2006a).

3. Analysis and Design Phase

In this paragraph the single steps of the iteration loop as the methodical core of the development process shown in figure 1 are described, followed by a discussion of the connections with the general Systems Engineering procedure.

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

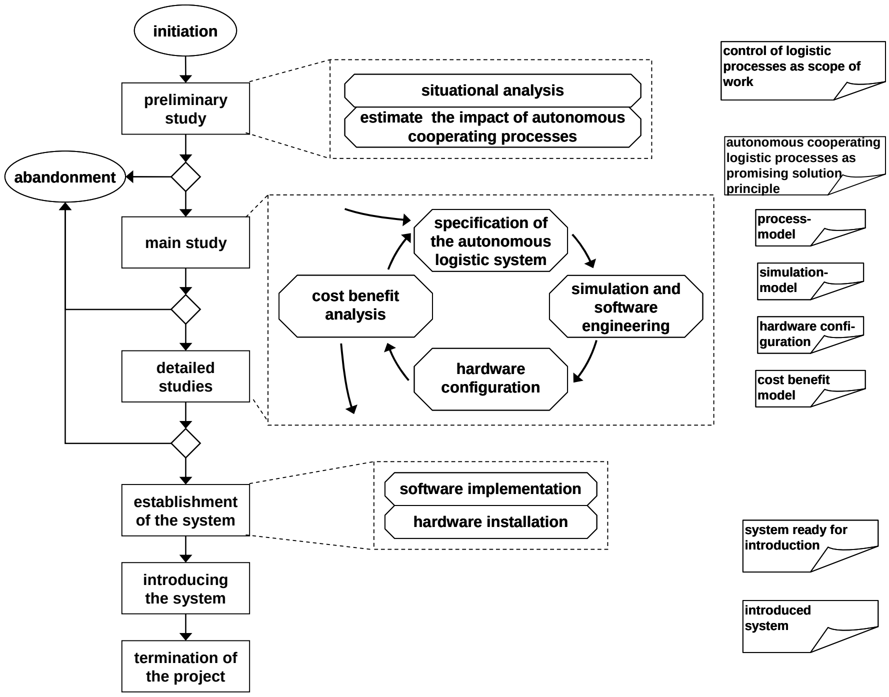

Fig. 1. Concretised Systems Engineering procedure model for autonomous logistics systems

- The first step of the iteration loop consists of the specification 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.
- 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 the implementation phase.

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

- 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.
- Every iteration loop is concluded by a cost benefit analysis. On the basis of the rating and subsequent decision the original process model can be adjusted 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.

The main study and design studies phases are not separated in two different ones, but combined in one phase. This phase is about an iteration loop, which on one hand serves the generation of variants and on the other hand produces more detailed and concretised solutions with proceeding iterations. A drawback of this abandonment of a phase separation is a lack of a clearly defined main phase, concluded by a decision about the cancellation of the project. But the cost benefit analysis concluding every iteration loop allows a decision about a cancellation on the basis of an economical rating in every cycle. This approach does not conflict with the Systems Engineering procedure because there is a close linkage between main study phase and detailed studies phase intended anyway. It is explicitly recommended to bring forward parts of detailed studies to the main phase if necessary.

Referring to the problem solving cycle as the micro-logic of Systems Engineering (Haberfellner et al. 2002), the focus of the main and detailed study phase lies in the search for solution and the selection (figure 2). The search for objectives primarily consists of an analysis of overall concepts and detailed concepts chosen before and the formulation of according objectives for the beginning iteration loop. The specification of an autonomous logistic system represents the synthesis of solutions, the constructive and creative activity. The simulation allows verifying the different solutions developed during the specification step concerning their functionality and capability and therefore represents the analysis of solutions. Afterwards on the basis of logistics as well as complementing objectives an

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

evaluation of the solutions that have been rated as basically suitable is done. In the hardware configuration step the different possible solutions are evaluated in respect of their feasibility concerning hardware-oriented aspects as well as of the anticipated implementation input. The cost benefit analysis provides a basis to economically evaluate and compare the solution variants. Thereafter the decision is made whether to detail and concretise a variant respectively to start the establishment of the system or to cancel the project.

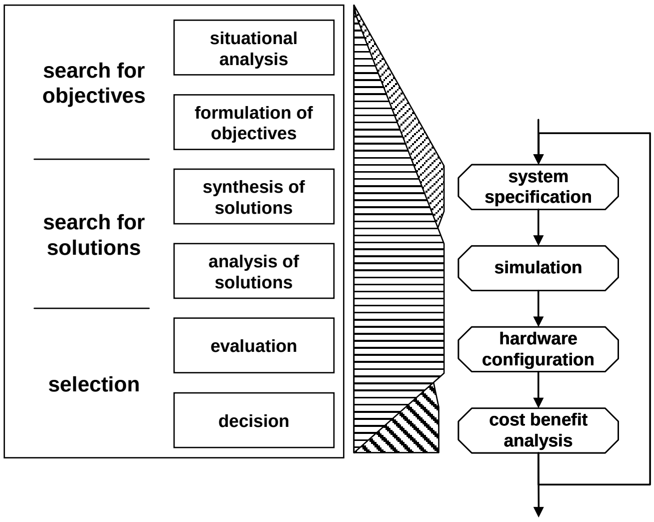

Fig. 2. Relevance of the steps of the problem solving cycle for the single steps of the iteration loop

4. Establishment of the system

During the establishment step the autonomous logistics system is realised. The main topics are the software implementation and the creation and integration of facilities and instruments. Ideally the software should be implemented using parts of the program code created during the simulation step.

5. Introduction of the system and termination of the project

Normally the introduction will involve huge and complex systems resulting in hard or even not calculable side effects. Therefore the introduction of the autonomous logistics system should be done stepwise if possible. After verifying the fulfilment of the objectives the system is handed over from the originating project team to the operating institution and the project is terminated.

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

Modelling autonomous control

Requirements to the modelling method

In this section requirements to the modelling method are formulated and structured following the distinction from requirements engineering of software sytems. Therefore the requirements and necessary characteristics of the modelling method resulting from the definition of Autonomous Logistic Processes are presented first. Subsequently general guidelines towards any modelling method are explained in the form of the Guidelines of Modelling (GoM, (Schütte 1998)), which also serve as general constraints for the modelling method presented here.

Primary requirements for the modelling method result from the fact that analysis and design of autonomous logistic processes has to be made possible for a logistic expert. Using the modelling method it therefore has to be possible to depict the constituent characteristics resulting from the definition of autonomous control given earlier in this book. The general definition results in the requirement that it has to be possible to model autonomous decision making of interacting system elements, i.e. a decentralised decision making in heterarchical structures. More specific requirements result from the specialisation of the general definition towards logistic processes which is relevant here. According to this concretised definition autonomous control of logistic processes is "[…] characterised by the ability of logistic objects to process information, to render and to execute decisions on their own" (Windt et al. 2007). A logistic object fulfilling this definition is called an intelligent logistic object; to support its design implicates an approach focused on these objects. The autonomous control characteristic of information processing requires a possibility to model information processing processes and that they can be assigned to the objects on which they are executed. Rendering of a decision entails possibilities to model the location of a decision, available decision alternatives and if necessary the knowledge needed by the intelligent logistic object for its decision. The characteristic of execution of its decisions finally requires an intelligent logistic object not only to render decisions autonomously, but also to initiate its execution and monitor its execution progress.

Furthermore the models created are the basis for subsequent software implementation. As a requirement this leads to the need to make this transition as frictionless as possible and already consider this during the design of the modelling method.

Following the distinction from the field of requirements engineering for software systems (Kotonya and Sommerville 1998) into functional and

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

non-functional requirements, the requirements presented so far are comparable to functional requirements, which specify, *what* a system is supposed to do. Contrasting those non-functional requirements represent constraints, *how* these functional requirements are to be realised. As such non-functional requirements the Guidelines of Modelling (Schütte 1998) can be identified: Relevance, Correctness, Economic Efficiency, Systematic Design, Clarity and Comparability.

Relevance: The guideline of relevance considers the problem adequacy and tractability of model construction that are highly dependent on the constructing engineer's perspective.

Correctness: The guideline of correctness addresses the syntactic and semantic correctness of a model.

Economic Efficiency: The guideline of economic efficiency points out the necessity of economic advantage for modelling projects.

Systematic Design: In order to reduce complexity the guideline of systematic design provides a description of different views of the domain and availability of a view spanning meta-model. A common practice differentiates between static and dynamic views.

Clarity: The Guideline of clarity bears on clearness of models for different users.

Comparability: The possibility of comparing different models has to be guaranteed, which is of particular importance in target/actual comparisons.

These guidelines of model creation, which have to be followed during the modelling process also build the frame of the modelling method. As further non-functional requirements and further general conditions a focus on the logistic expert as the modeller (and user of the method) and a focus on the domain of production logistics can be identified.

It becomes obvious, that these requirements altogether result in partly conflicting requirements to a model or a modelling method. These conflicts have to be identified and balanced.

Overview of the modelling method

The modelling method consists of the components illustrated in figure 2. The "Principles", shown in the center of figure 2, define the basic structuring of the method. They consist of a view concept, each emphasizing certain aspects of the system to be modeled, as well as elementary guidelines of modelling. The "Meta Model" specifies the modelling elements usable by the modeler in a view-spanning manner. "Diagrams" defines the graphical notation representing these elements and the contexts where they

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

can occur. It defines different diagrams each focusing on different facets of the system and visualizing them. Some examples of these diagrams are discussed later on in conjunction with discussing the view concept.

On the basis of the defined elements a reference model for autonomous cooperating processes is established. This reference model is available to the modeler as a set of building blocks easing model construction. The business process specialist will also get a modelling tool and the procedure model sketched in steps 1 and 2 of the system development process described in the previous section that is intended to guide the user through analysis and specification of autonomous cooperating logistic processes in the surveyed system.

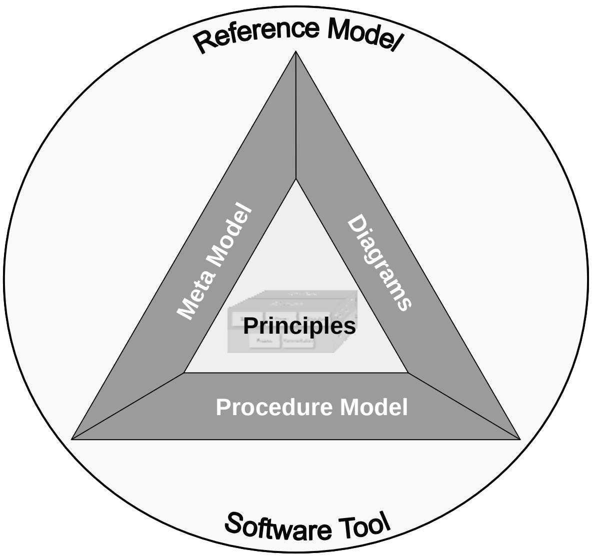

Fig. 2. Elements of the proposed modelling method

Modelling Concept

Before the next part gives further details of the modelling method and the view concept behind it, we will give an overview of the modelling concept on an abstract level, i.e. shown in the context of different modelling levels (see figure 3). The figure shows different modelling levels, from the mapping of the real system at the bottom to the model level as well as from the modelling layers to their respective meta-levels. The distinction between model and meta-model is the same as between the real system level and the model level: the higher level contains explicitly the elements that can be used to model the level below. This means the meta-model-level specifies the elements that can be used to model the system on the model level. Speaking of "elements" this refers only to one aspect of the level transition, the specification of the modelling language. This aspect is called "language-based metaisation" in contrast to "process-based metaisation" which shows the modelling procedure to be used on the level below.

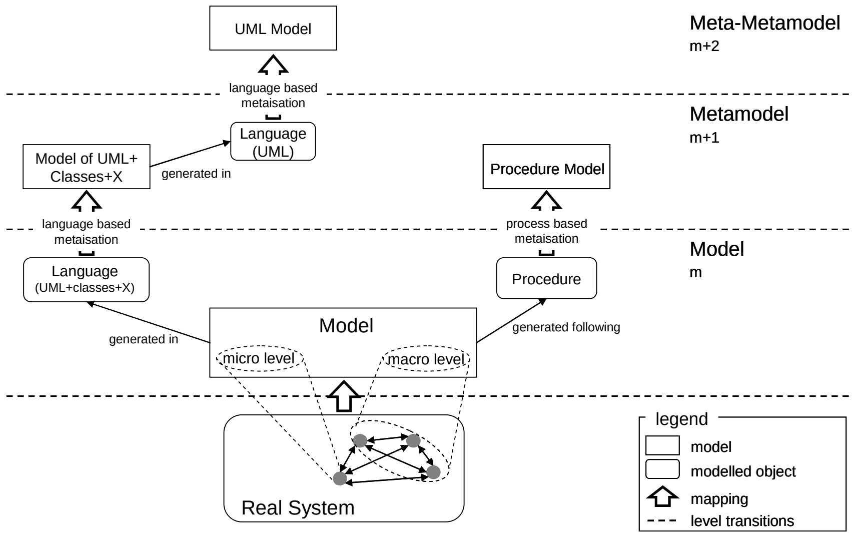

Fig. 3. Modelling method in the context of different modelling levels

On the lowest layer of figure 3 the (real or thought) system can be found. This is the system to be modeled; the modelling process itself is indicated by the lowest layer transition. Additionally the distinction between a macro- and a micro-level in modelling is indicated. Details regarding this point can be found in the next section. The model on one hand was created in a certain modelling language and on the other hand created following a certain modelling process. Therefore the layer transition from the model to the meta-model-layer distinguishes between language-based and process-based metaisation (for more information on metaisation refer to (Strahringer 1999)). Explicit representation of the creation process leads to the depiction of a procedure model for modelling. The procedure model will be represented using natural language and the process of its creation is not of particular interest to us thus nothing is shown in the figure on the meta-meta-model layer regarding the language- or process-based metaisation of the procedure model.

Concerning the branch of language-based metaisation and the transition from model- to meta-model-layer, the modelling language respectively

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

modelling notation is explicicated. Our modelling notation is based on version 2.0 of the Unified Modelling Language (UML). In addition to that the modeller will be supported by pre-defined domain-specific classes and logistic-specific process-parts and process-templates. The UML notation is extended to better show certain aspects of the logistic system, for example by elements taken from software agent modelling. These extensions of the modelling language are indicated in the figure by the “X”.

This (language-based) meta-model again is depicted in a certain way. At this point the distinction between language- and process-based metaisation could be made again, but only the first is of interest here. To represent the modelling notation, as a means of semi-formal modelling, UML will be used. To depict the fact that also this modelling language has to be specified somewhere, the top-layer shows the “model of UML”, being the UML specification (see (OMG 2006)). Relative to the modelling we aim at, this specification is on the layer of a meta-meta-model, strictly following language-based metaisation.

Concept of views and notational elements

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 (Scheer 1994) which is also reflected in the guideline of systematic design (see subsection Requirements to modelling). Based on the requirements mentioned above a view concept for modelling of autonomous logistic processes is proposed, whose views are depicted in figure 4. 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 (OMG 2006) that is also appropriate here.

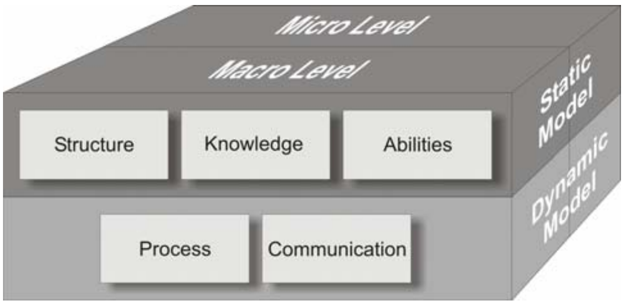

Fig. 4. View concept

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

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 (Sowa 2000). 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. Processes of a logistic system need certain abilities, which have to be provided by the logistic objects. These abilities are supposed to be seen as abstractions of problem types occurring in reality.

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 (Oestereich 2003).

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 (Weiß and Jakob 2005). 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.

In addition to the dynamic and static model just described we distinguish a macro and micro perspective. This distinction is also used in methods for software agent development (Weiß 2000). The macro view describes the interaction between the autonomous logistic objects. To some extend, it shows an external view onto the system, its elements and their relations and interactions. On the contrary the micro view describes the actions within and composition of the autonomous logistic objects. For the micro-level especially the process, knowledge and ability view are relevant, while all views proposed are relevant for the macro-level. This means

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

that the micro-macro perspective is orthogonal to the views shown in figure 4. Nevertheless not all views use both perspectives to the same extend.

As an example for the static model and to clarify the described modelling concept figure 5 shows a part of the classes available to the modeller. He can create instances of the existing classes as well as adapt and/or expand the class model. This means that the diagram is a basis that can be adapted for applications of the modelling method if necessary and furthermore be used to model a concrete scenario by creating instances of these classes, e.g. to model actual machines or work plans. The figure shows some relevant classes and the most important relations between them. For clarity reasons there are no multiplicities included in the diagram and most role names as well as attributes of the classes are omitted. To create the collection of domain specific classes (Scheer 1994), (Loos 1992) and (Schönsleben 2001) were used as references. The models presented there were used in context of information system development and are now adapted to our requirements of modelling autonomous logistic processes.

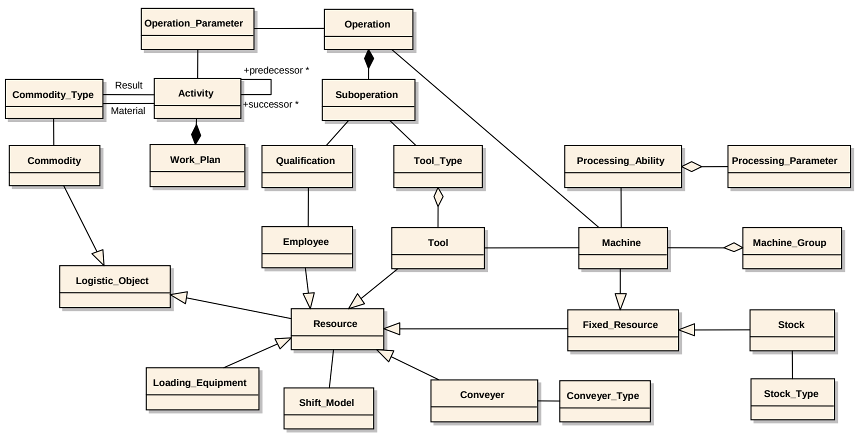

Fig. 5. Class diagram showing a part of the taxonomy supporting the user and selected relationships between the classes shown

As central classes “Logistic Object” and “Resource” (itself being a logistic object as indicated by the inheritance relationship) can be identified. Logistic objects are in principle able to be the autonomous objects of autonomous logistic processes. Kinds of logistic objects are commodity, all types of resources and orders (not shown in the selected classes above). Commodity represents a concrete logistic object in a material flow, e.g. an individual end-product, while commodity type is used when a commodity shall be referred to anonymously. A commodity type might be a type of

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

end product, intermediate product or raw material. Work plans, which are an aggregation of "Activities" specify how a commodity can be manufactured, i.e. which work steps to perform and what the required material(s) are and what the result of such a processing or assembly step is. This work plan is specified anonymously, i.e. for "Commodity type"s. "Resource" represents a common base class for physical and rather permanent components of a production system, each of them can be associated with a "Shift Model", which determines resource availability and therefore is an important factor for its capacity. Specialisations of the resource class are machine, tool or stock as well as conveyer, tool, loading equipment and employees, the latter being a software representation or an interface of/to workers on the shop floor.

In order to facilitate a loose coupling of the components of our logistics system there is no static mapping between the activities within a work plan and the machines or other resources to perform them. This is advantageous to achieve a more adaptive behaviour of the system. If new machines are added to the shop floor, they can start processing in a "plug-and-play"-like manner without the necessity to change all existing work plans. Work plans only specify which activity to perform and their parameters, as a simplified example drilling, 5mm wide, 7mm deep. To determine the next machine a commodity asks machines which of them can perform a certain activity. This negotiation process is further specified in the communication and process views. A machine is able to autonomously deduce whether it is able to perform an activity on the basis of its processing abilities stored within it (e.g. able to perform "drilling" in the range of 2-10mm wide, 1-20mm deep). Furthermore it is able to create operations on the basis of activities and processing abilities, which in detail specify which and how long tools and personnel are required to perform such an activity.

As an example for the dynamic model figure 6 shows an exemplary sequence diagram as part of the communication view. The example is rather simplified and concentrates just on commodity-machine communication although availability of conveyers must be considered in a resource selection process. The diagram shows a machine object and a commodity object. The exchanged messages are shown chronologically in vertical direction. The commodity requests a machining process answered by the machine with a quote. After the machine has selected a quote (the selection itself with its criteria and algorithms is modelled in the micro level process view) the chosen machine is booked by the commodity, the others are informed about the quote cancellation. In figure 6 this is modelled by a combined fragment of the type "alternative".

The presented example also shows some deficits of the UML 2.0 standard with respect to modelling autonomous logistic processes. It is not one

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

commodity communicating with one machine, but one commodity communicating with multiple machines. On the other hand the "maschine selected"-part of the alternative fragment is only executed with one machine. For increased clearness this should be modelled explicitly. One possibility to assure clearness could be an extended notation similar to cardinality which is proposed for software agent modelling with UML using specific extensions (Bauer and Odell 2005).

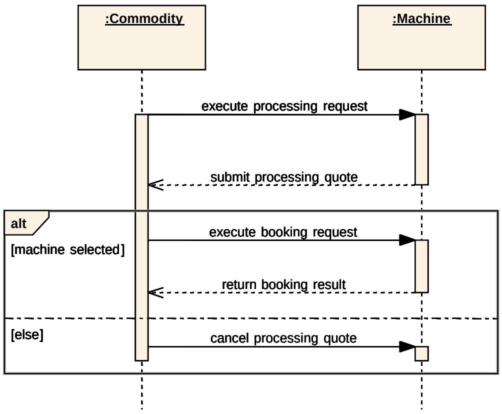

Fig. 6. UML sequence diagram machine selection

Fulfilment of requirements

After presenting requirements to the modelling method earlier in this paper, this section will investigate in how far the requirements are fulfilled by the designed modelling method as presented in the previous part of this paper.

First of all the fulfilment of the two general, i.e. non-functional, requirements of a focus on the domain of production logistics and the logistic expert as a modeller will be investigated. The latter requirement can for instance be found in the use of UML as the basis of the modelling notation used. As a graphical, semiformal notation it is broadly used – besides software development (especially agent-oriented approaches are of particular interest here, see for instance AUML (Odell et al. 2001; Bauer et al. 2001)) it is also used for knowledge modelling (Schreiber et al. 2002) or business process modelling (Oestereich 2003). Its broad use makes it likely that the logistic expert assigned to the system design already came in touch with this notation earlier in one context or the other. As it is furthermore an intuitive graphical notation, with its expressiveness reduced to only the

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

sub-set necessary here, the learning effort is accordingly low. The extensions by logistics-specific notational elements and the production logistic reference process also make the method easier accessible for the logistics expert. Both of these points, the extension of the notation with logistics specifics (e.g. a layout diagram) and the reference process consisting of an ontology of production logistic concepts and an exemplary definition of intelligent objects' processes express the requirement "focus on the domain of production logistics". Additionally the use of UML also fulfils the requirement of considering the later phase of software implementation. A language continuously used from the process model to the detailed analysis of the of the software system to be implemented, avoids a break in the development process, as the different fields involved all use the same semantic constructs (Oestereich 2003).

Regarding the primary requirements, supporting the design of intelligent logistic objects implies an approach focussed in these objects and opposes a strict top-down-design approach. This will be accounted for in the procedure model by its use of a mixture of a bottom-up and top-down-approach.

The interacting system elements (especially the intelligent logistic objects, but also other system components) can be shown in the Structure View. Here also intelligent logistic objects can be marked as such and their life-cycle described by an associated state-chart in the Process View. A description of the information-processing processes respectively decision processes also takes place in the Process View using Activity Diagrams. Not only an assignment of processes to the logistic objects they are located on (location of decision) is conducted here, but also the knowledge required for a decision can be modelled explicitly using object nodes. The structure of this knowledge and its initial distribution can in turn be shown in the Knowledge View, using Class Diagrams for the structure of the knowledge objects and Knowledge Maps to show its distribution. The interaction of the system elements among each other and their environment respectively is primarily described in the Communication View. UML Sequence Diagrams can be used here to specify interaction protocols. Event mechanisms (in Activity Diagrams and State Charts) can also be used to depict interaction with the environment and other system elements. They can also be used to initiate decision execution and monitor their execution progress. To be able to not only model direct communication between the intelligent logistic objects but also to allow to specify communication with the environment as a means of interaction is important to model stigmergy-based coordination (for a discussion of a stigmergy-based approach in the context of autonomous logistic processes see (Scholz-Reiter et al. 2006)).

The heterarchical decision structure is not a characteristic of the meta-model respectively the notation, but a property of the processes in their en-

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

tirety. The reference model created has this property; there is no central entity that renders a decision which is then delegated to executing instances.

Summary

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 requirements to a suitable modelling method were derived. The concept of our modelling method was presented subsequently, first giving a rough overview, then detailing selected aspects of it such as the view concept. The last section investigated in how far the designed modelling method fulfils the requirements derived in the beginning of the paper.

Further research will be concerned with the elaboration of the procedure model. The meta-model and graphical notation will be specified formally in a manner suitable to be used in later software implementation. This is important as our work is aimed at 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.

References

Bauer B, Müller JP, Odell J (2001) Agent UML: A formalism for specifying multiagent software systems. In: International Journal of Software Engineering and Knowledge Engineering 11 (3):207–230

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

Finkenzeller K (2003) RFID-Handbook – Fundamentals and Applications in Contactless Smart Cards Identification, 2nd edn. Wiley & Sons LTD, Swadlincote, UK

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

Haberfellner R, Daenzer W F, Huber F (2002) (eds) Systems Engineering. Methodik und Praxis, 11. edn. Verlag Industrielle Organisation, Zürich

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

Kotonya G, Sommerville I (1998) Requirements Engineering. John Wiley and Sons Ltd, Chichester et al.

Loos P (1992) Datenstrukturierung in der Fertigung. Oldenbourg, München

Oestereich B (2003) Objektorientierte Geschäftsprozessmodellierung mit der UML. dpunkt Verlag, Heidelberg

Odell J, Parunak HVD, Bauer B (2001) Representing agent interaction protocols in UML. In: Ciancarini P, Wooldridge M (eds) AOSE 2000, LNCS 1957, Springer-Verlag, Berlin Heidelberg, pp 121−140

OMG Unified Modelling Language Specification (version 2.0), Retrieved January 27 2006, http://www.uml.org

Scheer AW (1994) Business Process Engineering – Reference Models for Industrial Companies, 2. edn. Springer, Berlin et al.

Scholz-Reiter B, Freitag M, de Beer C, Jagalski T (2005) Modelling dynamics of autonomous logistic processes: Discrete-event versus continuous approaches. In: CIRP Annals 1 (2005), No. 55, pp 413–417

Scholz-Reiter B, Freitag M, de Beer C, Jagalski T (2006) Modeling and simulation of pheromone based shop floor control. In: Cunha PF, Maropoulos P (eds): Proceedings of the 3rd CIRP Conference Digital Enterprise Technology, Setubal, Portugal 18-20 Sept 2006

Scholz-Reiter B, Philipp T, de Beer C, Windt K, Freitag M (2006) Einfluss der strukturellen Komplexität auf den Einsatz von selbststeuernden logistischen Prozessen. In: Pfohl H-C, Wimmer T (eds): Steuerung von Logistiksystemen - auf dem Weg zur Selbststeuerung. Konferenzband zum 3. BVL-Wissenschaftssymposium Logistik. Hamburg: Deutscher Verkehrs-Verlag, pp 11–25

Scholz-Reiter B, Windt K, Freitag, M (2004) Autonomous Logistic Processes - New Demands and First Approaches. In: Monostori L (ed.) Proc. 37th CIRP

Scholz-Reiter, B.; Kolditz, J.; Hildebrandt, T.: Specifying adaptive business processes within the production logistics domain – a new modelling concept and its challenges. In: Hülsmann, M.; Windt, K. (eds.): Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow. Springer, Berlin, 2007

International Seminar on Manufacturing Systems. Hungarian Academy of Science, Budapest, Hungaria, pp 357–362

Schönsleben P (2001) Integrales Informationsmanagement: Informationssysteme für Geschäftsprozesse – Management, Modellierung, Lebenszyklus und Technologie, 2. edn. Springer, Berlin et al.

Schreiber G, Akkermans H, Anjwerden, A, de Hoog R, Shadbolt, N, van de Velde W, Wielinga B (2002) Knowledge Engineering and Management - The CommonKADS Methodology. MIT Press, Cambridge

Schütte R (1998) Grundsätze ordnungsmäßiger Referenzmodellierung: Konstruktion konfigurations- und anpassungsorientierter Modelle. Gabler Verlag, Wiesbaden

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

Strahringer S (1999) Probleme und Gefahren im Umgang mit Meta-Begriffen: ein Plädoyer für eine sorgfältige Begriffsbildung. In: Proceedings of the International Knowledge Technology Forum (KnowTechForum) '99. Potsdam, Germany

v. Uthmann C, Becker J (1999) Guidelines of Modeling (GoM) for Business Process Simulation. In: Scholz-Reiter B, Stahlmann HD, Nethe E (eds) Process Modelling. Springer, Berlin, pp 100-116

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

Weiß G, Jakob R (2005) Agentenorientierte Softwareentwicklung. Springer, Berlin, Heidelberg

Windt K, Böse F, Philipp T (2007) Autonomy in Logistics - Identification, Characterisation and Application. In: International Journal of Robotics and CIM, forthcoming 2007