Lessons Learned from developing Industrial Applications according to RAMI 4.0 by applying Model Based Systems Engineering

Christoph Binder
Center for Secure Energy Informatics
Salzburg University of Applied Sciences
Urstein Sued 1, A–5412 Puch/Salzburg, Austria
christoph.binder@fh-salzburg.ac.at

Boris Brankovic
Center for Secure Energy Informatics
Salzburg University of Applied Sciences
Urstein Sued 1, A–5412 Puch/Salzburg, Austria
boris.brankovic@fh-salzburg.ac.at

Christian Neureiter
Center for Secure Energy Informatics
Salzburg University of Applied Sciences
Urstein Sued 1, A–5412 Puch/Salzburg, Austria
christian.neureiter@fh-salzburg.ac.at

Arndt Lüder
Otto-v.-Guericke University
Universitätsplatz 2
D-39106 Magdeburg, Germany
arndt.lueder@ovgu.de

***Abstract*—The transformation from original production line manufacturing towards complex value creation networks causes new challenges for manufacturing companies to stay in touch with or even delimit from the competition. This new trend, resulting from the emergence of the Industrial Internet of Things (IIoT), offers new automation possibilities by interconnecting all system components with each other. However, as those components, mainly known as Cyber-physical Systems (CPS) are usually systems themselves, the complexity of such a manufacturing system continously rises, which even has to be classified as a complex System of Systems (SoS). Having recognized the issue of the upcoming difficulties when engineering such a system, several German institutes introduced the Reference Architecture Industrie 4.0 (RAMI 4.0), a three-dimensional framework that offers standards or methods for developing system architectures on basis of different views. Nevertheless, at the current point of view, this approach though looks good on paper, but almost none practical applications are existing yet. Thus, this paper focuses on the model-based development of system architectures according to the specifications of RAMI 4.0 in order to evaluate the practical applicability of this reference architecture for usage in actual industrial projects. This is done by utilizing two actual industrial case studies, whose models are created with the help of a specifically designed software tool, the so-called RAMI Toolbox. Based on the outcome of this work, the usage of RAMI 4.0 for application in the industrial area could be sustainably consolidated.**

***Index Terms*—System Architecture, Use Case Evaluation, Industrial Internet of Things (IIoT), Reference Architecture Model Industrie 4.0 (RAMI 4.0), Model-based Systems Engineering (MBSE)**

I. Introduction

In the past decade, manner of manufacturing significantly changed under the occurrence of the fourth industrial revolution. This is primarily encouraged by technological advantages resulting from research in the area of the Industrial Internet of Things (IIoT) or Cyber-physical System (CPS), which more and more find application in actual industrial manufacturing systems [1]. Accompanied by this trend is the emergence of superordinate value creation networks, where components are linked with each other to form an interplay of production units. This new interoperability causes the original unidirectional production process to turn into a dynamic modular manufacturing system, where decisions are made decentrally and each product can be manufactured individually. By doing so, each CPS, although being applied in a larger context, follows its own path and makes the economically most valuable decision on its own [2].

Considering this from a theoretic perspective, it has been pointed out that the resulting interplay of manufacturing units has to be considered as a complex system or even a System of Systems (SoS). Thus, the proposition of the Reference Architecture Model Industrie 4.0 (RAMI 4.0) is especially put on to deal with this complexity. By doing so, the three-dimensional layout of this Service-oriented Architecture (SOA) offers different viewpoints or reference aspects, where a manufacturing system can be created according to the design principles *separation of concerns* or *divide and conquer*. In order to become a common basis for industrial systems engineering, the reference architecture itself has been standardized in the norm DIN SPEC 91345 [3]. However although being described in detail and already standardized, there are almost no applications existing making use of RAMI 4.0 [4]. The main reason is, that this is mainly a theoretical concept missing specifications for actual industrial utilization. Thus, previously a piece of software has been created, which makes use of a Domain Specific Language (DSL) and a respective development process, that allows to create system architectures based on the reference model [5].

Therefore, the first step of making RAMI 4.0 applicable has been set, which is substantiated by the development of architectural models from several fictive case studies or excerpts of complex industrial use cases [6], [7]. As the RAMI Toolbox enables Model Based Systems Engineering (MBSE) for projects in this area, the next step is to actually apply this approach to real industrial and sophisticated applications. Therefore, this paper proposes a review and the lessons learned from modeling two real-world use cases according to the peculiarities of RAMI 4.0, which give insights into the structure and viewpoints of this reference architecture model as well as pointing out possible enhancements. In order to address a wide range of possible scenarios, the first use case deals with product line automation. In this example, a bottleneck where code for setting up the machine according to the respective order is typed in manually should be revised aiming to be created automatically. The architecture of the system and the potential for improvement is thereby modeled according to the layers of RAMI 4.0. The second case study makes use of a more extensive scenario, the development of an Electric Vehicle (EV). In this scenario, the interconnection of Industry 4.0 with the Smart Grid or Automotive domain is explained in more detail, while the individual manufacturing of each EV with the help of modular production islands is typically IIoT related.

To address all aspects of the use case evaluation, this contribution is structured as following: In Section II an overview of RAMI 4.0, Domain Specific Systems Engineering (DSSE) and the Software Platform Embedded Systems (SPES) is given. Next, the approach is stated in Section III. Then, in Section IV, the use case related to machine code generation is explained in more detail, while the application of the EV case study is stated in Section V. Finally, in Section VI the results are summarized, the findings are listed and a conclusion is given.

II. Related Work

A. RAMI 4.0

As mentioned in [8], the goal of RAMI 4.0 is to enable domain-specific systems engineering of industrial systems according to several viewpoints. Therefore, this model is structured in a three-dimensional cube, visualized in Figure 1, which allows to address different aspects of the system to develop. By doing so, it is meant to be a mutual basis for creating a common understanding of industrial manufacturing systems. Such a system can either be seen as a whole and throughout the whole reference architecture, furthermore single aspects can be pointed out in particular. Thus, the "Life Cycle & Value Chain" axis considers the life-cycle of such a system and its participants. Furthermore, the "Hierarchy Levels" axis integrates a standardized way of creating sections for information exchange, derived from the so-called automation pyramid. Each system component can thereby be aligned to one of the resulting panes. In order to enable a top-down discussion, the so-called "Interoperability Layers" have been introduced. Structured into six different viewpoints, aspects like business models, functions, information exchange issues or the asset itself can be assigned to the respective layer.

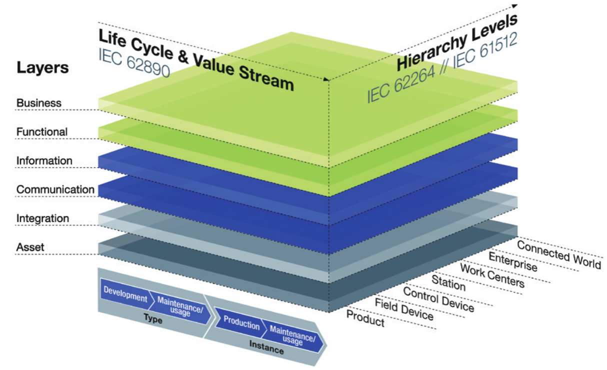

Fig. 1. Reference Architecture Model Industrie 4.0 (RAMI 4.0) [9]

B. Domain-specific Systems Engineering

A practical example enabling DSSE in the industrial area is the RAMI Toolbox [6]. This piece of software provides a framework inheriting domain-specific standards and offers additional functionality supporting the modeling process with Enterprise Architect (EA). Already established by publishing the first version, this tool is constantly developed further according to new technologies or environmental changes. With the goal to enable systems engineering throughout all life-cycle phases, a tool-chain being composed of eight different steps is introduced. Thus, in order to initiate the process of completing the so-called "Integration Toolchain", first the needed information is collected and subsequently modeled to a specific reference architecture. For evaluating this model, several methods like model checking, three-dimensional visualization tools or simulation frameworks need to be enclosed by this tool-chain. At last, the final step is to actually implement the system by applying code generation or methods for Round-trip Engineering (RTE).

C. SPES

The SPES framework belongs to the category of generic architecture frameworks and is primary applied with respect to MBSE. As the proposed framework considers model-based system development in the automation domain, it is suitable, for the modeling of industry-related systems. Basically, SPES provides two essential approaches, following the fundamental principles of *divide and conquer* and *separation of concerns*. Based on core concepts defined in [10], these are *Views and Viewpoints* and *Abstraction Layers*. Further, a two-dimensional engineering space, is formed out of the mentioned approaches, where the horizontal axis is divided into viewpoints, capturing different stakeholder concerns and the vertical axis into various abstraction layers, each representing a specific system level. Additionally, the intended combination of different domain-specific reference architectures to enable cross-domain systems engineering can be observed further in [11].

III. Approach

As already mentioned, the main goal of this contribution is to apply model-based systems engineering in order to develop the architecture of industrial use cases according to RAMI 4.0. This is done for evaluating the reference architecture towards its applicability for actual projects applied from the industry. As those systems are usually complex and contain a lot of aspects to consider as well as being consisted by a large number of CPS, systems engineering also needs to be executed on a superior level. This means, specific DSLs need to be available for addressing all aspects of the system and providing a modeling backbone widely applicable and understandable by a larger audience. Furthermore, new technologies or advances from research and development need to be continuously integrated into the modeling environment. Due to the dynamical alterability and the high rate of change in this domain, the methods of the Agile Design Science Research Methodology (ADSRM) are a suitable concept to apply for refining the RAMI Toolbox to create a comprehensive and sustainable framework. However, the whole process of ADSRM consists of iteration cycles, which is supported by so-called exploratory case studies.

Thus, this paper gives further insights into two separate case studies, which are used to better understand the industrial domain and adapt the modeling environment for sustainable usage. At first, a typical industrial use case is introduced. In this scenario, an already existing manufacturing process for creating drilled metal plates is taken into consideration. According to the vision of "Industry 4.0", some parts of the process need to be automated. More precisely, the code that gives work instructions for the single machines is created by hand for each order, which should be done automatically in the future. The change process is thereby modeled with regard to the specifications of RAMI 4.0 and analyzing the existing system as well as the new system that should be developed. The second case study makes use of a more comprehensive scenario, the cross-domain modeling of an EV with aspects of the industrial, the Smart Grid and the automotive area, as this concept would contribute to the concept of the so-called Smart Cities. However, in order to address the concept RAMI 4.0, the production of such an EV is thereby executed by utilizing a manufacturing system consisting of modular production units, which is modeled according to the three-dimensional reference architecture model by utilizing the SoS requirements.

IV. Industrial Case Study

A. Design & Prerequisites

Originally, this case study is derived from [12], which gives direction on digitalization potential of "Industry 4.0" as well as RAMI 4.0. The proposed work thereby differentiates between several Use Cases, which enhance currently used products or processes. On the one hand, they suggest to digitalize products or services in order to gain customer profit, on the other hand increased efficiency is suggested to be enhanced by digitalizing processes. With regard to the latter mentioned, one of the company partners provided insights into the use case scenario of producing drilled metal plates. In the current manufacturing process, several potentials for improvement can be recognized, which is explained in the remainder of this section. More precisely, manual processes are spread all over the production line, which should be replaced by digital ones. With the help of the RAMI 4.0 Toolbox and according to RAMI 4.0 the model of the system is created, which enables MBSE of the system to develop.

B. Modeling

The first step towards finding enhancements in the current process is to analyze currently used business models, processes or stakeholder concerns. This is done in the Business Layer of RAMI 4.0 by providing several models for either specifying the system context or the interaction with it. Based on this analysis, the business process can be modeled with several process modeling languages, according to whether it is an administrative or a production process. In this case, processes are illustrated with the most commonly used Business Process Modeling Notation (BPMN), *SIPOC*, the so-called "Wertstromanalyse" or *Makigami*. Based on these processes, enhancement potential can be recognized and indicated with so-called "kaizen"-bursts. In this example, three specific problems are fluctuating quality, prone to errors or high effort in regard to creating the work instructions for the machines. Based on these deficiencies, a new business case, the digitalization of the manufacturing process, is created, which is shown in Figure 2. This business case is the base for developing and modeling the System of Interest (SoI) across the RAMI 4.0 layers.

In more detail, after analyzing the system as it currently is, the next step is to elaborate the requirements for the system to develop, which are modeled with a SysML Requirement Diagram. This use case thereby uses typical industrial requirements like increasing the production efficiency by optimizing resource expenses. Those are building the base for the Function Layer, where the Functional Architecture for Systems (FAS) method [13] is applied in order to create functions and assign them to system components. In the aforementioned layer itself, those functions are depicted as black- and white-box models as well as on different abstraction levels. By doing so, input and output can be delineated, which specifies their interrelation. In this scenario, functions like measuring borehole sizes or generating alternative code are developed during this step.

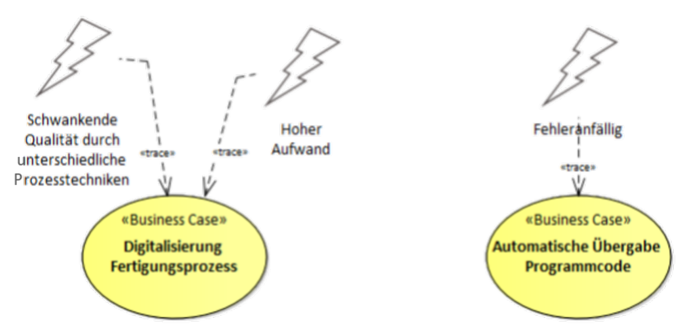

Fig. 2. Business Case Model

However, after assigning the functions to actual system components, the technical architecture can be modeled on the Information as well as on the Communication Layer of RAMI 4.0. Thereby, the transmitted information is explored in detail by showing used data model standards like Extensible Markup Language (XML) or JavaScript Object Notation (JSON) as well as protocols like Near-field Communication (NFC), 4G or Ethernet. Next, further insight into the components is given by creating the Integration Layer. In this example, Human-machine Interface (HMI) interfaces or the Information and Communication Technology (ICT) infrastructure is modeled for the respective system components. At last, the Asset Layer illustrates the virtual representation of a component as it is used in the actual manufacturing system, which would build the base for code generation or RTE.

C. Findings

The just elaborated case study evaluates the usage of RAMI 4.0 for application in a real industrial project. However, as the reference architecture itself is missing formalizations, additional specifications in the RAMI Toolbox like using FAS on the Function Layer or modeling the Asset Administration Shell (AAS) on the Integration Layer significantly helped during the creation of the architectural model. However, some issues have been recognized nevertheless:

- Differences in modeling whole production systems or single components
- Need for integrating OPC UA in order to create more sophisticated models
- Enabling of RTE by implementing AutomationML

V. Case Study SoS

SoS architectures consider systems, which are composed of multiple subsystems and have a strong dependability, as well as interoperability among each other. An example gives the following case study, concering the charging process of an EV, since different domains collide in such an architecture and thus form a complex SoS. The case study deals with different aspects typically occuring during the charging process at a Level II charging station and was first described in [11]. This very first study with respect to the charging behaviour primary focuses on the automotive architecture, especially on the Battery Management System (BMS) of an EV. Referring to the previously mentioned source, several modeling frameworks exist for model-based development of systems, these are e.g. RAMI 4.0, Smart Grid Architecture Model (SGAM) and Automotive Reference Architecture Model (ARAM). Those frameworks consider a system from different viewpoints, each contemplating different stakeholder concerns, but can only be used for modeling in a certain domain. According to the findings in the mentioned paper, these frameworks can be merged and used to model systems belonging to multiple domains i.e. SoS architectures. Therefore, the case study deals with the situation of modeling a system architecture after the principles of SPES and indicates how to enable cross-domain modeling, where it is possible to capture requirements of independent domains and pass certain design parameters in between, which can be further used for research and development. For instance, one possibility is to model the architecture of an EV starting from the very top level, moving to the lower system level, identifying important parameters e.g. the cell capacity of the *High-Voltage Battery* in EVs and passing this information further from the automotive architecture towards the automation/industry domain i.e. to the Battery Manufacturer. The passed on information is from high importance for the manufacturing process e.g. for the production of battery packs needed in EVs. As with the value of the exact cell capacity, one is capable of knowing how the *High-Voltage Battery* in EV performs in certain operational moments and where room for improvement exists.

A. Design

Starting point for modeling is the case study model stated in [11], where the requirements viewpoint of SPES is seen as common interface between the previously mentioned modeling frameworks. In more detail, each of the SPES functional viewpoints is represented by so-called Application Domains (ADs), which can further be used to describe the functional aspect of the considered system. Furthermore, a proper modeling approach must be established, in order to keep consistency and traceability trough the entire model, as otherwise the complexity of the system would be inextricable. Figure 3 visualizes a suitable modeling approach, alligned to the SPES engineering space and shows the proposed modeling direction. Basically, the approach, illustrated in the image, intends as first step to define the SPES requirements viewpoint in the top level and subsequently to perform an allocation to the functional viewpoint, or a decomposition towards the system level, based on the elicitated requirements, respective. The last two viewpoints contemplate the logical concepts and technical solution of the system architecture. However, the first two viewpoints are described in more detail in the following sections.

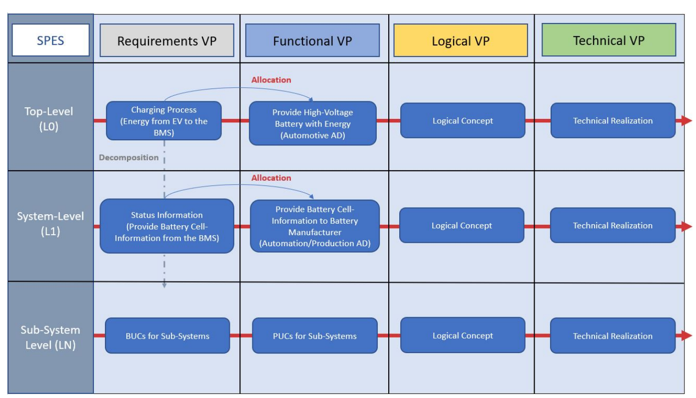

Fig. 3. Modeling approach, alligned to the SPES engineering space [10].

B. Prerequisites

To successfully perform the mentioned modeling tasks in the previous section, it is neccessary to have certain knowledge and expertise concerning DSSE frameworks and generic architecture frameworks, more precisely referring to RAMI 4.0 and SPES. Furthermore, one should be familiar with the usage of General Purpose Languages (GPLs) used for modeling, such as *SysML* and associated modeling techniques. Also important are DSLs, which are used in DSSE frameworks. Additionally required, is a suitable modeling software, which especially supports *SysML*. Regarding to this case study, the modeling of the SoS aspect is done with *IBM Rhapsody* and the manufacturing process utilizing the RAMI Toolbox. Additionally required, is experience in the field of Systems Engineering and tasks linked with this discipline e.g. *Requirements Engineering*.

C. Modeling

The modeling starts with the SPES requirements viewpoint, on the first abstraction layer (top level) and continues referring to the considered case study with the allocation and decomposition principles after [10], to the next viewpoint and abstraction layer (system level). According to the requirements engineering process, outlined in the before mentioned source, the first architecture model is the *context model*, which specifies the SoI and defines the main actors, as well as system context. The SoI focuses on the charging process of an EV, where the main interest relies on the energy transportation from the charging station to the *High-Voltage Battery*. Furthermore, with the definition of the SoI it is possible to specify certain use cases, related to this very interest, which can be further refined into detailed scenarios and requirements. The most meaningful outcome regarding to the case study is the *Business Use Case Diagram*, which illustrates the overall-scenario on the first abstraction layer (top level), which is the charging process of an EV . As previously noted, the definition of the SoI is decisive for the elicitation of requirements and as the system context concerns two domains i.e. the automotive and energy domain, two Business Use Cases (BUCs) must be declared within this very diagram. The goals, specified by these very use cases are significant for the further definition of requirements. Subsequently, Primary Use Cases (PUCs) are specified by those requirements, where each of these represents the functional aspect of the considered system and is assigned to certain ADs. According to the allocation principle of SPES and results stated in [11], PUCs are utilized, to move in a arbitrary direction during modeling i.e. those are further refined in one of the intended domains to model in. Therefore, a proper requirements elicitation in the SPES requirements viewpoint, can capture requirements belonging to different domains, which are finally documentated in the diagram, portrayed in Figure 4. Those requirements, contemplated in the referred image are refined into PUCs, which describe the functional architecture of the SoI and thus create the starting point for the development of the functional viewpoint of SPES. From this point forward, the modeling may develop towards one of the reflected domains i.e. the automation, or energy domain. This can either be achieved, through an allocation to a functional viewpoint of an AD, or through the decomposition of derived requirements from BUCs, to higher layers of abstraction e.g. system level. As the first abstraction layer primary focuses on the charging behaviour of the EV and with that merely on the automotive domain, it is required to enforce a decomposition from the requirements belonging to the top level to a lower system level.

Through this action, the context is switching, as the information exchange between BMS and Battery Manufacturer, is taken into account on this very layer and the new created requirements belong to the automation domain. To explain the decomposition in more detail, the requirement *REQ-001*, shown in the previously mentioned image, is decomposed into a new, more precise requirement, considering the information about cell capacity of the *High-Voltage Battery*. Based on those requirements the SPES functional viewpoint can be modeled by applying the FAS method after [13]. Further, *functional elements* and *functional groups* are defined, where the latter regards to activities in the automation AD and traces the corresponding functional elements i.e. elements related to this domain. Hence, the functional element relates in this case to the BMS of the EV, as this particular component contains important information about the cell capacity of the *High-Voltage Battery*. Another essential part of the functional viewpoint, is the refinement of the more accurate requirements into PUCs, as already mentioned. This process takes place on this very viewpoint and is stated as *Provide Battery Cell Information to Battery Manufacturer*. Thus, it addresses the SoI directly, as with its definition the information about the cell capacity should be provided to the Battery Manufacturer. Accordingly the behaviour of the PUC and with that of the functional elements, is explained by activity diagrams. Furthermore, this kind of diagrams contributes to a better understanding of the functional elements and helps to detect peculiarties concerning the internal structure.

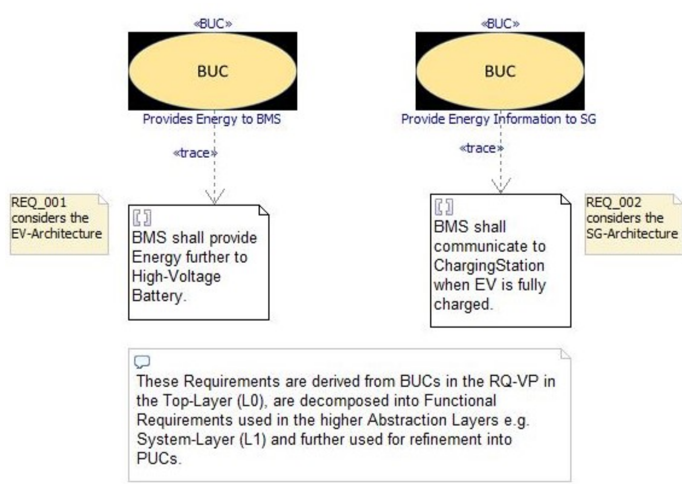

Fig. 4. Requirements Diagram in the SPES requirements viewpoint.

The remaining two viewpoints concern the logical and technical architecture of the SoI, which is the BMS in the system layer. Functional elements are allocated to logical elements, which are required for developing the logical architecture of the system layer. Finally, the technical realization is done in the last viewpoint proposed by SPES, the technical viewpoint, where logical elements are realized by technical components.

D. Findings

With the explained case study it is indicated that a modeling across domains is feasibile, by following the stated modeling theories and approach. This makes it possible to create a uniform and consistent system architecture, especially regarding to the information exchange between system contexts and consequently across system boundaries. In particular the idea of separating each considered domain into ADs and with that the definition of PUCs, enables to address important functional aspects of the considered SoI. As the behaviour of those PUCs is described by activity diagrams it may be combined with architecture based *co-simulation* frameworks like *Mosaik* to simulate certain activities within the created architecture model. Furthermore, this intention can be of outermost meaning, as utilizing *co-simulation* allows to create valuable predictions of specific system behaviour e.g. of the *High-Voltage Battery* in EVs. Following this principle, an EV can be a suitable use case for further research in this area, as multiple domains are affected. Thus, more general findings and particular results in this specific field need to be elaborated in future projects and more sophisticated case studies utilizing dynamic approaches.

VI. Conclusion & Future Work

Nowadays systems engineering is a difficult task entailing a lot of challenges, which leads back to the fact that systems increasingly become more and more complex. In particular, the industrial area especially profits from advances promoting automated manufacturing in the context of "Industry 4.0" and its expressions, the IIoT or CPS. However, accompanied with this rising complexity, new methodologies or tools for creating virtual representations of these systems appear. For example, as MBSE has proven to be a key enabler when it comes to industrial system development, RAMI 4.0 and its associated tool, the so-called RAMI Toolbox, are two approaches resulting from this trend. Nevertheless, as the current point of view, the reference architecture and its modeling framework are suffering from lack of specifications, which hinders extensive systems engineering of production lines. Thus, in this work the development of two separate case studies making use of the mentioned concepts is explained in detail, which helps evaluating the already existing work and helps refining it for future applications. By doing so, the approach for developing the RAMI Toolbox and for creating the virtual architectures is explained in detail in Section III. However, as all aspects of future industrial systems should be spanned, two varying case studies are used for the evaluation process. At first, a typical IIoT based use case is illustrated in Section IV, while a SoS example is further explained in Section V.

Based on the outcome of this work, new or follow-up projects can profit from this evaluation. For example, the RAMI Toolbox itself can be refined to enable systems engineering in a more complex context. This means, the framework should compensate the missing formalization of the RAMI 4.0 specification and provide more automation possibilities supporting non-practiced users in modeling. This means, to mention one example, an approach enabling systems engineering on multiple granularity levels has to be implemented. Furthermore, the overall tool-chain of the RAMI Toolbox can be refined on the basis of this work, which could include code generation with AutomationML, RTE with OPC UA or enabling co-simulation of the manufacturing system. Regarding the SoS context, the interface between RAMI 4.0 and other domains needs to be defined more precisely. In particular, the concept of domain-specific systems engineering needs to be extended towards domain-specific SoS engineering.

Acknowledgment

The support for valuable contributions of LieberLieber Software GmbH and successfactory consulting group is gratefully acknowledged. The financial support by the Federal State of Salzburg is also gratefully acknowledged.

References

[1] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in 2014 IEEE international conference on automation, quality and testing, robotics. IEEE, 2014, pp. 1–4.

[2] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems," Manufacturing letters, vol. 3, 2015, pp. 18–23.

[3] Deutsches Institut für Normung, "DIN SPEC 91345: Rami 4.0," 2016.

[4] R. Harrison, D. Vera, and B. Ahmad, "Engineering methods and tools for cyber–physical automation systems," Proceedings of the IEEE, vol. 104, no. 5, 2016, pp. 973–985.

[5] C. Binder, C. Neureiter, and G. Lastro, "Towards a Model-Driven Architecture Process for Developing Industry 4.0 Applications," International Journal of Modeling and Optimization, vol. 9, no. 1, 2019, pp. 1–6.

[6] C. Binder, C. Neureiter, G. Lastro, M. Uslar, and P. Lieber, "Towards a standards-based domain specific language for industry 4.0 architectures," in Complex Systems Design & Management, E. Bonjour, D. Krob, L. Palladino, and F. Stephan, Eds. Springer International Publishing, 2019, pp. 44–55.

[7] C. Binder, D. Draxler, C. Neureiter, and G. Lastro, "Towards a model-centric approach for developing functional architectures in industry 4.0 systems," 2019.

[8] M. Hankel and B. Rexroth, "The Reference Architectural Model Industrie 4.0 (RAMI 4.0)," ZVEI, 2015.

[9] Bitkom, VDMA, ZVEI, "Umsetzungsstrategie Industrie 4.0, Ergebnisbericht der Plattform Industrie 4.0," ZVEI, 2015.

[10] K. Pohl, H. Hönninger, R. Achatz, and M. Broy, Model-based engineering of embedded systems: The SPES 2020 methodology. Springer Science & Business Media, 2012.

[11] B. Brankovic, C. Binder, D. Draxler, C. Neureiter, and G. Lastro, "Towards a cross-domain modeling approach in system-of-systems architectures," in Complex Systems Design & Management, G. A. Boy, A. Guegan, D. Krob, and V. Vion, Eds. Cham: Springer International Publishing, 2020, pp. 164–175.

[12] C. Mosch and F. Prumbohm, "Leitfaden industrie 4.0 trifft lean–wertschöpfung ganzheitlich steigern," 2018.

[13] J. G. Lamm and T. Weilkiens, "Funktionale Architekturen in SysML," in Tag des Systems Engineering, M.Maurer and S.-O. Schulze, Eds. München, Germany: Carl Hanser Verlag, Nov. 2010, pp. 109–118, English translation by J. Lamm.