# Towards a generic Process-Model definition in Cross-Domain architectures

Boris Brankovic, Christoph Binder, Christian Neureiter and Goran Lastro

**Abstract** The ongoing integration of Cyber-physical Systems (CPS) into contemporary systems cause new challenges but also great potential in multiple domains. By the term of the so-called Smart City, affected domains thus are Industry 4.0, the Smart Grid or even the automotive area, which could benefit from decentralized decision making in automation systems. That is why Model Based Systems Engineering (MBSE) seems to be a suitable way for dealing with the rising complexity, although cross-domain modeling appears to be a major challenge at the current point of view. In addition, the Software Platform Embedded Systems (SPES) modeling framework and its follow-up projects offer a suitable groundwork for enabling systems engineering across multiple domains. However, although proposing a ready-to-use architecture framework, it is still unclear how to develop such a System of Systems (SoS) in a standardized uniform way. Thus, this paper introduces the definition of a process-model based on the ISO 15288, which concerns the development of complex SoS architectures spanning across several domains and aligned to the specifications of SPES.

## 1 Introduction

Systems engineering has recently transformed into a difficult task, caused by continuously integrating new advances in the area of Internet of Things (IoT) and Cyber-physical System (CPS), resulting them to become more and more complex. This trend is underpinned by analyzing recently proposed applications regarding Model Based Systems Engineering (MBSE) in various domains [15, 5, 2]. Although the vast majority of them considers MBSE as key enabler when it comes to deal with

Boris Brankovic, Christoph Binder, Christian Neureiter, Goran Lastro
Josef Ressel Center for Dependable System-of-Systems Engineering Salzburg University of Applied Sciences, Urstein Süd 1, A–5412 Puch/Salzburg, Austria, e-mail: {boris.brankovic, christoph.binder, christian.neureiter, goran.lastro}@fh-salzburg.ac.at

this complexity, most of the proposed methods claim not to be a mature ready-to-use methodology rather than a first step into the right direction. Theoretically speaking, by applying the classification scheme proposed in [8], the integration of more and more decentralized units into current and future systems result them to be considered as complex systems, while original ones are classified as complicated systems. Moreover, as CPSs being systems themselves, even the term System of Systems (SoS) has to be introduced.

However, accompanied by the recent emergence of the so-called Smart City, systems from several domains are slowly merging together. For example, considering the lifecycle of an Electric Vehicle (EV), this trend can be explained in detail. The infrastructure management and the placement of loading stations is thereby a particular issue of adapting future cities [12]. The development of the EV affects the Industrial Internet of Things (IIoT) due to the individuality of each vehicle and the resulting modular production lines [20]. Furthermore, the energy consumption and the interplay with other prosumers or Smart Home applications is dedicated to the Smart Grid [6], while the arrangement of the car parts and intelligent devices are addressed by automotive systems [14]. Summarizing, this means that future systems engineering not only needs to consider the complexity of a system within a certain area but rather has to enable the development of complex cross-domain SoS.

Having recognized this issue, the authors of [3] propose an approach for developing system architecture spanned over multiple domains by utilizing the Software Platform Embedded Systems (SPES). Although such a kind of methodology is still being in its infancy, the introduced framework appears to be promising concept for dealing with this issue. By utilizing the Requirements Viewpoint as common reference point for developing systems in different domains and with various architecture frameworks, cross-domain modeling is enabled. However, as neither SPES nor MBSE provide a specific development process for such a system, it is difficult for users to develop the architecture of such a complex system spanning across domains along to the columns and rows of SPES.

Therefore, this paper introduces a process-model proposing a model-based development process for the development of complex cross-domain SoS according to the peculiarities of SPES. By doing so, the approach is aligned to the lifecycle of systems engineering as applied by the V-model of the ISO 15288 and the concepts of Domain Specific Systems Engineering (DSSE). In order to evaluate the resulting process-model and its artifacts towards applicability, a real-world case study of an EV is utilized. The outcome of this work can thereby be considered as a major step towards enabling cross-domain SoS engineering.

To address the aforementioned issues, this contribution is structured as following: In Section 2 an overview of systems engineering in general, the addressed architecture frameworks and SPES is given. Subsequently, the approach is stated in Section 3. Then, in Section 4, the development of the process-model is further explained, whose application is illustrated in Section 5. Finally, in Section 6 the results are summarized and a conclusion is given.

## 2 Related Work

### *2.1 Systems Engineering Concepts*

Systems Engineering (SE) is an interdisciplinary approach, which is primary used for modeling of entire system architectures, by utilizing proven concepts in order to create completely new applications, satisifying the needs of stakeholders [7]. Due to the growing complexity in system architectures, as well as accumulating requirements, new methods towards a more consistent approach were demanded. Therefore, MBSE was introduced, aiming at the definition of one central element i.e. the system model, capturing all necessary information and keeping consistency through all stages of development [21, 22]. Another proven-key concept is DSSE, first stated in [15], which focuses on the implementation of a framework for MBSE and incorporates basically three main elements viz., *Domain-specific modeling framework, Process-Model* and *Tool-Chain integration* [16]. The mentioned approach finds its origin in the context of the energy domain i.e. in Smart Grid related research and additional domains. As one main endeavour is to enable dependency by design and to cover all engineering phases, a Process-Model aligned with the ISO 15288 standard has been invented, including the main phases *System Analysis, System Architecture* and *Design and Development*.

### *2.2 Domain Specific Architecture Frameworks*

Reference architectures serve as base for the development of complete system architectures and support the creation of models aligned to a specific system context. Examples herefore are state-of-the-art frameworks considering the application domains of *Automotive Engineering, Smart Grid Architecture* and *Inudstry 4.0*. Therefore, as baseline for those very domains, serve the Automotive Reference Architecture Model (ARAM) , Smart Grid Architecture Model (SGAM) and Reference Architecture Model Industrie 4.0 (RAMI 4.0). Each of those provides a three-dimensional model structure to enable a common understanding, as well as common basis of systems belonging to the corresponding domain. Furthermore, the mentioned frameworks are divided into different layers, or viewpoints, respective, which are used to capture specific stakeholder concerns. According to the international standard ISO 42010 [9] specific concepts were established for the description of system architectures and with that certain *Model-Kinds* i.e. diagrams for modeling, to frame those particular concerns. Hence, the authors in [15, 2, 5] define for each layer, or viewpoint, a set of *Model-Kinds*, based on the Architecture Description Language (ADL) *SysML*, used to explain the applicable architectural model, referring to ISO 42010. As the considered reference architectures primary concentrate on a certain system context, the proposed Model-Kinds are ment to be rather *domain-specific* than generic.

### *2.3 Generic Architecture Frameworks*

In contrast to domain specific frameworks, generic architecture frameworks are not tailored to a certain domain. Especially, if SoS architectures are considered, domain specific frameworks do not fulfill the requirements throughout the modeling of systems, as more than one domain must be taken into account. In particular the SPES modeling framework, has proven as leading approach for the engineering of collaborative embedded systems, referring to ongoing projects like the CrESt project [1] [19]. In more detail, SPES is used for model-based development of systems and thus utilized in the context of MBSE. As it supports the modeling of diverse domains inter alia the energy, automotive and industry domain, SPES is suitable for system development in a SoS environment. Substantially, two fundamental approaches were specified, which are *Views and Viewpoints* and *Abstraction Layers*. Those are based on core concepts defined in [18]. Furthermore, a two-dimensional engineering space is formed out of these methods, where the horziontal axis is separated into four viewpoints i.e. *Requirements, Functional, Logical* and *Technical*, capturing different stakeholder concerns, respectively. The vertical axis is divided into various abstraction layers, each representing a certain system level. Additionally, *Requirements Artifacts* were defined by the authors in the previously stated source, each providing specific models, to pertain the current viewpoint.

### *2.4 ISO 15288 - System Lifecycle Processes*

Following the lifecycle of systems, an appropriate standard is needed, providing a set of processes and with that a suitable framework to describe all phases during development. The international standard ISO 15288 meets aforementioned necessaries, as it delivers 25 processes, covering 403 activities, which result in 123 outcomes, contemplating a system from an engineering viewpoint [15]. This commonly used standard, emerged from the findings of Lake stated in [11]. It offers an optimized development process, as it adapts the defined phases by the mentioned author and integrates a V-model, to reach best possible stakeholder satisfaction. Processes, defined by the standard can be applied individually to the areas, which are decisive for the development and basically applied at an arbitrary level of hierachy, concerning the structure of a system. Figure 1, illustrates the structure of the particular stages and shows the available processes, which are separated into *Agreement, Enterprise, Project* and *Technical* processes. Through the introduction of the V-model, a framework was created, enabling the evaluation of each part of the system, which emerged from the execution of the Technical-processes [21, 10].

[1] https://crest.in.tum.de/

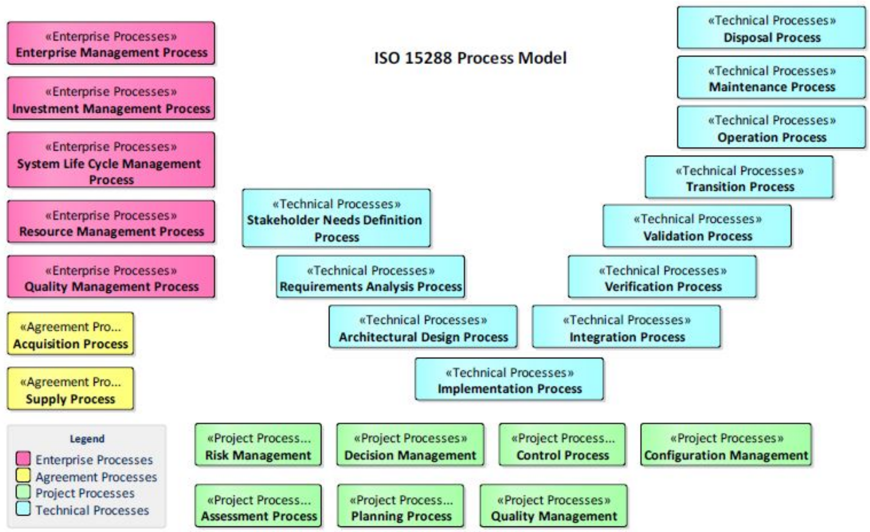


Fig. 1: System lifecycle structure, in accordance with ISO 15288.

## 3 Approach

### *3.1 Research Methodology*

The research methodology, with respect to this work, is based on the Agile Design Science Research Methodology (ADSRM) introduced in [4]. It in particular allows a continous improvement of the problem- and solution-space, as the chosen method combines agile concepts, used for the development of Information Systems (ISs) and rigor, needed by the Design Science Research (DSR) process, defined by Peffers et al. [17]. Furthermore, the agile concepts, give a better understanding of specific problems during the research and requirements process, where the latter is an essential part of ADSRM. Therefore, the methodology demands as first input a suitable case study, crucial for the requirements engineering process and to develop the main artifacts, which are in turn based on elicitated requirements. Those artifacts, are the result of each process iteration, used to model the case study and if necessary to remodel the architecture, with every new cycle. Accordingly, previous research in [3], deals with a very first example of a case study, focusing on the charging process of an EV, as well as with the development of the first main artifact. As a consequence, the findings have shown, that an amalgamation of the mentioned frameworks ARAM, SGAM and RAMI 4.0, is feasible. Thus, the definition of a common ground between those frameworks, with SPES as interface among, enables cross-domain modeling in a SoS environment. However, additional investigation is required to evaluate modeling peculiarities, as the first artifact of the ADSRM process, primary concentrates on the mapping between domain specific and generic frameworks. Moreover, with reference to the authors in [15, 2, 5], a complete Process-Model for the definition of the viewpoints in ARAM and layers in SGAM and RAMI 4.0 exists, defining each a set of Model-Kinds. Hence, the international standards ISO 42010 and ISO 15288 should be aligned with the viewpoints of SPES, in order to evaluate existing Model-Kinds, belonging to the respective modeling frameworks and to determine potential interfaces among them. Therefore, the requirement for the development of the second artifact regarding to ADSRM is stated as follows.

- *An appropriate Process-Model should be defined, for the purpose of evaluating existing Model-Kinds in the respective frameworks and to establish potential interfaces among them.*

Finally, ellaborated findings and expieriences serve as input for the second stage of ADSRM i.e. the next cycle in the process, which is after Conboy et al. the evaluation by domain-experts.

### *3.2 Case-Study*

According to the previously described section, an appropriate case study is one request by ADSRM. Therefore, an example is chosen, which encompasses multiple domains and thus constitutes a SoS environment. The study focuses primary on the charging process, regarding to EVs and charging stations, first described in [3] and briefly explained in the following. Basically three different types of charging stations exist *Level I, Level II* and *Level III*-Stations. Those, are categorized by the Society of Automotive Engineers (SAE), with the development of the standard SAE J1772, introduced in [1]. In a first draft of the case study, only Level II charging stations are of interest, together with its components which are typically a *On-Board Charger, Inverter, Battery Management System* and *High-Voltage Battery*. Subsequently, referring to [5], it is possible to model entire EV architectures from an automotive viewpoint. Further, a successfully conducted case study is indicated by the authors from the previously mentioned source, concerning a breaking system of EVs, which has been extended in the context of EVs charging at a charging station. This extension includes an advanced decomposition into supplementary subsystems and moves the main focus towards the Battery Management System (BMS) of an EV. Moreover, as the energy-consumption of EVs plays an essential role within the Smart Grid (SG), the development and production of *High-Voltage Batteries* becomes crucial, in particular because it affects the powergrid during runtime. Hence, the BMS is a significant component, since it serves as an interface between *On-Board Charger* and *High-Voltage Battery*. Thus, it is possible to determine how the battery behaves in certain operational moments. This ability is specifically important concerning the information exchange between system contexts and as well across system boundaries.

# 4 Implementation

## *4.1 Process-Model Evaluation*

According to [18] the SPES framework defines so-called *Requirements Artifacts*, which are identified as architectural models. Basically, these models define specific modeling methods related to corresponding viewpoints. Furthermore, as outlined in [9], Model-Kinds were introduced to express considered models and to frame stakeholder concerns. Consequently, a suitable process-model, aligned with the international standards ISO 15288 and ISO 42010 has been defined, in order to identify and evaluate Model-Kinds required by SPES. Figure 2 illustrates the evaluation process, developed in conformity with pre-defined process-models for the modeling frameworks ARAM, SGAM and RAMI 4.0, as well under consideration of the viewpoint mapping established in [3].

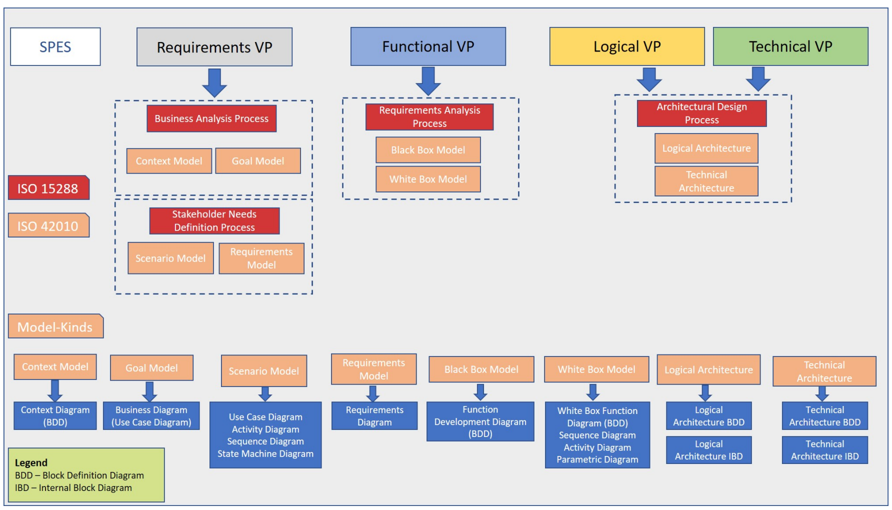


Fig. 2: Process-Model for the evaluation of Model-Kinds, with respect to SPES.

Relying on the mentioned DSSE approach, explained in section 2, three main phases are necessary to cover all engineering stages in development. As result, the process-model, shown in the image, makes use of the standard ISO 15288, to determine an appropriate system lifecycle, with suitable technical-processes. These include, *Business Analysis, Stakeholder Needs Definition, Requirements Analysis* and *Architectural Design* processes. Further, a classification of the SPES *Requirements Artifacts* i.e. architectural models, is done according to the standard ISO 42010.

### 4.2 Classification

In the following the result of the classification, in relation to the technical processes and associated Model-Kinds is listed.

***Business Analysis Process -*** This process contains the *Context* and the *Goal* model, which are present in the SPES Requirements Viewpoint and are as well part of the Business Analysis Process, defined by the ISO 15288. According to the evaluation process, depicted in Figure 2, the *Context Diagram* is defined by the Context model, inherited from the SysML *Block Definition Diagram* and the Goals model specifies the *Business Diagram*, which is based on the SysML *Use Case Diagram*.

***Stakeholder Needs Definition Process -*** The *Scenario* and *Requirements* model are established within this process. Relating to [10] diverse diagrams may be used for the description of required needs, as the proposed process concerns stakeholders, as well as the requirements definition. Therefore, multiple SysML based diagrams appertain to these Model-Kinds, as for instance *Use Case Diagrams, Activity Diagrams, Sequence Diagrams* and *State Machine Diagrams*. This process can be used along with the previously described on, to characterize the Requirements Viewpoint of SPES.

***Requirements Analysis Process -*** To explain the SPES Functional Viewpoint, a process is needed, regarding to the transformation of stakeholder and user-oriented views, into a more precise technical view, which addresses a particular solution and hence, the operational needs of a user [10]. For this reason, the Requirements Analysis process is required, as it specifies the Requirements Artifacts of SPES. These are, the *Black Box* and *White Box* Model taking into account the SysML Model-Kinds *Block Definition Diagram, Sequence Diagram, Activity Diagram* and *Parametric Diagram*.

***Architectural Design Process -*** With this process, the definition of system architectures is addressed and with that the Logical and Technical Viewpoint of SPES. It includes, as defined in the image, the SPES Requirements Artifacts, *Logical Architecture* and *Technical Architecture*. Subsequently, the proposed Model-Kinds are *Block Definition Diagram* and *Internal Block Definition Diagram*.

## 5 Application

In the following section the modeled *Requirements Artifacts* i.e. the most important architectural models with respect to SPES and cross-domain modeling, together with the corresponding set of Model-Kinds are explained in more detail, starting from the *Business Analysis Process*, as indicated above.Hence, the first considered architectural model is the *Context Model*, which defines the system context and specifies the System of Interest (SoI). The SoI, concentrates with respect to the case study defined in section 3, on the charging process of an EV, where the main interest relies in the energy transfer from the charging station to the *High-Voltage Battery*. Thus, the modeling focuses primary on necessary processes required by SPES i.e. to determine which Model-Kinds, regarding to the modeling aspect, apt to be utilized within the SPES framework. However, to provide a meaningful example, contemplating a SoS environment, the SoI is limited to the BMS of an EV and only the first layer of abstraction is examined. Considering the entire charging process with associated systems, as well as subsystems, exceeds the scope of this paper. Finally, the case study model itself is implemented utilizing the modeling software *IBM Rhapsody*.

***Case Study Model -*** Based on the classification scheme, defined in section 3, four processes in total are identified, where the first two represent the SPES Requirements Viewpoint. According to [18], the *Requirements Engineering Process* is used as support for the elicitation and documentation of requirements for a System under Development (SuD). Furthermore, architectural models serve as systematical contemplation of the final requirements and the context i.e. the SoI. Accordingly, the first identified technical process, with respect to the classification scheme, provides two models, the *Context* and *Goal* model. As the SoI is addressed by the context model, it is illustrated by the proposed *Context Diagram*, which is inherited by the *Block Definition Diagram* and is part of the SysML-Profile, used for modeling with IBM Rhapsody. The next model i.e. the *Goal* model aims to explain the overall-scenario, considered on the first abstraction-layer (top-layer), which is the charging-process of an EV. Suitable for this intention, is the *Business Use Case Diagram*, depicted in Figure 3. As two domains are affected by the context on the top-layer i.e. primary the automotive and secondary the energy domain, two Business Use Cases (BUCs) are defined, both invoking the High Level Use Case (HLUC). The image shows also that each BUC, specifies a certain goal, which has to be fulfilled by possible scenarios, defined by the *Scenario* model, in the second technical process, the *Stakeholder Needs Definition Process*. Further the stated goals of the BUCs are decisive for the definition of the requirements. Those, are belonging to the *Requirements* model, classified by the second process and create the starting point for the development of the Functional Viewpoint of SPES, as these very requirements are further refined into Primary Use Cases (PUCs), describing the functional architecture of the considered SoI. Subsequently, requirements can be captured referring to the second technical process, in appropriate diagrams, as for instance the *Use Case Diagram*, or the more accurate *Requirements Diagram* and resulting scenarios may be constructed in the *Scenario Diagram*.

From this point, the modeling may evolve after [3] towards the direction, of one of the considered domains in the SoS environment. As proposed by the authors, this is possible through the allocation of requirements from the previous viewpoint to functional elements in the SPES Functional Viewpoint, that are an abstract representation of the considered system. Therefore, the third technical process in the classification scheme, provides the *Requirements Analysis Process*, which explains the SPES Functional Viewpoint. This viewpoint can be developed, by applying the Functional Architecture for Systems (FAS) method defined in [13] and includes as result the mentioned functional elements and subsequently PUCs, refined from the allocated requirements. Referring, to the third process, the *Black-Box*, as well as *White-Box* model suits this viewpoint. Hence, *Block Definition Diagrams* contemplate the internal structure of the functional element, which refers in the top-level to the *OnBoard-Charger*, crucial for charging the *High-Voltage Battery*. Moreover, essential for this very viewpoint is the description of the behaviour relating to PUCs, which can be modeled in *Activity Diagrams*. Finally, the *Architectural Design Process*, which is the last process in the classification scheme, defines Model-Kinds, applicable to model the logical and technical viewpoint of SPES. Consequently, the development of the logical architecture, as well as technical architecture is feasible with *Block Definition Diagrams* and *Internal Block Definition Diagrams*.

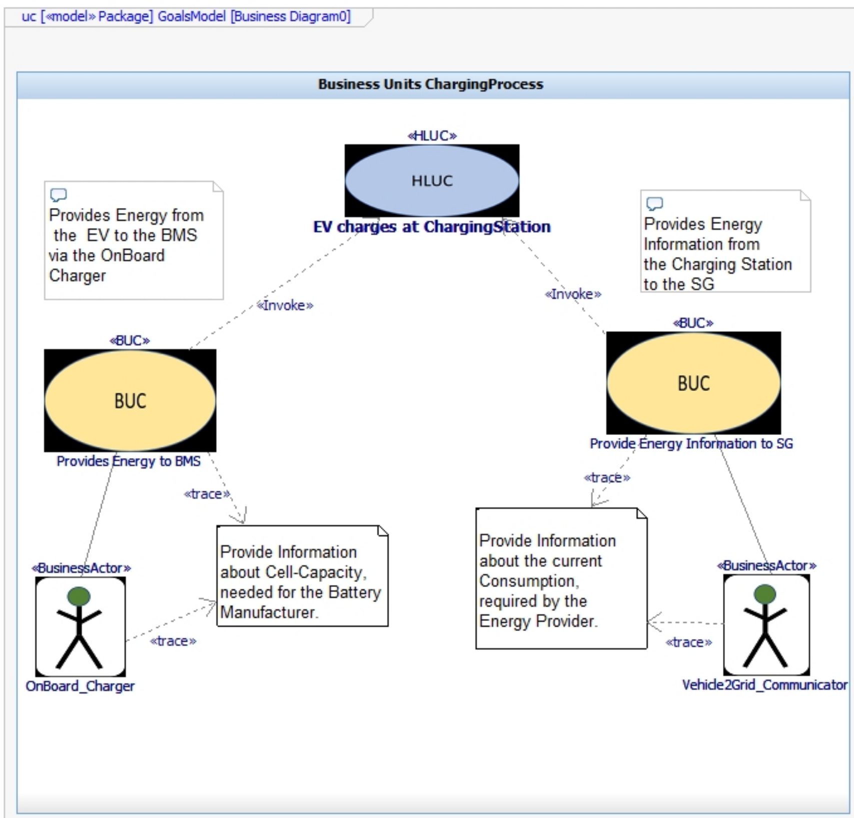


Fig. 3: Business Use Case Diagram, defined by the Business Analysis Process.

***Findings -*** Applying the proposed Model-Kinds from the classification scheme in section 4 to model the case study and thus the SoI, in particular points out that, with the evaluated processes a modeling in a SoS environment after the principles of SPES is feasible. Therefore, the development of a consistent and meaningful example, as stated in the previous section, indicates that all diagrams may be used for the main purpose of modeling the charging process of an EV. However, to provide also consistency through the considered domains e.g. if a context switch is enforced and a cross-domain environment must be taken into account, more precise processes and with that Model-Kinds need to be established.

## 6 Conclusion and Future Work

The evaluated set of processes defines a first approach towards a generic process-model for the modeling of complex SoS architectures. As the goal is to provide Model-Kinds, appropriate for modeling in the context of *Smart Cities*, the modeling framework SPES, is used to determine useful diagrams, suitable for the modeling in the considered domains *automotive, energy* and *industry*. Therefore, making use of the established standard ISO 15288 in combination with SPES, results in the process-model stated in section 4. Further, the outcome is utilized to model a real-world case study in section 5, where the applicability of the specified Model-Kinds is assessed with respect to the viewpoints of SPES. This intention has shown, that a modeling in a SoS environment is feasible and thus the provided diagrams are sufficient. However, with the introduction of SGAM, ARAM and RAMI 4.0, certain modeling characteristics tailored to the corresponding domains are specifically addressed, which require additional investigation relating to available diagrams within those frameworks. Consequently, a Model-Kind mapping must be conducted in order to find analogies between the mentioned modeling frameworks and SPES. Thus, this work can contribute to further projects including a standardized process-model for the development of cross-domain architectures and in particular concerning the development of generic modeling languages.

***Acknowledgements -*** The support for valuable contribution of Robert Bosch GmbH is gratefully acknowledged. The financial support by the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development and the Christian Doppler Research Association as well as the Federal State of Salzburg is also gratefully acknowledged.

## References

1. Angelov, G., Andreev, M., Hinov, N.: Modelling of electric vehicle charging station for dc fast charging. In: 2018 41st International Spring Seminar on Electronics Technology (ISSE),

pp. 1–5 (2018)
2. Binder, C., Neureiter, C., Lastro, G.: Towards a Model-Driven Architecture Process for Developing Industry 4.0 Applications. International Journal of Modeling and Optimization **9**(1), 1–6 (2019)
3. Brankovic, B., Binder, C., Draxler, D., Neureiter, C., Lastro, G.: Towards a cross-domain modeling approach in system-of-systems architectures. In: Complex Systems Design & Management, pp. 164–175. Springer International Publishing, Cham (2020)
4. Conboy, K., Gleasure, R., Cullina, E.: Agile Design Science Research. In: B. Donnellan, M. Helfert, J. Kenneally, D. VanderMeer, M. Rothenberger, R. Winter (eds.) New Horizons in Design Science: Broadening the Research Agenda, pp. 168–180. Springer International Publishing, Cham (2015)
5. Draxler, D., Neureiter, C., Lastro, G., Schwartzkopff, T., Boumans, M.: A domain specific systems engineering framework for modelling electric vehicle architectures. In: 2019 IEEE Transportation Electrification Conference and EXPO Asia-Pacific. Jeju, Korea (2019)
6. Fiori, C., Ahn, K., Rakha, H.A.: Power-based electric vehicle energy consumption model: Model development and validation. Applied Energy **168**, 257–268 (2016)
7. Guillerm, R., Demmou, H., Sadou, N.: Engineering dependability requirements for complex systems - A new information model definition. In: 2010 IEEE International Systems Conference, pp. 149–152 (2010)
8. Haberfellner, R., de Weck,, O., Fricke, E., Vössner, S.: Systems Engineering - Grundlagen und Anwendung, 13 edn. Orell Füssli (2015)
9. International Organization for Standardization: ISO/IEC/IEEE 42010: Systems and software engineering – architecture description (2011)
10. ISO/IEC/IEEE: ISO/IEC/IEEE 15288 International Standard - Systems and software engineering – System life cycle processes (2015)
11. Lake, J.G.: Thoughts about life cycle phases. INCOSE International Symposium **7**(1), 597–603 (1997)
12. Lam, A.Y., Leung, Y.W., Chu, X.: Electric vehicle charging station placement: Formulation, complexity, and solutions. IEEE Transactions on Smart Grid **5**(6), 2846–2856 (2014)
13. Lamm, J.G., Weilkiens, T.: Funktionale Architekturen in SysML. In: M.Maurer, S.O. Schulze (eds.) Tag des Systems Engineering, pp. 109–118. Carl Hanser Verlag, München, Germany (2010), English translation by J. Lamm
14. Lv, C., Liu, Y., Hu, X., Guo, H., Cao, D., Wang, F.Y.: Simultaneous observation of hybrid states for cyber-physical systems: A case study of electric vehicle powertrain. IEEE transactions on cybernetics **48**(8), 2357–2367 (2017)
15. Neureiter, C.: A Domain-Specific, Model Driven Engineering Approach for Systems Engineering in the Smart Grid. MBSE4U - Tim Weilkiens, Hamburg, Germany (2017)
16. Neureiter, C., Binder, C., Brankovic, B., Lastro, G.: Extending the concept of domain specific systems engineering to system-of-systems. In: 2020 IEEE 15th International Conference on System of Systems Engineering. IEEE (2020), to appear.
17. Peffers, K., Tuunanen, T., Rothenberger, M., Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems **24**(3), 45–77 (2007)
18. Pohl, K., Hönninger, H., Achatz, R., Broy, M.: Model-based engineering of embedded systems: The SPES 2020 methodology. Springer Science & Business Media (2012)
19. Rumpe, B., Schaefer, I., Schlingloff, B.H., Vogelsang, A.: Special issue on engineering collaborative embedded systems. SICS Software-Intensive Cyber-Physical Systems **34**(4), 173–175 (2019)
20. Waltl, H., Wildemann, H.: Modularisierung der Produktion in der Automobilindustrie. TCW, Transfer-Zentrum (2014)
21. Weilkiens, T.: Systems Engineering mit SysML/UML, 3 edn. dpunkt, Heidelberg (2014)
22. Weilkiens, T., Lamm, J.G., Roth, S., Walker, M.: Model-Based System Architecture. John Wiley & Sons (2016)