Available online at www.sciencedirect.com
ScienceDirect
IFAC PapersOnLine 50-1 (2017) 7584–7591

Auto-Generated Specification of Assembly Units by Formalized Requirements for a higher Maturity in the Engineering Process

Andreas Schlag * **Michael Vielhaber** **

* *CoC Emission Free Mobility, Daimler AG, 68305 Mannheim, Germany (e-mail: andreas.schlag@daimler.com)*
** *Institute of Engineering Design, Saarland University, 66123 Saarbrücken, Germany (e-mail: vielhaber@lkt.uni-saarland.de)*

Abstract: Today's engineering work flow of resources, e.g. units of automotive assembly lines or robotic cells in the body shop, shows up some gaps in provision of information in different engineering disciplines like mechanical-, electrical- or software engineering. The common issue is addressed by the kind of description of the technical and functional needs that mostly production planners are using in the planning phase. This article focuses the gap of describing required resources and introduces a method to auto-generate formalized object-related requirements by using a formalized process-description language. The general functionality of the method is presented by two use cases. Further, a way to exchange and provide this auto-generated engineering data in the engineering work flow is introduced by a suggested implementation in ReqIF. Towards seamless engineering data provision a call for integration of the ReqIF in AutomationML is given.

Keywords: Smart Structures; Design methodologies; Mechatronic systems; Modeling; Modeling of assembly units; Production planning and control.

1. INTRODUCTION

With the continuously rising demand of individual products, increasing assortments and the shorter time-to-market objective the product development process has changed and has to change to fulfill the needs of the customers. Through this change progress the development process of resources is affected too and the requirements on a production plant getting more complex to handle the different variants and derivatives which have to pass the production plant (See Drescher (2013)).

In this context a production plant has to be described, developed, build-up and start-up in an even shorter time span. With digital validation methods like virtual commissioning (VC) an theoretically efficient way to validate a production plant is given and respectively used to save costs and time during the commissioning of the physical production plant. With this method it is possible to validate the behavior of a production plant according to the available test program. Today's work flow to get those test programs is time-consuming and for the most part a manual kind of work. Well-founded in the missing validation criteria in a formalized format.As in Schlag (2016) mentioned the advantage of the method VC compared to the way to get the validation model is in most case lost.

This paper focuses the gap of describing required resources and introduces a method to auto-generate formalized object-related requirements by using a formalized process-description language to get validation criteria in a formalized format. The general functionality of the method is presented by two use cases. Further, a way to exchange and provide this auto-generated engineering data in the engineering work flow is introduced by a suggested implementation in ReqIF. Towards seamless engineering data a call for integration of the ReqIF in AutomationML is given.

Fundamental for the study the following Hypothesis is served:

Hypothesis 1. With a model-based description of requirements for a production plant the development process of resource can be more efficient.

To confirm this hypothesis, the following five questions emerge as research questions:

(1) What possibilities and approaches exist with regard to the model-based plant specification?
(2) How can a method promote the consistent availability of requirements in the development process of resources?
(3) How the development methodology of resources itself has to be changed in terms of tools, processes, methods?
(4) How can this methodology be able to generate start-concepts for the tools of the individual disciplines from the plant specification?
(5) How could the complexity of an installation (in focus on the virtual validation phase) be mastered?

* This work was not supported by any organization

2405-8963 © 2017, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2017.08.998

2. THE STATE-OF-THE-ART ENGINEERING PROCESS

In this section the state of the art of the plant development process will be described and with focus on the phases Requirements Engineering and - Management and Validation Methods the motivation will be introduced.

The plant development process can be divided in four different categories (See Figure 1):

(1) Conception
(2) Engineering
(3) Validation
(4) Administration/Operation

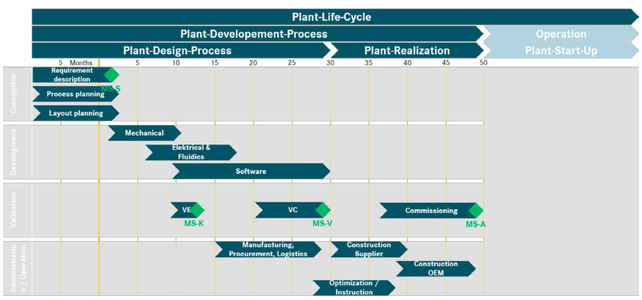

Fig. 1. The Plant-Developement-Process, Drescher (2013)

For the focus of this paper the two relevant methods Requirements Engineering and -Management and Validation Methods will be present in detail.

2.1 Requirements Engineering and -Management

According to Gilz (2014) Requirements Engineering obtains the development of requirements out of abstract specifications of the customer. It is an cooperative, iterative and incremental process with the aim to determine, analyse, understand and establish requirements.

In Mannion (1995) Requirements are defined as goals which are desired to be achieved and should be modeled as SMART requirements. SMART is the acronym for:

(1) S - specific
(2) M - measurable
(3) A - attainable
(4) R - realizable
(5) T - traceable

With using the SMART framework the requirement specification can be checked and verified as correct in terms of expression, see Mannion (1995).

In the last two decades the usage of Requirements Engineering increases and a lot of tools and framework were created and realized, but one fact wasn't considered. In all tools a requirement is build as an object with an id and some attributes, but the content is in most cases only a string. So it is impossible to verify this content. The are functions for dictionaries and keywords, but they have in common that they are software implemented and not part of the data model, see Gilz (2014), Hull (2011), Rupp (2014) and Pohl (2015).

In short, a requirements engineer has to set up a data type to define a glossary.

Through the fact that requirements engineering has its roots in software development the application in mechanical- or electrical engineering is not pronounced as it could be. E.g. in the automotive industry a specification of an production plant is written mostly as a text document in MS Word for example. With many pictures and the possibility of occurrence of a dialect. The technical and functional needs were described in the technical understanding of the planner which doesn't match the technical understanding of the engineer of the supplier. That's a main reason why the requirements engineering is a iterative and incremental process because of to clarification of misunderstandings.

There are existing templates to create a valid requirement. In Figure 2 the core of a requirement is shown.

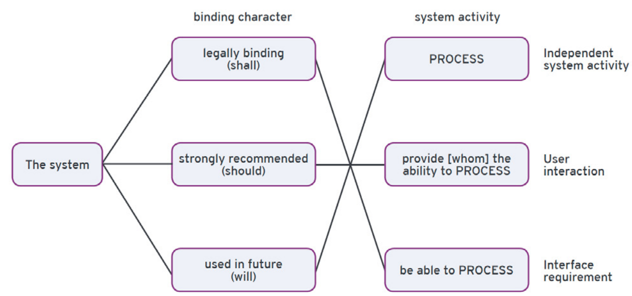

Fig. 2. The core of a requirement, Rupp (2014)

Another reason why requirements engineering doesn't spread in the mechanical- and electrical discipline is that there is no systematical usage in the further steps of the development process of resources.

2.2 Validation Methods

Validation Methods in this context are defined as Methods to verify that the production plant is able to fulfill required claims e.g. the described Process, Layout and other requirements in the context of the production plant behavior. For example a FEM analysis is not part of the category Validation, because it is an element of the phase mechanical engineering phase of the category Engineering, see Schlag (2016) and Drescher (2013).

The category validation includes three phases two virtual and one real validation method. In the context of the paper only the two virtual methods Virtual Engineering (VE) and VC are in focus.

These both methods are depending on milestones, see Figure 1. With the VE the supplier gets the release to start the physical construction of the production plant. At this point the mechanical setting fulfills the required claims without consideration of the control program. VE requires a kinematics model of the virtual production plant and an overview about the actuators in the model. With an simplified flowchart the process has to be simulated without any collision between the resource and the product as well as with there own components. The outputs are proofed cycle time, free of collision and a correct operation flow.

Within the phase VC the virtual model of the production plant is tested with the real control program running on