Modeling the Control System Infrastructure for Autonomous Logistics Processes

B. Scholz-Reiter[1], St. Sowade[1], D. Rippel[1]

Dept. of Intelligent Production and Logistic Systems, BIBA – Bremer Institut für Produktion und Logistik GmbH at the University of Bremen, Bremen, Germany

Abstract

Autonomous control is able to increase the flexibility and robustness of logistic systems by enabling decentralized decision making and execution at the system elements. This paper presents a qualitative model of terms and drivers being relevant to configure the infrastructure of autonomous logistic control systems. First, it presents the terms autonomous control, infrastructure, configuration, and logistic system in the application area of production logistics. Second, the paper discusses a control system's macro and micro architectures. The former are characterized by the kind of control representation, the extent of ability transfer to logistic objects, and the localization of the abilities. The latter are compared to embedded systems and their components.

Keywords:

Autonomous Control System Design, Control System Infrastructure, System Architecture

1 INTRODUCTION

Manufacturing involves several production and assembly processes in order to fulfill customers' demands for industrial and consumer goods. The efficiency of each logistic process is crucial for the profitability of a company as well as for the overall value creation in a supply chain. Hence, companies employ powerful planning and control (PPC) systems which manage the achievement of their logistic objectives, e.g. lead time, produced amount, capacity utilization, and adherence to scheduling [1]. Centralized PPC systems are used widely today and perform well in general case. However, they lack flexibility and robustness in case of uncertainty, like fluctuating supply or demand of machine capacity due to machine breakdowns or rush orders. For this reason, researchers investigate alternative PPC concepts. The paradigm of autonomous control in logistics is a decentralized PPC approach being able to cope with highly fluctuating supply and demand of manufacturing capacity [2]. Flexibility and robustness shall be increased by the use of intelligent logistic objects, which utilize decision methods in order to render and execute decisions locally by themselves [3]. For this purpose, logistic objects, like orders, resources, and commodities, are equipped with abilities for gathering and processing of information, i.e., a precondition for decision making. Realization of the abilities requires additional hard- and software components which form the infrastructure of an autonomously working control system [4]. However, design and selection of infrastructure components is a complex task due to a high number of possible infrastructure configurations. Logistic process experts have to consider several scenario-specific and model-theoretic aspects. Thereto, this paper presents a qualitative model of terms and drivers which structures the configuration task for logistics process experts. In addition, selected basic terms and drivers are described in detail.

After the subject area has been introduced in section one, the remainder of the paper is structured as follows. Section two draws a model of terms and drivers for the configuration of the infrastructure of autonomously working control systems. Its subsections recapitulate basic model-theoretic terms and describe the micro and the macro perspective of control system architectures. The paper closes with a summary and an outlook to future work.

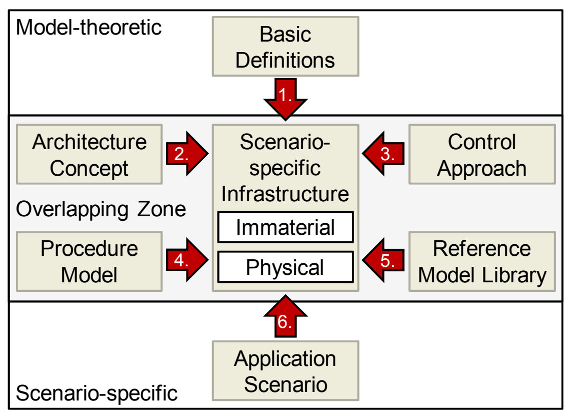

Figure 1: Terms and Drivers of the Infrastructure Configuration of Autonomously Working Control Systems. [4]

2 MODEL OF TERMS AND DRIVERS

Several aspects influence the configuration of an infrastructure for autonomous logistic processes. This section classifies them into six terms and drivers and presents them in a single qualitative block diagram model pictured in Figure 1. The numbered arrows indicate the order of each element's investigation. The terms and drivers influence either the appearance of immaterial or physical scenario-specific infrastructure or determine the process that helps to derive its components. In general, all six elements are either model-theoretic or they are scenario-specific. Model-theoretic elements contribute generic ideas to infrastructure models. Contrarily, scenario-specific elements introduce a specific scenarios' view and semantic into the infrastructure configuration process. However, both views form an overlapping zone whose elements own model-theoretic and scenario-specific characteristics. Indeed, basic definitions are strict model-theoretic, while application scenarios are strict scenario-specific. Elements two to five are overlapping. All terms and drivers are considered as tools in order to determine an autonomously logistic control system's infrastructure. The overall infrastructure of such control systems provides logistic objects with all capabilities necessary in order to follow up their objectives.

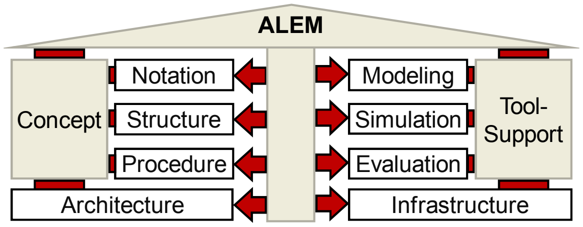

Figure 2: ALEM Framework [5].

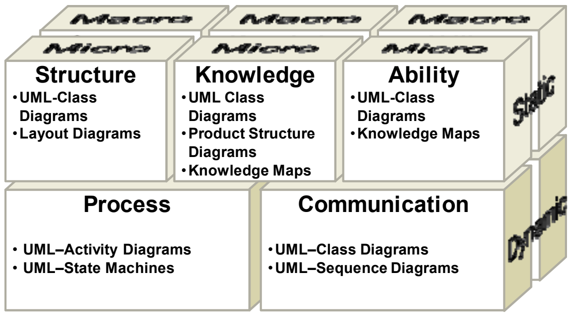

Figure 3: ALEM View Concept [5].

This section recapitulates basic definitions, and then introduces the second and the third model element in detail. Remaining elements are described briefly in order to gain a system-theoretic understanding. Subsection 2.1 describes characteristics of autonomous control in logistics. Subsection 2.2 reviews the terms infrastructure and configuration. Subsection 2.3 shows infrastructure elements of production logistic systems. Subsection 2.4 introduces two aspects of control system architectures: a macro level which refers to the overall autonomous system and a micro level referring to the structure of single logistic objects.

2.1 Autonomous Control

Autonomous control is one possibility to handle the increasing complexity and dynamics of logistic systems. Their elements are named intelligent logistic objects and are able "to process information, to render and to execute decisions on their own." [3]. Accordingly, Hülsmann and Windt define autonomous control in the application area of logistics as "processes of decentralized decision-making in heterarchical structures. It presumes interacting elements in non-deterministic systems, which possess the capability and possibility to render decisions independently. The objective of Autonomous Control is the achievement of increased robustness and positive emergence of the total system due to distributed and flexible coping with dynamics and complexity." [3]. Thus, presence of decision alternatives is the first precondition in order to allow local decision making by logistic objects [6]. Second, intelligent logistic objects require decision competence in form of knowledge about methods and algorithms, as well as environment and object specific information.

Böse and Windt developed a catalogue of criteria in order to characterize autonomous systems by their level of autonomous control [7]. The catalogue assigns several criteria to a system's decision layer, information layer, and execution layer. Each criterion expresses a component of a system's degree of autonomous control. The relative importance of each criterion to each other is weighted by a pair-wise comparison. The properties of the criteria span the minimum and the maximum level of autonomous control of a system [3].

In order to enable modeling of autonomous logistic business processes researchers of the Collaborative Research Centre (CRC) 637 are developing the Autonomous Logistics Engineering Methodology (ALEM) displayed in Figure 2 [8], [9]. ALEM comprises a notation, a structuring view concept, and a procedure model. In addition, the modeling process relies on the decision for a specific system architecture and on the selection of appropriate infrastructure components that are required for autonomous controlled logistic processes. The ALEM software tool (ALEM-T) integrates all methodical components and guides logistic process experts through the process of model creation, simulation, and evaluation. ALEM's notation bases on the Unified Modeling Language (UML). However, ALEM extends the UML by diagrams that are required in the domain of autonomous logistic processes, i.e., knowledge maps, a layout diagram, and product structure diagrams [10], [11].

ALEM uses a view concept in order to handle a models complexity [6], [12]. Each view focuses on single aspects of an overall model and enables designing of lower complex model segments [13]. Five primary views divide an overall model into distinct, semantic aspects (Figure 3). Additional views group static and dynamic model segments. Static aspects describe time invariant model components. Dynamic aspects subsume time depending procedures performed by logistic objects. Further, micro views picture an object's internal model, while macro views describe aspects of the overall system.

The semantic views differentiate ALEM models by their structure, knowledge, abilities, processes, and communication. Each view depicts a definite aspect in one or more diagrams types [5]. The structure view primarily contains static and macro model elements. It defines all logistic objects present in a system as well as their relationships. The structure view includes also a spatial layout of the logistic scenario. The knowledge view focuses on static and micro aspects and describes all knowledge an object owns with the help of UML-Class diagrams. Moreover, the knowledge view employs product structure diagrams and knowledge maps. The ability view is a static view and includes micro as well as macro aspects. It represents the abilities of logistic objects in UML-Class diagrams. In addition, knowledge maps assign the abilities to specific logistic objects. The process view is a part of the dynamic view and incorporates micro and macro aspects. It denotes a logistic objects' behavior in UML-State Machine and UML-Activity diagrams. The communication view focuses on dynamic and macro aspects. UML-Class diagrams define messages exchanged between logistic objects, while UML-Sequence diagrams represent communication protocols.

[9] developed the procedure model ALEM-P (procedure) in order to guide logistic process experts through the modeling process of autonomous logistic business processes (Figure 4). Each of its eight steps deals with one model aspect. Although the procedure model's steps show a straight sequence, the modeling order can vary. Reordering of the procedure steps depends on whether a top-down or a bottom up approach is used for modeling. For instance, if a specific method or algorithm for autonomous decision making shall be employed, the decision process has to be described before modeling an object's abilities. In addition, feedback loops allow the integration of new system aspects in previous steps. Several aspects are worth to be mentioned in detail. However, for a full description of each step see. Objectives are modeled in the first step and are a precise kind of knowledge logistic objects owns. Step four and five refer to different views of the process model. Decisions focus on the decision rendering itself (micro view), while processes are sequences of tasks performed by logistic objects (macro view). Finally, modelers instantiate all ALEM submodels

in order to set up a logistic scenario which shows all logistic objects' states expressed by parameter values, like the spatial configuration of the system. However, ALEM-P lacks to provide a procedure for the configuration of the infrastructure of autonomous logistic processes. Hence, such a model is currently being researched.

2.2 Infrastructure and Configuration

The term infrastructure has several meanings across very different application areas, e.g. economic, military, politics, and informatics. In origin "infrastructure" denotes durable, stationary facilities which are connected to the ground [6]. Politics uses infrastructure often for service-network-oriented basic works, e.g. power grids. Informatics understands hardware and software equipment as information technology (IT) infrastructure. [14] defines infrastructure as: "elementary human-made facilities, which are a precondition for a high developed economy and may change over time. Its (…) characteristics are its base character, artificiality, indispensability for proper functionality, and changeability." Thus, infrastructure is an economical and organizational foundation.

However, these approaches neither include logistic specifics, nor do they lead to a general understanding in terms of a logistic control system. For this reason, [4] define infrastructure in a system-theoretic view:

"Infrastructure includes all system elements which are placed artificially into a given system, called native system. These system elements must be essential to enable specific higher order services within the system by use of capabilities supplied by native system elements and by artificially inserted system elements." [4].

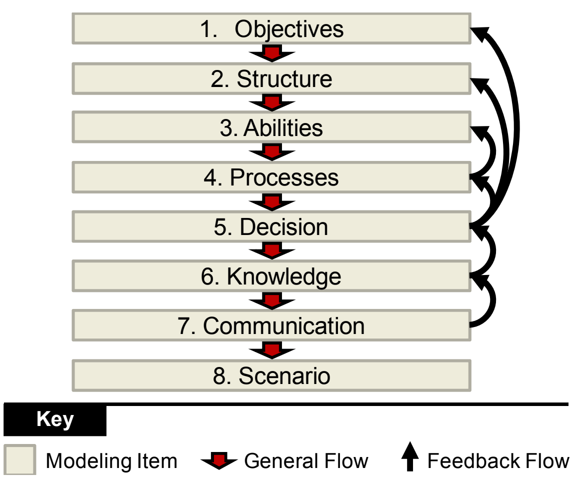

Figure 4: ALEM Procedure Model [9].

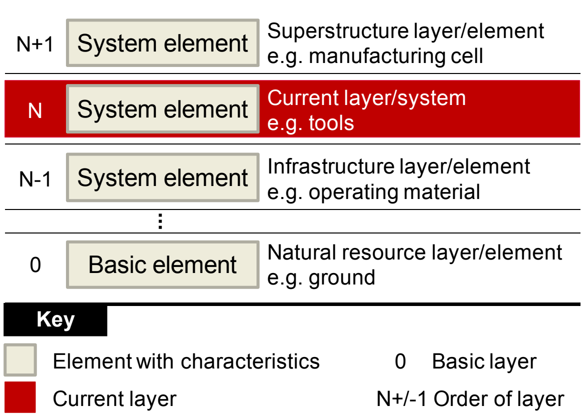

Figure 5: Infrastructure Layer Model. [4]

In conclusion, neither capabilities of a native system nor new system elements are able to perform demanded activities themselves. Instead, artificially inserted elements are a precondition in order to execute higher order tasks in spatial delimited logistic systems. Further, the elements addressed in this definition denote a hierarchy which can be represented in a generic infrastructure layer model (Figure. 5). System elements are assigned to distinct layers which provide specific functional services to higher layer elements. Thus, all elements located in layer N-1 are infrastructure from layer N view. Elements placed in layer N+1 are superstructure components. Elements placed at the bottom belong to the native system.

The characteristics of infrastructure as proposed by [6] remain. However, the artificiality refers to the process to add another element, but not to the type of element. Further, establishment of infrastructure requires resources and usually leads to sunk costs. The socio-technological development determines an infrastructure's social impact. Several authors distinguish infrastructure by its usage, dedication, materiality, network orientation, and level type [6], [7]. Often, institutional/personal is used in order to describe regulation frameworks and capabilities of people. However, this understanding is incomplete and imprecise, because it neglects technical norms and standards. Thus, [4] uses the term immaterial instead. Further, an infrastructure is network-based, if its elements are nodes being interconnected via links and the resulting network enables the system's functionality. Secondary infrastructure requires elements in more than one infrastructure layer. System elements located in layer N-1 are primary infrastructure, while elements in layer N-2 are secondary, subordinated infrastructure that is required for the primary infrastructure's functionality.

A configuration is an arrangement of objects, respective system elements, being used together for a specific purpose [8]. The type of possible configurations depends on a systems' elements logical design and characteristics. A basic configuration is a recurring or frequent arrangement of objects. Predefined basic configurations reduce the efforts on configuring a system. They are used as a starting point for further modifications in order to ensure inclusion of all necessary or desired objects [9]. Reference models are examples of basic configurations.

2.3 Production Logistic and Logistic Infrastructure

In order to help logistic process experts to identify infrastructure components in manufacturing systems, Figure 6 presents a model of two selected layers of manufacturing systems. Functions located in the execution system layer require functional units which provide demanded tasks. Functional units decompose into components of the main structure, e.g. machines and trolleys, and of the infrastructure, e.g. tracks, energy grid, and mounting clamps. Accordingly, the control system layer requires functional units that process and manage data for manufacturing control. These units contain control algorithms and technical components for information processing and propagation.

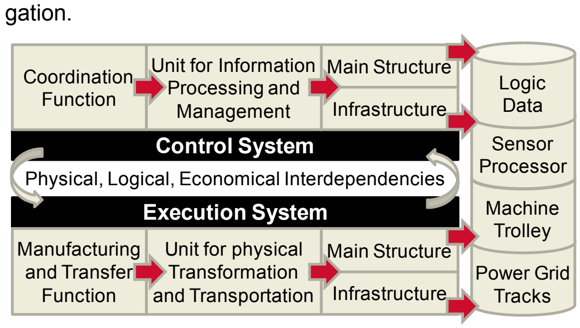

Figure 6: Infrastructure in Manufacturing Systems. [4]

Thus, control systems' functional units sub-divide into main structure, e.g. processing logic and data, and infrastructure, e.g. sensors, processors, and transceivers. Finally, the infrastructure of a production control system includes all necessary components to enable coordination functions [4].

However, it is difficult to position the components in an autonomously controlled system as well as in the corresponding control system, as they miss integration into the system layers of autonomous control. Thus, Figure 7 presents an approach in order to arrange required infrastructure components to the three system layers of autonomously controlled processes. The system layers demand various coordination functions and thus require different sets of infrastructure components. These are derived from the coordination function of the control system and address primarily one system layer. However, they also loom into neighbor layers. Their classification reduces the complexity of the discussion of specific infrastructure components.

Algorithms and rules belong to the management layer. Infrastructure components selected for this layer affect the information-processing layer as well. The components of the information-processing layer focus on devices for gathering, processing, and distribution of information. They provide major functionality for local decision-making and execution. This layer's components affect both surrounding layers. Especially, the execution layer is directly affected by decisions made in the information-processing layer. Further, components, like sensors and actuators, form the infrastructure of the execution layer. They enable logistic objects to interact with their environment. Although this layer's components operate primarily in the execution layer, they take effect in the information-processing layer as well [4].

2.4 Control System Architecture

Previous subsections introduced basic terms that affect the configuration of a control system's infrastructure. This subsection discusses architecture concepts as second item in the model of terms and drivers presented in Figure 1. An architecture's design is crucial for important system characteristics, like performance, reliability, resources consumption [15].

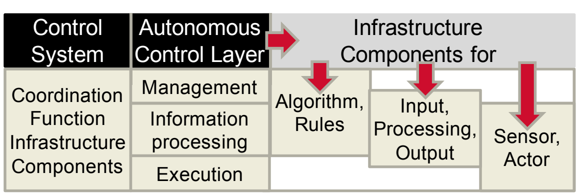

Figure 7: Control System Infrastructure Components [4].

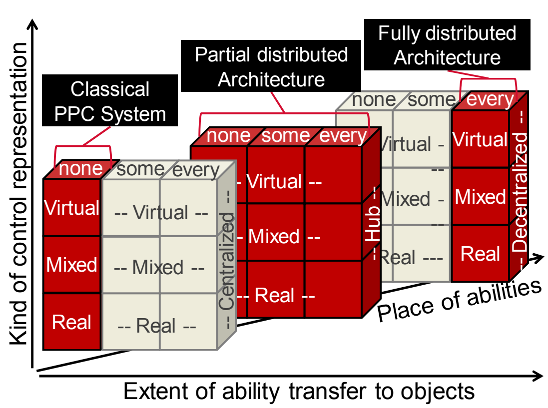

Figure 8: Macro View on Architecture Types.

Generally, architectures describe the arrangement and linking of system components. Architectures of a control system are presented in the meaning of options for design, arrangement, and dimensioning of underlying system components. Control system architectures can be categorized into macro architectures which relate to the overall system and into micro architectures describing the structure inside of system elements. In both cases, an architecture has to fulfill specific functional and non-functional requirements. Utilization of already present system components leads to reuse their capabilities for new infrastructure purposes and may reduce the control system's initial costs. Thus, selection and arrangement of system components in order to form a control system is an important issue. This task directly influences the appearance of obligatory infrastructure components, e.g. for purposes of communication and energy supply. Moreover, the architecture type determines the locations where functional infrastructure components have to be present and affects the required interfaces between autonomous logistic objects, too.

Macro View on Architecture Types

Logistic process experts face several architecture design options in order to realize autonomous controlled logistic systems. The options can be differentiated by three dimensions: the kind of representation of a control system's elements, the extent of ability transfer from a central instance to logistic objects, and by the degree of distribution of the abilities – meaning the localization of abilities to logistic objects. The bigger both of the latter are, the more the concept of autonomous control is established in each system element, and the higher is the technologically complexity of the overall system. Figure 8 shows all three dimensions as axis and illustrates the location of the three macro system architecture classes: classical PPC system, partial distributed architecture, and fully distributed architecture.

In fully distributed control system architectures every logistic system element works as an intelligent logistic object which contains abilities in order to render and to execute decisions autonomously. Logistic systems use three classes of relevant objects: commodities (e.g. finished products, half-finished products, components, and raw materials), resources (e.g. production centers, machines, and transportation devices), and orders (e.g. customer orders or production orders). At this, commodities consume logistic services that are offered by resources. Logistic service classes are transportation, production, assembly, or storage. Orders represent the target state of commodities, while product structure diagrams denote their logistic transformation path. A fully distributed control system architecture allows a very flexible and adaptable material flow system due to distribution of decision making competence to all system elements. However, the resulting technical and logical complexity leads to high initial costs.

Contrary, partial distributed control system architectures extend only some logistic objects with decision-making abilities. Moreover, intelligent logistic objects offer their functionality as a service to less intelligent objects and thus constitute a client-server-relationship. The less intelligent clients are able to store their objectives and the demanded decision method. Further, they are able to transmit both to a server object which actually computes the decision and transmits it back to the client object. Finally, a client object executes this decision. For instance, servers are machines that own all decision making abilities, while clients are commodities that can only store and transmit small amounts of information. Partial distributed control system architectures are beneficial in scenarios, whose existing system elements offer sufficient computing and networking capacity that can be used for this purpose. This architecture type is a good approach if equipping each system element with all abilities is uneconomical, impractical, or inappropriate. In result, the systems' complexity concentrates at the server side. The initial costs are assumed to be lower than in the case of fully distributed control system architectures.

In both cases, logistic objects follow own objectives and act as agents for themselves. This applies also, if a server computes a decision for a client to its specifications in terms of objectives, decision methods, and processed information. In contrast, centralized PPC-systems compute schedules and plans with regards to a foreign objective system that excludes local objective systems, even if local logistic objects are present. Instead, centralized PPC-systems follow system-external objective systems and make local objects obey to it.

In any case, selected control system elements or a complete control system can be represented either in real or in virtual manner. In the first case, real logistic objects directly contain and use control functionalities. In the second case, a control systems' functionality is located apart from a logistic object, e.g. as agents representing objects in a single central computer system. Here, a control system's realization is virtual. Agents render decisions on behalf of logistic objects and transmit the results back to them for decentralized, real execution. The agents determine an objects behavior, but the logistic system is still autonomously controlled. If a complete logistic system is mapped into a central, real-time operating computer system and software agents represent every single physical object, all system information is represented in this computer system and every agent can use this knowledge. A logistic object's task is reduced to collect and forward information and to execute commands being provided by the virtual autonomous control system. The system's complexity concentrates in a central computer.

The decision for a control systems' architecture depends on the type and the number of already present system elements and their functionalities which can be used in order to provide specific control system abilities. Another role plays the targeted degree of autonomous control, the kind of control strategy realization, as well as social, operational, and economic factors [10].

Micro View on Architecture Types

Intelligent logistic objects are usual logistic objects being enhanced with specific properties and capabilities. In order to be more precise, they are autonomously working logistic objects which are able to perform information gathering, processing, and exchange, as well as decision making and execution by their own. They are denoted as intelligent. Implementation of the capabilities at each object requires the placement of appropriate hardware and software components. On a closer look, these additional components form computer systems that are located at logistic objects and thus, are embedded into the overall logistic system. Hence, autonomous logistic control systems are one application type of embedded systems as a comparison with embedded systems shows [16]. Holzmann characterizes embedded systems as "computer systems that are parts of larger systems and realize dedicated functions." [17]. In this sense, they integrate mechanisms for information exchange with other system elements. Especially, they "comprise sensing, actuating, computing and wireless communication capabilities" [17]. In result, they are context-aware and are equipped with all capabilities required to work autonomously. Wireless sensor networks or mobile phones are typical examples of embedded systems.

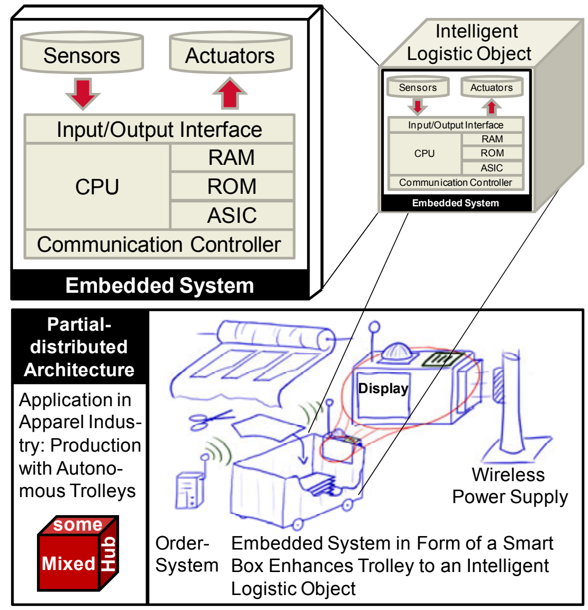

Figure 9: Micro View Architecture Elements and Intelligent Logistic Object in Exemplary Macro View Architecture [10], [19].

Further properties of embedded control systems are: spatial awareness, real-time capability, mobility, small size, limited energy and computing performance, spatial distributed elements, which is true for intelligent logistic objects as well [19], [17], [16]. Embedded systems are affected by the problem of hardware and software co-design [16]. Common computer hardware architectures can be employed in embedded systems, e.g. a von Neumann or a Harvard architecture [16]. They are distinguished by the location used in order to store the program instructions and the operand data. A von Neumann architecture is used, if the information is kept in the same memory. If both are stored in separate memory devices, a Harvard architecture is employed. The operating software of the embedded system and the hardware components are highly interwoven with each other.

Figure 9 shows relevant components of a single embedded system that is part of an intelligent logistic object. Each embedded system has a central processing unit (CPU) processing information received by sensors via an input/output interface or via a communication controller that is able to retrieve information from compatible logistic elements. A read only memory (ROM) stores static knowledge, like a firmware, a random access memory (RAM) keeps changeable information, and application specific integrated circuits (ASICS) can accelerate specific tasks. In this sense, the communication controller operates independently from the CPU [18], [16]. The type of required components and the micro view architecture depend on requirements that have to be derived from an application scenario. Hence, figure 9 also presents an extract of an apparel production scenario discussed in [10]. The scenario employs a partial-distributed macro view architecture, i.e. trolleys act as hubs for each garment piece and route them in lots through the shop floor. For the scenario pictured, all embedded system processing components have to be used. In addition, power supply of the embedded system is proposed to be wireless.

2.5 Outlook towards Remaining Terms and Drivers

Besides the terms discussed previously, a method is required in order to ensure that all important infrastructure design aspects are considered during the development process of autonomous logistic systems. A procedure model for the configuration of the control system's infrastructure is able to fulfill this purpose. Further, reference models may limit the applicability of infrastructure components by proposing obligatory elements in specific situations. For instance, selection of a specialized control method may decrease the number of adequate infrastructure components. A decision for a specific control method allows identification of control-method specific infrastructure components. Moreover, selection of a control method dimensions the capabilities of affected infrastructure components and determines the systems' behavior under dynamic influences. Finally, a selected application scenario influences the exact infrastructure configuration. It provides details of all planned logistic system elements, for example their capabilities, location, and parameters, as well as requirements and interdependencies with other objects of the logistic system, its infrastructure components and its environment.

3 SUMMARY

This paper presented a qualitative model of six terms and drivers that influence the selection of components for the control infrastructure of autonomously logistic systems. The model's elements were arranged in three groups indicating if they are strict model-theoretic, strict scenario-specific, or span both areas. The understanding of autonomous logistics was summed up and its definition [3] was quoted. Further, the generic infrastructure layer model of [4] was presented in order to sharpen the readers view on this central term. Additionally, the paper summarized the basic terms configuration and logistic system in the application area of production logistics. Finally, the paper introduced and discussed various architecture options for a corresponding control system. Macro architectures relate to the overall system and micro architectures describe the structure inside of a system's elements. Especially the introduction of micro architecture aspects led to several complete new aspects for the infrastructure design, because autonomous logistic control systems are a special application of embedded systems. Thus, the plentiful research results in the subject area of embedded systems can be used in order to reflect, to deepen, and to verify our own ideas in terms of design and configuration of autonomous logistic control systems. Hence, it is planned to address the latter issues in future research. Moreover, the remaining elements of the terms and driver model will be developed and described.

4 ACKNOWLEDGMENTS

This research is funded by the German Research Foundation (DFG) as part of the Subproject B2 of the Collaborative Research Center 637 "Autonomous Cooperation Logistic Processes – A Paradigm Shift and Its Limitations" (CRC 637).

5 REFERENCES

[1] Luczak, H., Eversheim, W., Schotten, M., 2001, Produktionsplanung und -steuerung: Grundlagen, Gestaltung, Konzepte, Springer Verlag, Berlin

[2] Scholz-Reiter, B., Rekersbrink, H., Görges, M., 2010, Dynamic Flexible Flow Shop Problems – Scheduling Heuristics vs. Autonomous Control, Anals of the CIRP, 59/1:465-468.

[3] Hülsmann, M., Windt, K., 2007, Understanding of Autonomous Cooperation and Control in Logistics – The Impact of Autonomy on Management, Information, Communication and Material Flow, Springer Verlag, Berlin

[4] Scholz-Reiter, B., Sowade, St., Rippel, D., 2010, Modeling the Infrastructure of Autonomous Logistic Control Systems, European Conference of Systems (ECS'10), WSEAS Press, Puerto de la Cruz, Tenerife, 295-300.

[5] Scholz-Reiter, B., Rippel, D., Sowade, St., 2010, Modeling and Simulation of Autonomous Logistic Processes, NAUN European Conference of Control (ECC'10), WSEAS Press, Puerto de la Cruz, Tenerife, 148-153.

[6] Scholz-Reiter, B., Höhns, H., Kolditz, J., Hildebrandt, T., 2005, Autonomous Supply Net Coordination, Proceedings of 38th CIRP Manufacturing Systems Seminar, Florianopolis

[7] Böse, F., Windt, K., 2007, Catalogue of Criteria for Autonomous Control, Springer, Berlin

[8] Scholz-Reiter, B., Kolditz, J., Hildebrandt, T., 2009, Engineering autonomously controlled logistic systems, International Journal of Production Research, 47/6:1449-1468.

[9] Kolditz, J., 2009, Vorgehensmodell zur Erstellung von Fachkonzepten für selbststeuernde produktionslogistische Prozesse, GITO Verlag, Berlin

[10] Scholz-Reiter, B., Sowade, St., Rippel, D., Teucke, M., Özsahin, M., Hildebrandt, T., 2009, A Contribution to the Application of Autonomous Control in Manufacturing, International Journal of Computers, 3/3:279–291.

[11] Scholz-Reiter, B., Kolditz, J., Hildebrandt, T., 2007, UML as a Basis to Model Autonomous Production Systems, Springer: Berlin, pp. 553–560.

[12] Scheer, A.-W., 1994, Business Process Engineering - Reference Models for Industrial Enterprises. Springer, Telos

[13] Scheer, A.-W., 2001, ARIS – Modellierungsmethoden, Metamodelle, Anwendungen, Springer Verlag, Berlin

[14] Klaus, S., 2009, Deregulierung der netzbasierten Infrastruktur, Ph.D. dissertation, Universität Zürich

[15] Berns, K., Schürmann, B., Trapp M., 2010, Eingebettete Systeme: Systemgrundlagen und Entwicklung eingebetteter Software, Vieweg+Teubner Verlag, Wiesbaden

[16] Hristu-Varsakelis, D., Levine, W. S., 2005, Handbook of Networked and Embedded Control Systems, Birkhäuser: Boston

[17] Holzmann, C., 2009, Spatial Awareness of Autonomous Embedded Systems, Vieweg+Teubner Verlag, Wiesbaden

[18] Pop, P., Eles, P., Peng, Z., 2004, Analysis and Synthesis of Distributed Real-Time Embedded Systems, Kluwer Academic Publishers, Dortrecht

[19] Ganssle, J., 2008, The Art of Designing Embedded Systems, Elsevier, Burlington