

# Towards Adaptable Manufacturing Systems

Nadine Keddis, Gerd Kainz, Christian Buckl  
fortiss GmbH  
Guerickestr. 25, 80805 Munich, Germany  
Email: {keddis, kainz, buckl}@fortiss.org

Alois Knoll  
Technical University Munich  
Boltzmannstr. 3, 85748 Garching, Germany  
knoll@in.tum.de

**Abstract**—Life cycles of many products are becoming shorter. In addition, the number of variants of one product is growing. As a fact, volume of one specific product that is being manufactured is decreasing. This leads to more frequent modifications of production lines. To cope with these changes, adaptable manufacturing systems are required. Current manufacturing systems can only be adapted to certain (predefined) situations. Other changes require high effort accompanied with high cost and setup time. In this paper, we focus on adaptivity with respect to IT systems. To increase the adaptability of IT systems for automation, we propose a model-based plug & play approach for integrating new stations. This helps in reducing changeover time and efforts. We propose different models describing stations and their capabilities, the setup of the factory, and the production plans. The system is then monitored automatically and the production is planned using models @ run-time. To abstract from different platforms and communication technologies data transfer is handled by a middleware. We evaluate our approach using an industrial production system used for educational purposes.

## I. INTRODUCTION

The demand for adaptable manufacturing systems has increased in the past years due to changes in the market. Turbulences in markets are no longer the exception. They are a result of fast changing technologies, environment (e.g., scarce resources), politics, society, or economy [1]. The demand also arises from the shift away from mass production towards mass customization. Manufacturing systems nowadays have to be able to produce several variants of the same product and even different products without much reconfiguration. At the same time, product quantities vary largely. Moreover, product life cycles have become shorter, leading to the need for a fast changeover process. Thus, the adaptability and changeability of production systems are increasingly becoming key features for manufacturing.

Today, production plans are fixed and optimized for efficient production of a single product. Therefore, they are rarely changed during production. In order to change the setup of a production system, the production has to be stopped and some of its parts have to be reconfigured, reprogrammed, or even replaced. The required changes involve high manual efforts [2] and reconfiguration of information technology (IT) systems. This is costly and time consuming – if at all possible. However, future manufacturing systems need to be reconfigured quickly to keep up with the fast pace of changes in markets.

Current approaches mainly focus on mechatronic compatibility to enable changes in factory setups with less configuration effort and time. Production lines are modularized from a

mechanical as well as an electrical point of view. It is possible to add new components that are mechatronically compatible with the available components. However, IT systems need to be reconfigured as well to integrate the new components. To achieve the vision of real adaptable manufacturing systems, IT systems have to be modularized as well [3].

As IT is becoming more and more important for manufacturing [4], new concepts from the IT domain are introduced to the automation industry. A standard software architecture featuring a modular construction and component layout as known from other domains would enable a fast and inexpensive reconfiguration of production lines. Ideally, this reconfiguration can then be done without any technical expertise. Future manufacturing systems should in addition offer services such as self-description, self-configuration, data acquisition, real-time monitoring, and planning of production [5]. For high adaptability on IT level, manufacturing execution systems (MES) have to be maintained and kept consistent with the highest degree of automation possible [3]. To be able to maintain and use the necessary information with minimal manual effort, the information can be stored in models. Therefore, we propose a model-based plug & play approach. The goal of this approach is to reduce manual efforts for setting-up and reconfiguring factories. Furthermore, it aims at automating required processes, thus enabling adaptability on IT level.

The suggested approach is based on the concept of models @ run-time. In this concept, models are not only used during design time, but also during run-time. The system can itself change the models to reflect changes of the system also at modeling level. Tools can then calculate necessary modifications on model level and trigger them at system level. We suggest using models to describe manufacturing stations and their capabilities. Automatic station and topology detection functions can build up a model of the factory at run-time. The model of the whole factory with its stations and the possible material flows is then used by the MES to automatically calculate production schedules. To demonstrate the benefits of our approach, we use a modular production system where mechatronic compatibility is already provided. With this approach a fast adaptation and reconfiguration as described in Fig. 1 should be possible with minimal configuration effort on IT level. It should be possible to easily change the production system to produce different products by adding and removing stations, which are a modular mechatronic part

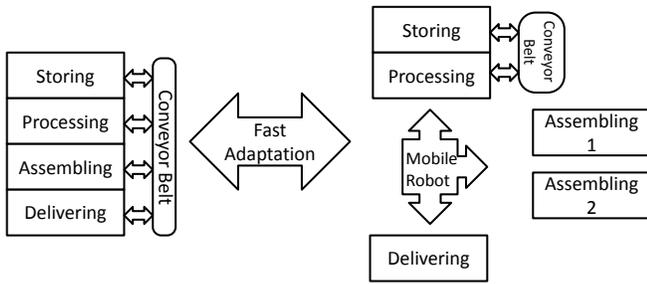


Fig. 1. Adaptation to production changes. The stations and the production line are reconfigured to increase the production volume.

of the system with a specific task. Rearranging the factory should easily be possible as well. Additionally, configuration effort and changeover time are minimized by this approach.

We first start with an evaluation of existing approaches and related work in the field of adaptable manufacturing systems in Section II. The first contribution of this paper is using a model @ run-time-based approach to describe station types and their capabilities. The different models we use for this are described in Section III. The second contribution is the automatic detection of stations and their topology in a factory. The collected information is then automatically forwarded to the MES. Section IV explains the details and how models can be used during run-time to configure and monitor the system. The industrial production system is described in Section V. We can show that manual efforts for setting-up and reconfiguring factories and process changeover times can be reduced. Section VI summarizes the paper.

## II. RELATED WORK

There has been plenty of work in the field of changeable and reconfigurable manufacturing systems due to the increasing importance of this topic for future manufacturing. In the following some approaches to achieve adaptability are explained.

*Neugschwandner* [6] proposes a plug & play approach tailored for home and building automation. Similar to our approach he suggests using self-description methods and discovery algorithms to integrate new devices. However, the main focus lies on the process automation for homes and buildings, whereas our approach addresses manufacturing processes. Additionally, we suggest the use of a model-based approach to facilitate the integration of new elements.

The approach described by *Naumann et al.* [7] introduces the concept of capability descriptions to automatically integrate new devices. Unlike our approach, they only use the approach for robot cells and do not extend it to include the rest of the manufacturing system. Besides, they focus more on reducing the programming effort, whereas our approach aims at reducing configuration time and increasing the adaptability of production systems. In addition, our approach uses models to encode capabilities, which can then be used to automatically configure the different manufacturing stations.

*Horbach et al.* [8] propose the concept of building blocks to design and plan adaptable factories. In addition to the different focus, their approach differs from ours in the sense that information is exchanged through RFID tags in the products, whereas our approach does not depend on a specific data exchange technology. We abstract from communication technologies by using a middleware. This makes our approach suitable for both centralized and distributed control.

The approach of *Reinhart et al.* [9] describes the automatic configuration of industrial Ethernet networks. For this, capabilities are described in a device description file. An automatic discovery can then integrate new devices into the system. This is similar to our approach, but limited to Ethernet networks. Our approach is not restricted to a specific communication technology since we rely on a middleware to abstract from communication technology.

## III. SYSTEM MODELING

As stated in the introduction our goal is to reduce the construction/adaptation effort when setting-up or modifying production systems. To achieve this goal, a model describing the internal structure and the supported processing steps of stations are required for each kind of station in the factory. The information in these models in combination with a model of the current factory configuration can then be used during run-time to automatically schedule the production orders according to the available stations in the factory. In this section the different models used to describe the system and the product plans are explained in detail.

### A. Station Type Models

The station type models are used to describe the internal structure of different station types and their supported processing steps. To simplify the modeling of a station type the modeling is split-up in separate steps. Each of these steps handles a dedicated task and builds upon the result of the previous step as depicted in Fig. 2. Currently we support three steps: capability, module, and station type modeling.

*Capability Modeling:* The capability model forms the base of the station type modeling and is used to define all known processing steps. This can be, for example, drilling, weighing, measuring, assembling, and so on. In addition to the bare definition of capabilities, each capability can be enhanced with attributes. These attributes are later on used to limit the ability of a station type for performing processing steps. E.g., if a station is able to drill, there will be some restrictions with respect to the supported material, the size of the material, and the dimension of the hole that can be drilled.

*Module Modeling:* After all the capabilities are defined, modules can be constructed, which support one or more of these capabilities. Modules in our context refer to a collection of components which belong together to perform a certain processing step, e.g., a drilling machine and the linear axle with its end point sensors used to move the machine up and down. For each module, it is possible to specify to which extent the capability is supported. By doing so, a misuse of the

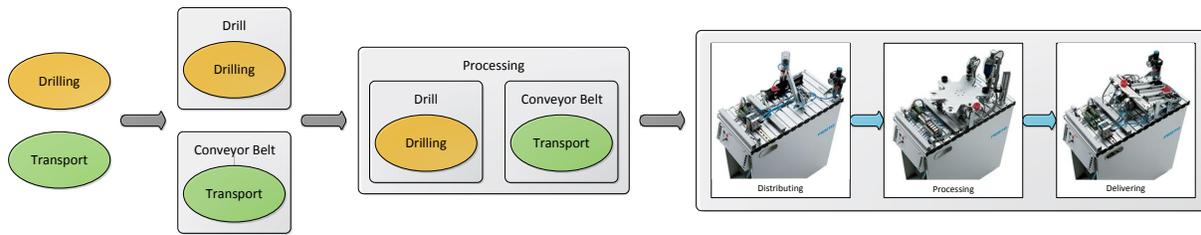


Fig. 2. Modeling of the station types and the production system. The grey arrows indicate the different modeling steps: Capabilities, modules, station types, and production systems with their stations.

module is prevented, because it is not possible to drill a 10 mm hole with a module which only supports holes with a diameter up to 8 mm. The modules also define their internal material flow, which is afterwards used to ensure that a path exists between the different modules scheduled for the production.

*Station Type Modeling:* In the last step station types can be constructed out of the previously defined modules as depicted in an example in Fig. 3.

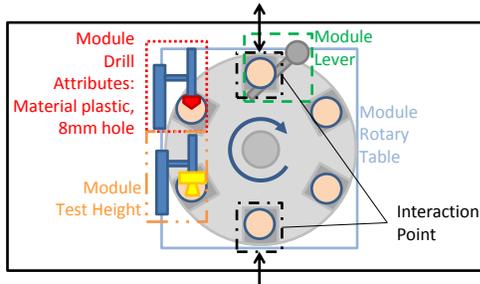


Fig. 3. The modules drill, test height, lever, and rotary table with their respective capabilities are combined to model the processing station type. Additionally, the station has two interaction points that define how this station type is connected to other station types.

We use model-to-metamodel (M2MM) transformations to combine the different models [10], [11]. M2MM transformations take model objects and create metamodel classes out of them. These classes can then be used to instantiate new objects. In this sense, the capability model is used to create parts of the module metamodel and the module model is used to create parts of the station type metamodel. By using M2MM transformations, changes in one model directly affect the structure of subsequent models.

### B. Factory Model

The factory model is used to represent the current factory configuration. This includes what kind of stations are available, how many of them, and how they are connected to each other. The stations can be either arranged in a line or as manufacturing cells. In a line, one station is directly connected to another station. When one station outputs a workpiece the other station will receive it as input. If they are arranged as manufacturing cells, a material transportation system is used to fetch and deliver the workpieces. For example, the transportation system can be a conveyor belt or a mobile robot. In contrast to the other models, the factory model is

automatically constructed during run-time by the system to reflect the current situation in the factory. An example of the factory model of our demonstration setup is illustrated in Fig. 4. The construction of the factory model during run-time is described in more detail in Section IV.

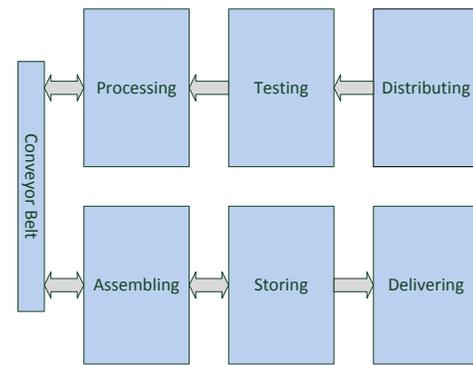


Fig. 4. Stations and material flow used in the demonstration setup.

### C. Product Plan Model

The last model is the product plan model, which specifies how different goods are to be manufactured. A product plan for a specific good contains the required production steps and the order in which they have to be performed. Production steps are attributed with properties, e.g., size and weight of the workpiece. This information is used to check whether a concrete station is able to perform a certain processing step on the current workpiece.

The station type model, the factory model, and the product plan model are then used to automatically schedule product orders in the factory and validate that all processing steps for a good can be performed in the current factory setup. If the validation fails, a message is triggered to inform the operators about the cause of the problem. For example, a product cannot be produced due to a missing transportation possibility from one station to another. In this case, the operators would be notified about the missing link, so that they can work on a solution for this problem.

## IV. MODELS @ RUN-TIME

This section describes how the capability model, the module model, and the station type model are used during run-time

to derive information about the factory and how to construct the factory model. The factory model contains the information needed by the MES and has to be kept up-to-date to ensure correct and immediate production planning. The factory model has to reflect the state of the factory at every point in time. To build up the factory model, two steps are required. In the first step, information about available stations and their types is gathered. How the available stations are connected with each other is then determined in the second step. In the following these steps are described.

#### A. Station Detection

To have an up-to-date model of the current factory setup, the system needs to be informed about new as well as no longer existing stations. This information is then stored in the factory model described in Section III. To reduce manual efforts, this step is done automatically. The stations periodically send liveness notifications containing their station identifier to the master control system. The master control system manages all received liveness notifications within a given period of time and checks for new stations. It additionally checks whether it has received a liveness notification from all known stations to detect station breakdowns. If a station is in an error state or a maintenance state, it can also indicate this to the master control system through defined signals. The factory model is updated according to the observed station states. This ensures that the factory model only includes stations that are available and ready for production preventing production time loss. In addition, when a new station is detected the system asks the new station about its type. The type is used during planning to reason about the production steps that can be performed by the station. All the information is then forwarded to the MES to adapt the planning to the current factory setup.

Currently, different strategies are supported to detect stations. Either a station is capable of sending a liveness notification directly to the master control system informing it about its existence or the master control system has to periodically poll for stations. The concrete realization depends on the used hardware controllers and communication technologies, but most communication technologies provide the possibility to list all connected communication participants.

To lower resource requirements on used controllers, the station type is encoded via a unique station type identifier. This identifier can be used to look-up the related station type description. On powerful controllers it is even imaginable to store the station type description directly in the controller and send it to the master control system on demand.

#### B. Topology Detection

In order to generate valid production schedules, knowledge about the setup of the factory has to be present. This setup refers to the position of stations as well as their connections to each other within the factory. The position provides information about neighboring stations if the stations are arranged in a line. If a material transport system, such as a mobile robot or a conveyor belt, is used, the topology additionally includes

all stations that can use the material transport system and their attachment points. Acquiring information about the topology of a production system enables reasoning about possible material flows in the production system. The material flow information can then be used to generate feasible schedules for the production.

To reduce manual effort, we suggest an approach for automatically detecting the topology of a production system. For this, we add a neighborhood detection state in the control program of every station. A station can switch to the neighborhood detection state whenever its current state allows for a short interruption. The stations are interrupted to prevent any damages of produced goods or the used infrastructure. Alternatively, the topology detection can be performed in parallel to the production process, if the station controllers allow this. The detection consists of three steps:

- 1) Switch to neighborhood detection state
- 2) Start discovery algorithm
- 3) Update factory model

Whenever a new station is integrated into the system, the master control system signals to all involved controllers that a switch to the neighborhood detection state is necessary. The switching signal from the master control system does not have to occur immediately after a new station is integrated. Typically, the system would wait for a short while to check whether other stations are also integrated to perform the topology detection only once for several stations. As a result, production downtimes are kept at a minimum. As soon as all the stations have entered the neighborhood detection state, the topology can be determined. To prevent a complete stop of the production, only stations that might be affected by the change need to be notified and triggered to switch to the neighborhood detection state. The affected stations are the ones belonging to the same production line or production hall. The discovery is achieved by sending out neighborhood detection signals and determining the receivers of the signals. In our case we use an optical connection between the stations. This connection consists of an infrared light emitter connected to one station and a corresponding receiver attached to another station. Other possible solutions could be hard wired or dedicated plug connections between stations attached to their interaction points. When a mobile robot is part of the material transport system used in the factory, the robot can be used for the topology detection as well. The robot starts searching for available stations in the defined production area to gather topology information. Each station has a unique identification mark through which the robot can identify the type of the station. The robot then stores the location of the station and builds up a map of the factory. A reachable station indicates that the robot can be used to fetch / deliver workpieces from / to this station. In our scenario we use optical markers as unique IDs, but RFID tags or other techniques can be used as well. If a station has no neighbors yet or cannot be reached by a mobile robot, no information about the topology is inferred and the station is considered isolated. When the discovery step

is done, the acquired information is used to update the data set in the MES to reflect the current factory setup.

To have a complete model of possible material flows, information about whether a material flow is bidirectional or unidirectional is required. Furthermore, the direction of unidirectional connections is required. This information cannot be retrieved through topology detection only. Instead, the type of the station and the information encoded in the corresponding station type model can be used to determine possible directions of the material flow. This is achieved by defining interaction points. Interaction points are places where material can be exchanged between stations. Their definition specifies if they can be used as input points, output points, or both. To determine the possible material flow between two stations, the corresponding interaction points have to be looked-up to calculate the feasible material flow. Thereby, it is possible to detect misconfigurations, e.g., the interaction points of the connected stations are both only input points.

For example, if we connect a processing station to a conveyor belt, the topology detection would only know that they are connected without knowing the supported directions for material exchange. By looking-up the used interaction points in the corresponding station types it can determine that both are defined as in- and output, which results in a bidirectional material flow.

### C. Integration with the Manufacturing Execution System

The factory model is used to update the data set used by the MES. Together with the production plan model it is the foundation for the planning performed by the MES. The MES has the information about available stations from the factory model and can thus look-up which operations can be performed by the current configuration. Additionally, it decides which operations are scheduled on which machine. If several stations can perform an operation, the MES schedules the fittest one. Fitness varies depending on system-defined criteria. Possible criteria are, for example, makespan, utilization rate, energy efficiency, delivery time, or combinations of them. However, optimizing scheduling algorithms computed by the MES is a very challenging task and is computationally hard. There is a lot of research in this area and the topic is beyond the scope of this paper. Apart from these criteria, the MES has to choose a set of stations that are all connected to each other to ensure that a material flow is possible. The required operations to produce a certain product can be obtained from the product plan model. Using this information together with the topology information, the MES can search the solution space to match required operations with available operations while adding necessary transport operations in-between. During the calculation of the production schedule, the MES ensures that the next operation is only scheduled on a machine that can be reached from the previous one. This guarantees feasible schedules with respect to material flow. After planning the production, the MES sends the processing instructions to the selected stations, and hence starts the production. Details of this process are out of scope of this paper.

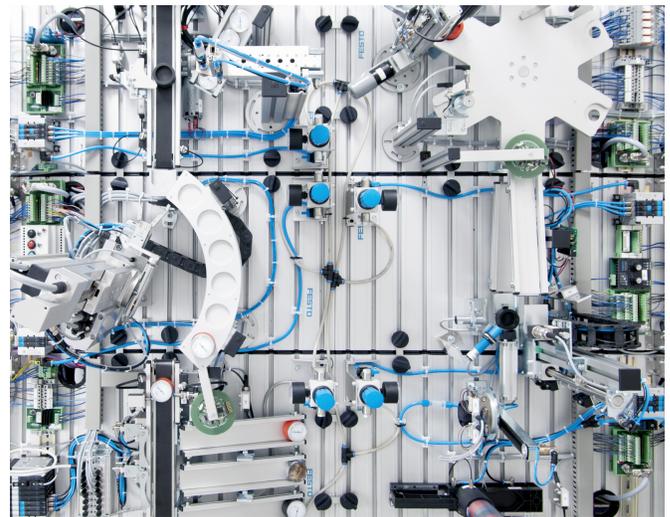


Fig. 5. Setup of the production system. This setup is used to produce black, red, and silver temperature sensors.

The station and topology detection are responsible for updating the factory model. The changes (e.g., adding and removing of stations, connections between stations) are then automatically forwarded to the MES. This ensures that the MES always operates on the up-to-date setup of the factory.

## V. DEMONSTRATION SCENARIO

To illustrate the approach, we use a simplified example from the automation domain used for educational purposes. The setup is a Festo modular production system shown in Fig. 5. Hence, mechatronic modularity and compatibility is provided and we can focus on the related issues in the IT domain.

The setup consists of six different stations and one conveyor belt. The manufacturing system produces temperature sensors in three different colors: black, red, and silver. The different steps required for such a production include distributing of material, testing, processing, assembling, storing, and delivering. The setup is depicted in Fig. 4. This figure also shows which stations are connected to each other and whether the material flow is bidirectional or unidirectional. The possible directions of the material flow are indicated by the arrows.

Each of the stations is controlled by a programmable logic controller (PLC) that performs its production steps. The PLCs are independent of each other and only exchange simple I/O signals. They are all connected via a proprietary multi point interface (MPI) bus to exchange information. The conveyor belt is controlled by a microcontroller. Additionally, a PC is used as a centralized master control system that realizes some of the MES functionalities. The master control system is connected to both the production system and the microcontroller. Its main tasks are to coordinate stations, monitor available resources, and plan production. To simplify the setup and save resources, the station detection and the topology detection also run on the computer used as master control system. We use a graphical user interface (GUI) to monitor the production system that shows the current setup of the factory. Adding a

physical station to the factory is immediately reflected in the GUI as a new node annotated with the corresponding station type. Whenever a station is removed from the system, the corresponding node is removed in the GUI, validating that the factory model has been updated correctly.

To update the factory model of the MES, the station detection and the topology detection have to communicate with the MES. Since we support different platforms, we use a middleware to abstract from communication technology. Thus, they can run on the same machine or can be distributed on several machines. The middleware is used for vertical communication from MES to shop floor (e.g., the different controllers of the stations). The horizontal communication between different elements of the system, such as the different stations or the MES and the GUI, is handled by the middleware as well. Using the middleware, we support different platforms creating a heterogeneous setup. The controllers can be PLCs, microcontrollers, industrial computers, or standard PCs. Moreover, heterogeneous communication interfaces such as Ethernet and MPI through a gateway can be used. We use the CHROMOSOME middleware<sup>1</sup> for this purpose. CHROMOSOME is a data-centric middleware with a publish-subscribe concept that abstracts from specific senders and receivers. Publication and subscription information is propagated through the system and corresponding routes are established through the middleware. This reduces the effort of configuring communication channels manually and enables the plug & play concept at communication level.

In addition, we support virtual stations that include the software part but are not connected physically to the system. This setup allows testing different scenarios and facilitates analyzing the plug & play concept without having to move the physical stations around. The virtual stations are also connected to the master control system and communicate with it through the middleware.

*Evaluation:* With this demonstration setup we show that heterogeneous platforms can be integrated into a system and configured for use in the factory. Without any manual effort, new stations are added to the data set of the MES and are automatically integrated into the planning system as soon as they show up. By using models to describe capabilities and topologies, all the required information is easily accessible by the MES for reasoning and production schedule generation without further interaction with the operators. The engineer has to define a station type model only once for each type of station and can reuse them in different systems and factories. This reduces setup time of production systems resulting in faster process changeover times. Since the models, the station detection, and topology detection are reusable, the configuration effort is less error-prone, which further reduces the configuration time. Moreover, the engineer can focus on the design of the control of the factory and no longer has to worry about each configuration step.

<sup>1</sup>CHROMOSOME middleware: <http://chromosome.fortiss.org/>

Since the factory model reflects the current factory configuration at each point in time, the MES can quickly react to machine breakdowns and changes in the manufacturing system. This can be done without any further configuration effort making the system more adaptable.

## VI. CONCLUSION

In this paper we proposed a models @ run-time-based approach to enable plug & play in the automation industry at IT level. This approach helps reducing manual efforts of setting-up and reconfiguring manufacturing systems making them more adaptable to changes. We presented how models can be used to describe capabilities of stations. Furthermore, these models are then used during run-time to reflect the current setup of the factory. The MES can then use this information for monitoring the setup of the system and planning the production schedule. We used a simplified industrial setup to show how our approach can be applied. This setup demonstrates that we support heterogeneous platforms with our approach and that we abstract away from different communication interfaces by using a middleware. Furthermore, the manual efforts were reduced, since all the steps required for configuring or reconfiguring the system were done automatically during the detection cycles, which we proposed in this paper.

## REFERENCES

- [1] H.-P. Wiendahl, H. ElMaraghy, P. Nyhuis, M.-F. Zäh, H.-H. Wiendahl, N. Duffie, and M. Bricke, "Changeable Manufacturing – Classification, Design and Operation," *CIRP Annals-Manufacturing Technology*, 2007.
- [2] M.-F. Zäh, M. Beetz, K. Shea, G. Reinhart, O. Stursberg, M. Ostgathe, C. Lau, C. Ertelt, D. Pangercic, T. Rühr *et al.*, "An Integrated Approach to Realize the Cognitive Machine Shop," in *Proceedings of the 1st International Workshop on Cognition for Technical Systems*.
- [3] O. Sauer and J. Jasperneite, "Adaptive information technology in manufacturing," in *CIRP Conference on Manufacturing Systems*, Madison, WI, USA, Jun 2011.
- [4] B. Vogel-Heuser, G. Kegel, K. Bender, and K. Wucherer, "Global Information Architecture for Industrial Automation," *ap*, 2009.
- [5] *Cyber-Physical Systems: Driving Force for Innovation in Mobility, Health, Energy and Production*, ser. acatech Position. acatech – National Academy of Science and Engineering, 2011.
- [6] G. Neugschwandtner, "Towards Plug and Play in Home and Building Automation Networks," in *IEEE Conference on Emerging Technologies and Factory Automation. ETFA'06*. IEEE, 2006, pp. 461–464.
- [7] M. Naumann, K. Wegener, and R. Schraft, "Control Architecture for Robot Cells to Enable Plug'n'Produce," in *IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 287–292.
- [8] S. Horbach, J. Ackermann, E. Muller, and J. Schutze, "Building Blocks for Adaptable Factory Systems," *Robotics and Computer-Integrated Manufacturing*, 2011.
- [9] G. Reinhart, S. Krug, S. Huttner, Z. Mari, F. Riedelbauch, and M. Schlogel, "Automatic Configuration (Plug & Produce) of Industrial Ethernet Networks," in *9th IEEE/IAS International Conference on Industry Applications (INDUSCON)*. IEEE, 2010, pp. 1–6.
- [10] G. Kainz, C. Buckl, S. Sommer, and A. Knoll, "Model-to-Metamodel Transformation for the Development of Component-Based Systems," in *Model Driven Engineering Languages and Systems*, ser. Lecture Notes in Computer Science, D. Petriu, N. Rouquette, and y. Haugen, Eds. Springer Berlin / Heidelberg, 2010, vol. 6395, pp. 391–405, 10.1007/978-3-642-16129-2\_28.
- [11] G. Kainz, C. Buckl, and A. Knoll, "Automated Model-to-Metamodel Transformations Based on the Concepts of Deep Instantiation," in *Model Driven Engineering Languages and Systems*, ser. Lecture Notes in Computer Science, J. Whittle, T. Clark, and T. Khne, Eds. Springer Berlin / Heidelberg, 2011, vol. 6981, pp. 17–31, 10.1007/978-3-642-24485-8\_3.