Integration of a formalized process description into MS Visio® with regard to an integrated engineering process

Lars Christiansen, Tobias Jäger, Martin Strube, Alexander Fay
Helmut Schmidt Universität Hamburg, Holstenhofweg 85, D-22043 Hamburg, Germany
[lars.christiansen, tobias.jaeger, martin.strube, alexander.fay]@hsu-hh.de

Abstract

This paper describes an improvement of the engineering process of automated plants by combining information of the technical process and plant structure. Integrating a semi-formal process description in MS Visio® for the requirements engineering phases and using these planning results throughout the engineering process can be an approach for an integrated engineering. By linking the semi-formal process description with a plant structure description in AutomationML the engineering efficiency can be increased. The potential, arising through the presented approach, is shown by an example from the engineering of a hot strip mill. The requirements have been modeled within the MS Visio® implementation; these results can then be used for a consistent observance of the stated requirements on the technical resources that have to be engineered

1. Motivation

The planning process of automated industrial plants is characterized by interdisciplinary cooperation of different disciplines. The similarities in the engineering of manufacturing and process facilities are a breakdown into many phases with phase-related tool support and a lack of data exchange solutions [1, 15]. The planning time becomes shorter and across the disciplines and all project phases the communication extends and data exchange increases strongly [2]. Especially the exchange of information between the various disciplines is one of the biggest challenges, because each discipline's description has specific methods and specific tools engineering tasks [15].

Nowadays the necessary coordination between the engineers of different disciplines is often handled on the basis of a non-formal description of the realizable technical process [16]. Thus, this non-formal process description is a central document for system engineering. Beginning in the requirements elicitation the requirements are today written down in a non formal way, using textual descriptions and drawings prepared with tools like MS Word®, MS PowerPoint® and partly MS Visio®. Due to the different usage of various semantics some misunderstandings and misinterpretations are preprogrammed. Furthermore, the increasing usage of concurrent engineering in the planning process of automated industrial plants leads to the problem that upstream disciplines have to work with unsecured planning results. If these unsecured planning results change during the planning process, this may result in new requirements on the technical solution [16]. The combination of lack of data exchange solutions and a heterogeneous tool landscape prevents a transparent requirements management in the engineering of automated facilities. This results in an increased potential for errors in the engineering process. To counteract the aforementioned problems, in recent years some methods and approaches for a formalized process description and modeling of the plants topology were developed. These methods and approaches are already being used successfully in practice but isolated from each other.

To create the basis for optimal information exchange during the planning process, this article presents an approach for a holistic formal plant description. Furthermore, it will be shown how a transparent requirements engineering can be achieved by the use of this holistic formal plant description.

2. Process description in the Engineering Process

As introduced in the first section misunderstandings and misinterpretations, e.g. of the technical process, are pre-programmed if the communication is based on an informal description, e.g. textual.

For the description of batch processes in [17], formally known as ISA-88, there already exists an approach for the separation between process and resource. Also the assignment of the process structure to the plant structure is included herein. A suggestion for modeling and representation of the causal process flow is not described in [17].

For our purposes a general and formalized description is required which is domain independent (i), easily understandable to all engineers involved (ii) and usable throughout the life-cycle of the system (iii) [6].

Accordingl to [7] a "technical process" is defined as "a complete set of interacting operations in a system by which matter, energy or information is transformed, transported or stored".

2.1. Formalized Process Description with VDI/VDE-Guideline 3682

The formalized process description (FPB) offers the possibility to easily and understandably describe a process. Besides easy comprehension by all participants the process description provides process-relevant information throughout the entire life cycle of the systems in a clear and structured way [6].

For this the guideline defines a small number of symbols for the graphical description of the process. The classes of objects are: operators (=process operator, technical resource), states (=product, energy) and relations (=flow, utilization, system boundary) (Fig. 1).

A technical process is modeled inside of a system boundary. This enables the identification of the systems input and output variables and thus, the process which is carried out in the system. The advantage of defining a system boundary is the breakdown of a system into different sub-systems which results in a hierarchical structure of the process. This leads to a detailed description of the process and identification of interactions between the objects inside the system and its environment. In the guideline states define the input and output variables of the process and the process operators and are linked by directed arcs (=flow). With the process operator a *pre*-state is transferred into a *post*-state. For carrying out the process a technical resource is associated with a bidirectional correlation (U).

The formalization is supported by rules defining in which manner states and operators are interlinked with each other inside the system boundary.

The technical process is first described by a graphical model in which states and operators are defined and linked with flows. By decomposing process operators a detailed description of the process is possible. The decomposition of the process supports a structured proceeding and results in a hierarchical process model. Therefore the FPB supports the creation of a graphical flow of the process which is more understandable than a textual specification as it is performed today. Besides detailing a process by decomposing, states and operators can be detailed by assigning attributes.

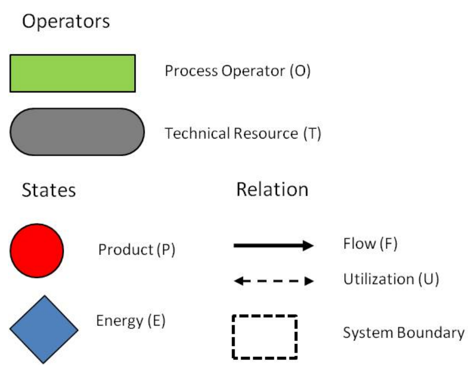

Figure 1. Graphical symbols of the guideline [6]

Attributes are next to the name and a unique ID process-specific data such as duration of a step, necessary pressures, temperatures, as well as material and energy quantities. These data provide the essential information of the process and are stored and managed in an information model. The information model also contains the causal relations between states and operators (Fig. 2).

The relational part describes e.g. different views on the technical process like logistics, asset management or security [6].

2.2. Advantages of the Formalized process description for modeling process requirements

The graphical description of the technical process is intuitive, easily understandable and comprehensible to all participants without having extensive previous knowledge of the process. The separation within the process description between products (=states), processes (=process operator) and resources (=technical resource) provides a solution-neutral use at the beginning of the engineering process. More important than the graphical information are the process-specific information like pressures, temperatures or product quantity. These values of the states and operators enable the derivation of requirements to the later technical resource. Therefore at the beginning of the engineering process the technical resource is assigned a role that serves as a placeholder in advance. This role is gradually substituted in the engineering process through the concrete plant component, which meets the requirements of the role [9]. For computer-based application and exchange of relevant data and information of the process a computer-readable exchange format is needed. For the seamless exchange of engineering data in the field of automation the description language XML (eXtensible Markup Language) has become de-facto standard [8].

Existing software tools which are available for using the FPB do not support an XML-based exchange of re-

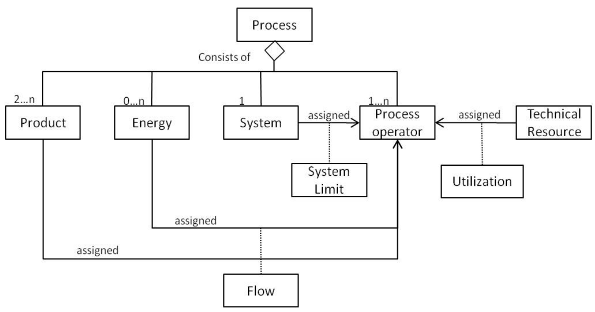

Figure 2. Information model of the FPB [6].

levant process data. Therefore, the institute of the authorsinvestigated MS Visio® for implementing the information model which is presented in section 5. The advantage of MS Visio® is the possibility to store information in an XML data format which offers potentials for a consistent data exchange with other CAE-tools (section 5).

3. Integration of the FPB information model in MS Visio®

The tool MS Visio® has already been successfully used in the engineering process in many ways. Therefore it offers different Shape-Galleries for modeling P&ID (Process & Instrumentation Diagrams), circuit diagrams or various flow charts. In addition to the predefined shape-galleries it is possible to develop individual galleries according to one's own needs. This provides a basis for adjusting the requirements of the formalized process description into MS Visio® with Visual Basic for Application (VBA).

3.1. MS Visio® Shape-Gallery

Within MS Visio® graphical objects (=symbols) are provided in so called shape-galleries which act as classes for instantiating objects. The required symbols are taken by "drag&drop" onto the worksheet and finally linked to each other.

To describe processes with the formalized process description in MS Visio® the authors implemented a shape-gallery with the symbols and rules defined by the guideline (Fig. 1). For data management MS Visio® is based on so-called *shapesheets* which are comparable to excel sheets. These *shapesheets* contain general information of objects like size, color, coordinates of textual content (=graphical information) as well as information defined by user. This allows assigning additional information like predecessor and successor relationships of an object (=causal relationship) or assigning characteristics defined by the formalized process description (Section 3.1). The possibility to store information in shape-sheets creates an extensive information model.

Besides the guideline-specific functions e.g. rules for linking symbols, assigning of attributes the implementation of tool-specific functions increases the usability of the tool. With the representation of the hierarchy the user gets a comprehensive overview about the process structure with its decompositions. Furthermore, with the possibility to assign different views like quality or safety to each attribute only relevant information for the engineering process can be presented. This provides the possibility to derive only those requirements which are important for the engineer.

3.2. Data exchange with MS Visio®

Due to a reduced period of planning a cross-trade exchange of information based on XML becomes increasingly important. With the use of MS Visio® for modeling and describing technical processes according to VDI 3682 it is now possible to provide process-specific data and information to different CAE-Tools as well as AutomationML.

Besides process-specific information like sequences and duration of process steps or type of input/output products the process description provides information for basic engineering like requirements engineering which is discussed in the following section. Furthermore, causal information about the process can be used during the whole life cycle of the plant e.g. for diagnosis actions in the operational phase of the plant [10].

4. Bridging the gap between process and structural description

The integration of process description and plant structure information, in just one model will have a significant influence on the engineering results. Through the integration of the neutral process description with a multi-disciplinary plant structure description, the consistency in engineering can be improved.

Although on the one hand the formalized process description provides references to the technical resources and thus describes the plant structure and on the other hand AutomationML/CAEX provides a hierarchical structure, both models are used isolated nowadays. This contradicts the basic idea of an integrated engineering process.

5. AutomationML/CAEX

This section gives an outline of the concept and terms of the data format AutomationML and CAEX. Furthermore the concepts of AutomationML, mainly derived from CAEX, used for the approach are described in detail.

5.1. AutomationML

The aim of the neutral data format AutomationML™ is the storage and exchange of complex engineering data within planning phases to bridge the gap between separate and phase-specific CAE-Tools within the engineering process. The problem is enhanced because during the engineering the data are subjected to a continuous and iterative enrichment and change. Therefore, a continuous exchange of engineering data by the vendor-independent data format AutomationML™ was developed which is based on the concept of the CAEX format and supports object-oriented aspects like re-use and inheritance [3].

Besides the storage of the plant topology AutomationML supports the integration of additional engineering data. Therefore AutomationML references to other exchange data formats [3]. In the field of robot planning information about geometry and kinematics can be stored in AutomationML by using COLLADA™.

COLLADA™ defines a XML-scheme to exchange 3D plant information between different 3D applications [12]

For storing associated behavior of e.g. the handling device AutomationML supports the integration of programmable logic control (plc) data and information by referencing to PLCopen XML. PLCopen XML was developed to exchange project data and libraries of plc programs between different environments [13].

Storing data about plant structure, geometry and kinematics and behavior offers a possibility to exchange complex engineering results. For the described approach only the opportunity for the storage of plant structure with CAEX plays an important role.

5.1. CAEX (Computer Aided Engineering eXchange)

The meta model CAEX sets the definition of structure and storage of objects with their properties and relationships. CAEX is the basis of a general data format for the storage and exchange of complex engineering data. CAEX defines four fundamental elements, which are described in the following.

The *SystemUnitClass* describes physical or logical plant objects and units including their technical realization and internal structure with a reference to specific roles from the *RoleLibrary*. A plant object consists of different attributes, interfaces, nested internal elements as well as internal connections. Attributes including e.g. value, unit and constraints like maximum or minimum value.

The *InterfaceLibrary* comprises different types of interfaces. Interfaces are needed to define connections between objects e.g. for product, signal or information flow.

Objects, derived from the *RoleLibrary* are used for an abstract description of physical plant objects. The use of roles within the engineering process supports the independent description of the concrete technical realization of the objects' functionality. Roles are used as

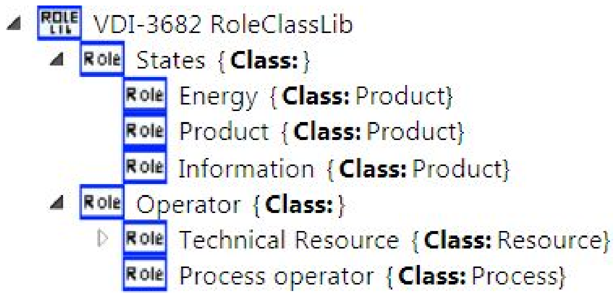

Figure 3. RoleLibrary of VDI-guideline 3682

placeholders at the beginning of the engineering process and are consecutively specified and replaced by the concrete object. Furthermore, roles can be used to determine requirements which the realization has to fulfill.

The *InstanceHierarchy* enables the description of the plant's topology with all its components, instantiated from the *SystemUnit,* and all physical and electrical connections. Furthermore, roles from the *RoleLibrary* can be referenced to the objects in the instance hierarchy to describe their function [5].

6. Integrating the FPB in AutomaionML/CAEX

The first step to integrate the formalized process description in CAEX is done by the expansion of a role library derived from the directive containing roles shown in Fig. 3.

Since the objects of the formalized process description are connected through flows and utilization the necessary interfaces have to be defined in the interface library. One approach is to expand the interface library with the interface classes *process-product*, *process-energy*, *process-information* (=flow) and *process-resource* (=utilization). With these additional role and interface classes the formalized process description can be mapped in AutomationML [9].

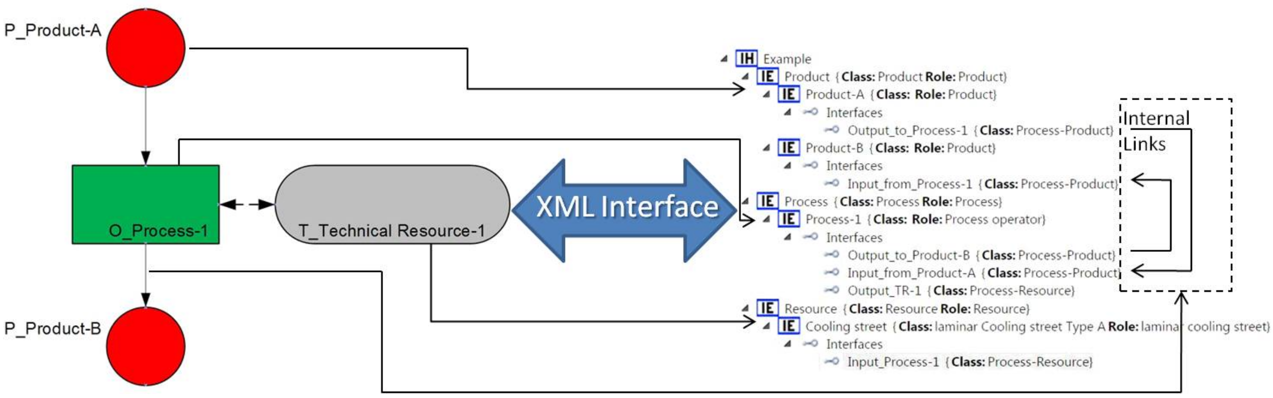

Figure 4. Mapping of the graphical process description to AutomationML

6.1. Mapping of the process description in AutomationML

The integration of roles and interfaces into an AutomationML library offers the possibility of transferring the graphical relationship as well as process-related information into a structural description with CAEX. As the FPB AutomationML also supports the separation of the three aspects *product*, *process* and *resource* as it can be seen in Fig. 4 [11]. Therefore the graphical objects of the process description in the left part of Fig. 4 are already assigned to the views in AutomationML on the right side. The interfaces derived from the corresponding interface class build knots for modeling the input and output relation of the process. Due to lack of space the representation of the relational part with InternalLinks (IL) between the InternalElements (IE) has been dispends with.

As shown in Fig. 4 a transfer medium like XML is required for an automatic assumption of process information.

7. Use Case: Integrated Engineering

Based on the concept presented in section 4 the following section describes which possibilities arise from this proceeding for a software assisted requirements elicitation and requirements management. For a better understanding this will be shown by the simplified example of a steel strip production model. The following use case deals with the planning of the laminar cooling section of a hot strip mill.

The task of a laminar cooling section is to cool the hot rolled steel strip distributed locally to adjust the desired material properties. Therefore the steel strip is applied selectively with water from both sides. Depending on the material properties to be achieved, different requirements for the technical realization of the cooling section arise. If the plant operator decides to modify the production process during the planning phase, e.g. to expand the product range, this leads to new requirements for the technical realization of the cooling section.

If the concept described in section three is implemented, this provides not only the ability to specify the resulting requirements from the process description formally, but additionally the option, to match this automatically with the parameters of the chosen technical implementation and point out the discrepancies that occur to the planning engineers (requirements management).

7.1. Requirements modeling with the FPB

At the beginning of a project the technical process will be described with the formalized process description. For this purposes each process step, e.g. cooling the steel band, is represented by a process operator.

As described in section 3 a technical resource is assigned to each process operator in the process description. The technical resource is a placeholder for particular types of technical realization. The attributes of these placeholders describe the requirements which have to be fulfilled by the technical realization. After these requirements have been modeled in MS Visio®, the results can be exported into a CAEX file. The placeholder of the technical resource of the FPB is represented as an Internal Element and is connected with the process operator in CAEX as exemplarily shown in Fig. 4. Furthermore a predefined role from the Role Class Library is assigned to this Internal Element and the previously defined attributes are transferred to the role requirements of this Internal Element. This procedure creates a formal description of requirements of the technical process. Through the mapping into CAEX these process requirements can be reused by the following disciplines for their subsequent tasks. Fig. 5 summarizes this with an example of the role requirements of the laminar cooling section.

7.2. Synergy by bridging the gap in the planning process of a plant

Based on the results of the requirements modeling the selection of appropriate technical resources has to be supported. Therefore the engineers compare the defined attributes in the role-requirements of the placeholders with the available technical resources. When those resources are part of the unit library, this comparison can be automated. In the next step the placeholders can be replaced by instances of the Unit-

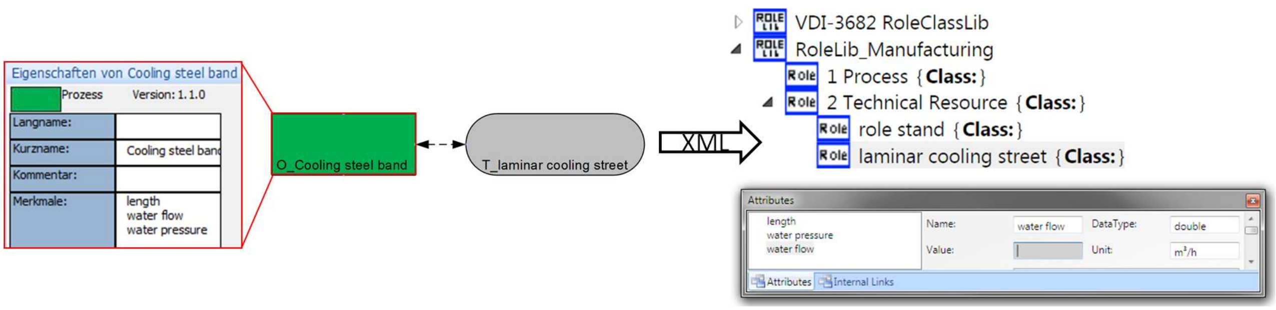

Figure 5. Mapping of the graphical process description to AutomationML

library. The predefined role-requirements remain with the Internal Elements.

Changes in the project, e.g. expanding the product range, which though leading to new requirements create a discrepancy between the requirements of the roles and of the technical realization exemplarily shown in Fig. 6.

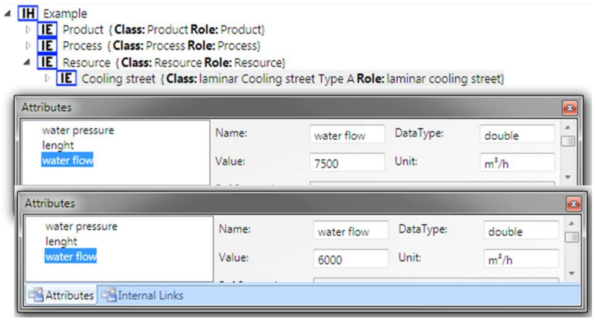

Figure 6. Discrepancy between requirements and attributes of the technical realization

The model-based merging of the resulting requirements from the process description with the information of the technical resources contained in the plant forms the basis for a software based support of requirements management. As shown in [11] ruled based queries are appropriate to identify inconsistencies like this in the plant model. The following rule points this out in an example: IF the attribute water flow in the RoleRequirements of an Internal Element is higher than the attribute water flow of this Internal Element, THAN an unsuitable technical resource was chosen.

A software assisted requirements management based on a neutral data exchange format can lead to a significant reduction of possible mistakes in the plant engineering process, without losing discipline specific tool support.

8. Conclusion

Within this article an approach for bridging the gap between process and plant description based on a formalized process description is presented. It has been shown how to integrate a semi-formal process description in MS Visio® and how to make the results automatically available in the neutral engineering data exchange format AutomationML. Based on this it was shown how the requirements resulting from the process description can be matched with the properties of the technical resources of the plant model which allows a software assisted requirements management

References

[1] R. Drath: “The future of engineering: challenges to the engineering of manufacturing and process plants” (in german) , *In: Karlsruher Leittechnisches Kolloquium* 2008, pp. 33–40, 2008

[2] S. Brandner:” Integrated product data and process management in virtual factories”, iwb Forschungsbericht -Band 136, Herbert Utz Verlag, Munich, 2000 (in german)

[3] M. Fedai, U. Epple, R. Drath, A. Fay: “A Metamodel for generic data exchange between various CAE Systems” *In: Proceedings of 4th Mathmod Conference, ARGESIM Report*, vol. 24, pp. 1247 - 1256, Vienna, February 2003

[4] R. Drath, A. Lüder, J. Peschke, L. Hundt: “AutomationML - the glue for seamless automation engineering”*: In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA),* pp. 616–623, 2008.

[5] IEC 62424: Specification for Representation of process control engineering requests in P&I Diagrams and for data exchange between P&ID tools and PCE-CAE.

[6] VDI/VDE-guideline 3682: “Formalized process description” 2005

[7] IEC 60050-351: Control technology

[8] M. Wollschläger and P. Wenzel: “Common model and infrastructure for application of XML within the automation domain”*: In: IEEE International Conference on” Industrial Informatics*”, pp. 246–251, 2005.

[9] M. Strube, A. Fay: “Bridging the gap between process and plant description*”* atp-edition, no. 9, pp. 26–27, 2010. (in german)

[10] L. Christiansen, A. Fay, B. Opgenoorth, J. Neidig: “Improved Diagnosis by Combining Structural and Process Knowledge”, *In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Toulouse, France, 5.-9. September 2011

[11] M. Strube, S. Runde, H. Figalist, and A. Fay: “Risk Minimization in Modernization Projects of Plant Automation – a Knowledge-Based Approach by means of Semantic Web Technologies*”*, *In: IEEE International Conference on Emerging Technologies an Factory Automation (ETFA)*, Toulouse, France, 5.-9. September 2011

[12] AutomationML™: Whitepaper AutomationML Part 2 - AutomationML Libraries

[13] COLLADA™: https://collada.org

[14] PCLopen: http://www.plcopen.org

[15] T. Jäger, A. Fay, T. Wagner, U. Löwen: Mining technical dependencies throughout engineering process knowledge, *In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Toulouse, France, 5.-9. September 2011

[16] M. Föhr, A. Lüder, T. Wagner, T. Jäger, A. Fay: Development of a method to analyze the impact of manufacturing systems engineering on product quality, *In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Toulouse, France, 5.-9. September 2011

[17] IEC 61512-2: Batch control. Part 2: Data structures and guidelines for languages