NordDesign 2022
August 16-19, 2022
Kgs. Lyngby, Denmark

Access Logics for Situation-Appropriate Selection and Introduction of Methods in Engineering Design

Theresa Ammersdörfer[1], Johannes Müller[2], Jonas Heimicke[2], Helena Reichardt[1], Simon Rapp[2], David Inkermann[1], Albert Albers[2]

[1]IMW – Institute of Mechanical Engineering, Technische Universität Clausthal, Germany
{Ammersdoerfer, Reichardt, Inkermann}@imw.tu-clausthal.de
[2]IPEK – Institute of Product Engineering, Karlsruhe Institute of Technology (KIT), Germany
{Johannes.Mueller2, Jonas.Heimicke, Simon.Rapp, Albert.Albers}@kit.edu

Abstract

Development methods from Systems Engineering (SE) or agile product development support companies in dealing with current challenges like increased interaction of subsystems and multidisciplinary. However, for many companies, these methods and approaches are still unfamiliar. Developers are often unaware of a suitable development method. Therefore, guides are used to find the appropriate methods and assist their application to the company's specific problems. The main shortcoming of existing guides is that they rely heavily on fixed activities and process-maturity levels and give little attention to the specific situation of the application context. Consequently, methods often miss the actual demand and do not trigger the hoped-for process improvements. The objective of this research is to improve the access to development methods by assisting in clarifying the problem situation and interactions to process activities be addressed when selection methods. The essential goal is to increase the contextual awareness required to identify and implement suitable development methods. To characterize different engineering activities, goals, variables, and key performance indicators were identified based on a literature review. These elements were used in two different projects to develop access logics which are used to select the methods. The results of this short-term research are an online guide and a simulation game to develop an awareness of development methods and enable developers to identify methods according to their specific context information. Within the online guide, the link between goals and variables, a series of predefined questions and the characteristics of the goals, and a set of described methods are used to guide the user to his individual method selection with the help of interactive question trees. Within the simulation game, a set of variables is used to characterize each method, and interactions with process activities, tools, and roles are given. Based on a simulation model effectivity and impact of the methods are indicated. It becomes possible to raise awareness for the use of the method and suggest methods that fit the situation at hand by capturing (online guide) and simulating (simulation game) the context for method application.

Keywords: Design Methods, Systems Engineering (SE), Methodical Design, Method Implementation, Advanced Systems Engineering (ASE)

1 Introduction

Adaptation of processes and application of (new) methods in product development become increasingly important as such more stakeholders are involved in development activities (Hjartarson et al., 2021). However, developers often lack systems expertise and the development context is constantly changing. Furthermore, the challenges to the validation of technical systems require immense efforts due to the increasing networking and automation of systems. (Dumitrescu et al., 2021) Companies are responding by improving their engineering departments by introducing agile and SE methods (Atzberger et al., 2020). To highlight method implementation and adaptation challenges in existing organizations, Inkermann (2021) and Gericke et al. (2020) propose the term Method EcoSystem. A Method EcoSystem is defined as a system of methods embedded in an organization in which different design variants are shared and in which users can adapt and combine different methods depending on the development task (Inkermann, 2021). This results in an increased need for a detailed analysis of organizational structures and processes to identify improvements and to introduce new methods or adapt existing ones. The range of modern engineering methods - especially from science - is very wide (Heimicke, Duehr, et al., 2021). Designing the adoption process challenges development teams to adapt the methods to contextual conditions as well as to integrate the new methods into established processes (Heimicke, Roebenack, et al., 2021; Wallace, 2011). Therefore, it is crucial to link the Method EcoSystem and context factors. Context factors are used to characterize the Method EcoSystem. Most existing approaches to provide knowledge about methods focus on process- and feature-based selection. Here, selection requires awareness and detailed knowledge about the development context and detailed goal of the (new) method. Development teams, therefore, need support in selecting and implementing appropriate methods (Albers et al., 2015). The selection of methods is determined by a mixture of process-related criteria (e.g., required output) and other characteristics (e.g., size of the team, time available for method application). To select methods, developers typically have to answer a self-discovery question dialog that is driven by an algorithm that narrows down the number of suitable methods (Albers et al., 2015; Bavendiek et al., 2018). However, this access logic mainly considers the maturity level of the respective development project, which in turn is mapped into generic phases and activities. Since the choices are very generic, a clear delineation of the possible methods is difficult for developers. To improve the selection process of suitable development methods, in this paper two different concepts of access logic are presented. The objective is to enable access to method knowledge and facilitate selection and raise awareness for method implementation. In the future, more information from the specific development context should be incorporated into the selection of the method as well as simulations on their effectiveness, making it easier for developers to explore and implement suitable methods.

2 State of the art

2.1 Methods in Product Development

Based on the system theory according to Ropohl (1975), the process of product development can be described in the extended system triple (Albers et al., 2011) through the interaction of three systems. The operating system contains all resources necessary for the realization of a development project (e.g. developers, knowledge, infrastructure). It creates the system of objectives (objectives, their rationale, dependencies, requirements, constraints) and the system of objects (all results in the product engineering process such as sketches, lists, models, prototypes, and the final product). The synthesis and analysis of the systems of objectives and objects are performed in continuous iteration cycles so that an early and continuous validation of all objects created in the process against the objectives is ensured. (Albers et al., 2011) In Figure 1, an extension of the system triple model is shown, which also highlights the support and models used in Model-based Systems Engineering. Furthermore, Figure 1 indicates two levels, one representing the human-centered control cycle and the other the support (methods and tools). An extension of the support level could also be a common understanding of systems thinking or the use of methods in the development environment. Furthermore, at the human-centered level, individual and team needs could also be included.

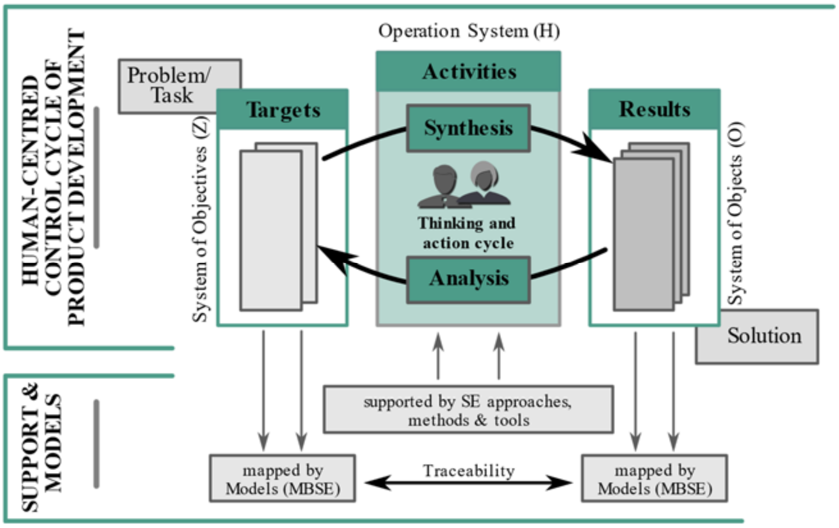

Figure 1. Extended System-Triple model to divide the development process into two levels.

When proceeding through the cycles of synthesis and analysis, the actions of product developers can be supported by the application of methods (Badke-Schaub et al., 2011). Gericke et al. (2017) define design methods as "*A specification of how a specified result is to be achieved. This may include specifications of how information is to be shown, what information is to be used as inputs to the method, what tools are to be used, what actions are to be performed and how, and how the task should be decomposed and how actions should be sequenced.*" Inkermann (2021) adds the term principle to highlight the core ideas underlying the structure of a process or method. Due to the high diversity of situations in the development context, many development methods (c.f. Section 2.2) exist that can support development teams in synthesis and analysis (Bavendiek et al., 2018; Gericke et al., 2017).

2.2 Methods in Advanced Systems Engineering

Originally, SE was developed as an interdisciplinary approach to support the development of large-scale systems. Currently, SE methods and tools lack adaptability and acceptance. This results in the demand for an advanced methodology. Albers and Lohmeyer (2012) propose Advanced Systems Engineering (ASE) as a model-based and human-centered methodology based on the fundamentals of SE (e.g. system thinking). Human-centered SE is described as a dimension of ASE that aims to support the scalability of procedures and methods for the specific application and problem context, as well as to promote acceptance (Albers & Lohmeyer, 2012). Especially in the complex environment of ASE, methods can help to overcome barriers to thinking and promote the creativity necessary for the development process. The use of methods in ASE improves cooperation between the development process' interfaces and strengthens communication between individuals for the coordination of technical and organizational processes. (Lindemann, 2009) Nowadays, (new) types or roles of methods and approaches are developing that refer to the ASE approach.

2.3 Existing Approaches to Access Methods

As Bavendiek (2018) quotes Lutters et al., (2014) "*the user of methods* […], *the level of expertise with methods* […] *and training have a direct influence on the quality, time and cost of the development of a product*". Roth and Binz (2022) analyze a variety of aspects whose consideration influences the design of context-appropriate methods. From this, an approach for the development of context-appropriate methods is derived. Recommendations for achieving appropriate context adaptation include the following: Identify the problem in the context, consider the usefulness of the method from the beginning, and develop the support provided by a method, make improvements to the users by the method (Roth and Binz, 2022). In addition, methods can be classified according to their scope of action regarding the degrees of the resolution presented in section 2.1, the underlying fields of action, and possible parameters that can be positively influenced by the methods (Heimicke, Ng, et al., 2021). Various solutions have been proposed to facilitate the selection of methods and provide knowledge about methods, c.f. e.g. Bavendiek (2018). Their basic functions are explained by two online tools. *InnoFox* is an interactive online tool that recommends methods according to the users' contexts and needs (Albers et al., 2015). Based on the selection of generic activities and generic objectives an individual set of suitable methods is provided. The description of the method's procedure provides the first overview and supports developers during the application (Albers et al., 2015). Heimicke, Ng, et al. (2021) recommend methods to develop teams based on a suitability analysis. Here, development teams are supported in the identification of relevant fields of action and possible parameters that are supposed to further develop the current development process through their method-supported improvement. Depending on the selection, methods from a catalog are suggested that show the best possible suitability for use case-specific process improvement. (Heimicke, Duehr, et al., 2021; Heimicke, Ng, et al., 2021) Attributes account for another approach to selecting a method. The online tool *Methodos* provides access to basic methods and their descriptions in German and English (Bavendiek, 2018). Here, all methods are depicted in a drop-down menu and when selected are portrayed by attributes. For selection, Bavendiek (2018) proposes a structured approach comprising four phases to obtain method attributes. In the *scan* phase, various methods are collected and filtered by descriptive method characteristics. In the *regroup* phase, those methods are sorted by similar content into attribute groups. At this stage in the analysis synonyms and translations of the same content are retained only to eventually determine one name for the attribute in the third phase. In the *define* phase, all name suggestions for one group of attributes are being compared with each other and only one attribute name is chosen according to three criteria. Having selected one attribute name for every group of attributes the content of the methods is assigned in the last phase. Depending on the depth of the method description and a detailed *regrouping* phase the number of attributes obtained varies. The main shortcoming of the two examples of existing tools is that they rely on set activities and process-maturity levels and give little attention to either the specific application context or the Operation System.

3 Research Approach

To address the weaknesses of existing approaches described in the previous chapter this paper takes a closer look at the access logic used to suggest fitting development methods as a success factor for a broader consideration of the context of the operation system and successful implementation of methods and conceptually answers the following research questions:

1. *What information from the development context needs to be considered in the selection and introduction of development methods to integrate the specific requirements of development teams into the selection?*

2. *How can the amount of information to be collected for use case-specific access be reduced when selecting development methods?*

The approach in this paper is structured following the Design Research Methodology (Blessing and Chakrabarti, 2009). After analyzing existing approaches to method access (Research Clarification), the focus on method selection and adoption was conducted based on observations of engineering departments within two research projects (Descriptive Study I). Finally, two conceptual approaches to analyze (online guide) and simulate (simulation game) the context relevant to raising awareness about methods and selecting suitable methods are proposed (Prescriptive Study). An evaluation of the concept analogous to Descriptive Study II will be conducted in future research. The research environment is provided by two research projects funded by the German Federal Ministry of Education and Research. Consortia of industrial companies and research institutes are working on each of these.

4 Results

4.1 Generic Elements of a Guide for Method Access

To propose methods context- and target-oriented it is important to identify the goal and purpose to which the method is being introduced. Moreover, a guide for method access must be able to consider the context in which the method is introduced to a high degree of detail and relevant interactions to existing methods and processes. This requires elements like specific factors, a wide range of objectives, and further criteria to represent the context and characterize different methods. The proposed distinction of factors, objectives, and criteria is explained in more detail in the following.

- **Factors:** Factors describe variables that influence the ability of an operation system to apply a method. This makes it possible to change individual factors in a targeted manner to adjust the operation system. To describe the multitude of use cases in agile and SE transformation, the factors are distributed among the groups for describing the product development context of the company, management, project, and individual according to (VDI, 2019). Thus, individual groups are referred to as *fields of action*. These fields of action are in turn based on context groups defined by Gericke et al. (2013). The 228 factors used are based on a collection of factors from literature research, expert interviews, and expert workshops (Albers et al., 2020). For the factors to have a measurable influence, they must be able to take on different values. Each factor can take on values in two areas, namely agility and SE. The value of the factor can be low, neutral, or high in both areas independently. This can lead to synergies and conflicts between the characteristics in the areas of agility and SE. For example, the factor *understanding of roles* is high for agility if there are no defined roles but responsibilities; in the SE area, the factor is high if predefined roles are used.
- **Objectives:** To identify which parameters need to be changed, a clear objective or need must be defined. The user of the guide is supported by 87 pre-formulated objectives in an intelligent and interactive question tree. These objectives were collected from various sources and then combined into seven higher-level clusters, e.g. *change management*. A large proportion of the goals are based on the ASE initiative's performance survey, which identified the needs of future engineering (Dumitrescu et al., 2021). The database contains information for each objective on which of the 228 factors must be influenced to achieve the objective.
- **Criteria:** The criteria which describe questions that can be used to query the status quo of an operation system are within the question tree. The questions are separated into the areas of agility and SE. An example of a question in the area of agility is: *How to deal*

with goals, requirements and constraints? In the area of SE, a possible question is: *Is there an overview of the interrelationships of the existing data?*

The introduced factors, objectives, and criteria serve as a basis to characterize both, the context a method is used in (method selection) as well as to characterise different methods. To identify a method that satisfies the specific demands and situation, the parameters to be adjusted are specified via the individual target selection (online guide, c.f. Section 4.2). At the same time, a reduced number of factors, objectives, and criteria can be used to build up a simplified simulation model pointing out the impact and effectiveness of methods (simulation game, c.f. Section 4.3).

4.2 Case: Guideline for implementing agile Systems Engineering Methods

The guide is intended to help organizational units at different organizational levels to integrate agile SE methods into their processes in a targeted manner which is divided into four steps. In the first step, the initial situation and the objective which is relevant to the specific context are recorded. Based on that a suitable method for further developing competence in agile SE is given in the second step. In the third step, the guide supports the user to develop an individual implementation strategy for their selected method as a team to implement the method successfully. In the last step, the user can use the guide to measure the success of the method using previously defined key performance indicators to further develop the method in an iterative process. The guide uses various elements to record the initial situation and objectives (c.f. Figure 2): 87 objectives on three levels are listed in the guideline for the selection of the main objective. The user can choose between seven superordinate goal clusters, e.g. *improve requirements management* or *improve engineering change management*. Subsequently, he/she can choose between further subordinate objectives from the respective cluster to specify his or her objective as extensively as possible. The algorithm can use the objective to identify which variables in the organization need to be changed to achieve this objective (1). For this purpose, exactly those factors are always linked to a specific objective in the database, through the adjustment of which the objective can be achieved. The factors have the characteristics of low, neutral, and high. Based on these values, the algorithm identifies which factors still offer the potential for optimization and which are already high.

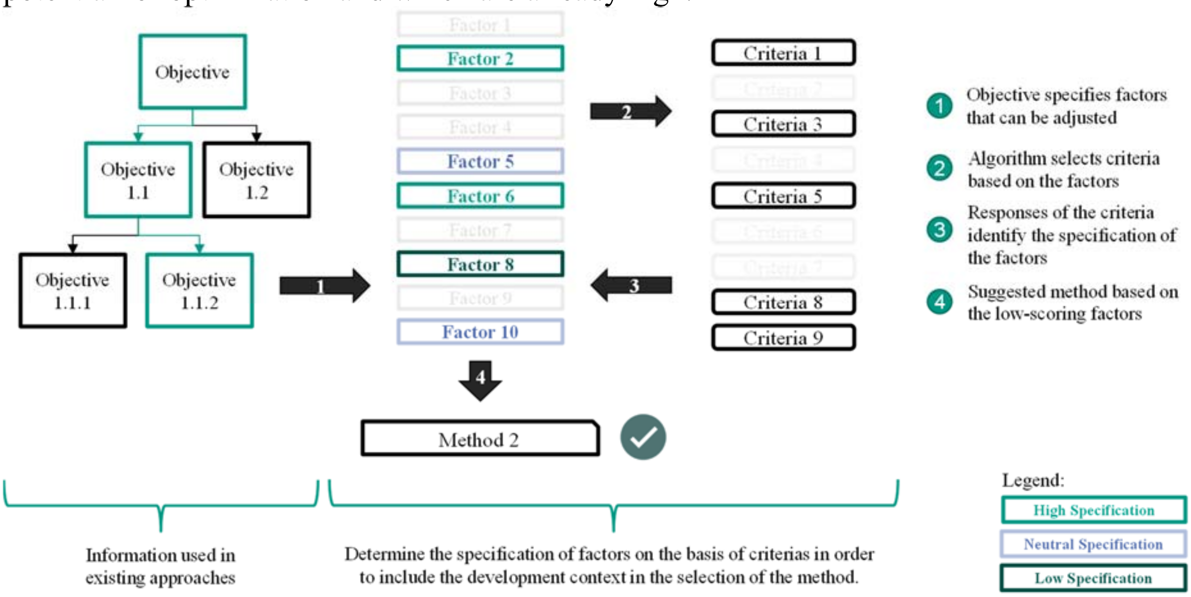

Figure 2. Assignment and characterization of the factors by criteria and method proposal.

Before the algorithm can suggest a method, it must identify how the factors are characterized in the current use case. For this purpose, questions are suggested to the user by an intelligent question tree, based on whose answer options the algorithm can determine the factor expression (2). This ensures that the context is captured and considered to a high degree of detail. Since the guide now knows, which factors are needed to improve (3), it can suggest suitable methods. This proposal is based on the fact that the database contains information on which factors can be influenced by each method. In this way, the algorithm automatically selects the method that can improve best the low-level factors (4).

4.3 Case: Simulation Game Sə’stemic

Simulation games are a promising approach to initiating transformation processes and simulating the impact of measures within an experimental space. The simulation game Sə'stemic was developed to convey and test the preconditions and interactions of different measures for the implementation of SE. Development methods like the “risk cube method” or “system context analysis” are one measure represented within the simulation game. In Figure 3, the integration of methods within the Method EcoSystem - a system of methods embedded in an organization in which different design variants are shared - is illustrated. The interactions represented are intended to raise awareness of processes, tools, or roles needed to successfully implement a method in practice (awareness level). Moreover, each method is characterized by a set of seven factors and an overview of its goals, and the basic idea is presented to the players by a card, c.f. Figure 3.

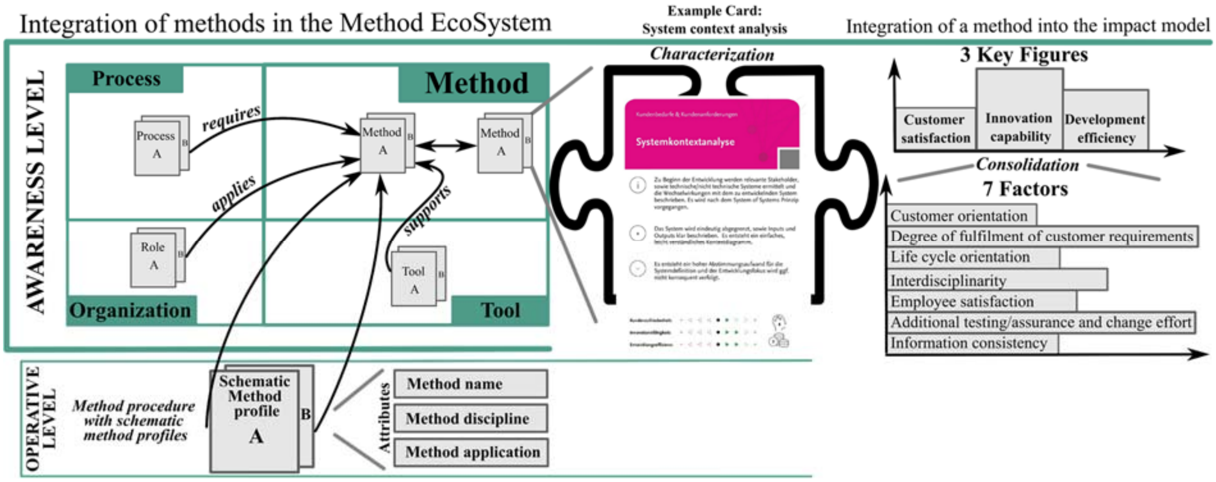

Figure 3. Access to method knowledge in the Method EcoSystem of the simulation game Sə'stemic.

Within the simulation game, method access is based on the one hand on the fact that a process needs a method and on the other hand that the organization or a role uses the method (preconditions). In addition, the method selection impacts 3 key figures (customer satisfaction, innovation capability, and development efficiency) used in the game. Within the game session, the players in a first step are called to identify those methods that have the strongest impact on the key figures. Therefore, the team of up to 16 players has to negotiate the need for different methods and prioritize their application since resources are limited. The impact of each method but also measures concerning processes (e.g. architecture definition process), tools, and roles (e.g. requirements manager) are simulated using a simplified model representing reconditions for successful application of the method and its impact on the key figures. Since in most cases a chosen method does not become effective in the first round, the players are called to analyze missing preconditions that foster the awareness needed for successful method application in practice. The first prototype of the simulation game has been developed by the Chair of Integrated Product Development at TU Clausthal and has been evaluated in test games with students. These results indicate that gamification is emerging as a target-oriented approach to raise awareness about the use of (new) development methods. To provide more detailed knowledge about the specific methods in future work method profiles will be provided (operative level, c.f. Figure 3). These method profiles are intended to guide the application of methods in engineering projects.

5 Conclusion and Outlook

Methods based on the ASE paradigm have proven to help deal with increasing complexity, the interconnectedness of systems, and the ever-growing number of functions. In particular, the methods help to break down complex problems into more manageable sub-problems, identify conflicting goals, and focus on important next steps. For an organization to identify a concrete method that helps it to develop its organization in a targeted manner, status quo guides such as InnoFox or Methodos are used to provide knowledge about methods. In both approaches, development methods are evaluated regarding their attributes such as possible goals, the method description, or required resources. However, the guides base their method proposal on firmly defined phases, activities, and goals and neglect real-world processes. This leads to methods being introduced without considering actual needs and not achieving the hoped-for positive impact. To be able to introduce methods in a more targeted way, it is necessary to better represent the existing situation within the method selection process. For this, the guide must be able to capture the context in a high level of detail. The integration of methods into the Method EcoSystem is done in the simulation game Sə'stemic, via the characterization of the methods by selected factors and the consolidation by key figures. Accordingly, the simulation game provides access to method knowledge and method selection through interactions and dependencies (over 120 interactions available in the simulation game) with other parts (processes, organization, tools). For example, a process needs a method, or an organization uses a method. In addition, it is possible to improve the three metrics through methods and thus pursue a certain goal. Based on this impact model, the game can therefore simulate what impact the introduction of a method would have on the key figures and thus provide the user with a decision-making aid for method selection. The simulation game combines the context analysis with a gamification strategy as a targeted approach to support the introduction of SE and the knowledge transfer of the basics of SE according to Walden et al. (2017) and ISO15288 (2015). The MoSyS guide can capture the context in detail with the help of 228 variables from the areas of the company, management, project, and individual. The state of the operating system is described by an expression of these parameters in the dimensions of agility and SE, which can be low, neutral, or high. Based on the user's objective, the algorithm can determine which variables need to be changed in the company to achieve the goal. Through an intelligent and individual question tree, the guide captures the expression of the relevant variables and can thus determine the individual optimisation potential based on the user's individual context. From a method database, the guide can then select exactly the method that is most helpful in realising this optimisation potential. Due to the strong inclusion of the context, both guides can suggest targeted methods. Linking the two guides can ensure that the strengths of each concept are combined. For example, the gamification approach of the Sə'stemic lends itself to increasing participation in the implementation of the guide. Using the 87 objectives and 228 factors from the MoSyS guide, the context can be captured in detail. In addition, the simulation approach can be used to show the user how the KPIs or variables might change before the method is systematically implemented. To improve the access logic in general, it is necessary to constantly integrate new objectives, factors, and questions to cover all possible context areas. The integration of these new elements is done in close cooperation with the collaborative projects of the ASE initiative and within the collaborative projects MoSyS and RePASE.

Acknowledgments

The research presented is part of the project RePASE - Reflective Process Development and Adaptation in Advanced Systems Engineering. This project is funded by the German Federal Ministry of Education and Research (BMBF) within the Program "Innovations for Tomorrow's Production, Services, and Work" (02J19B149) and managed by the Project Management Agency Karlsruhe (PTKA). The author is responsible for the contents of this publication.
This research project is funded by the German Federal Ministry of Education and Research (BMBF) within the "Innovations for Tomorrow's Production, Services, and Work" Program (MoSyS Funding reference: 02J19B090 Duration: 01.10.2020 to 30.09.2023) and implemented by the Project Management Agency Karlsruhe (PTKA). The author is responsible for the content of this publication.

References

Albers, A., Heimicke, J., Trost, S., & Spadinger, M. (2020). Alignment of the change to agile through method-supported evaluation of agile principles in physical product development. Proceedings of 30th CIRP Design Conference, 600-614. https://doi.org/10.1016/j.procir.2020.02.218

Albers, A., & Lohmeyer, Q. (2012). Advanced Systems Engineering – Towards a Model-Based and Human-Centered Methodology. Proceedings of TMCE 2012, 407–416.

Albers, A., Lohmeyer, Q., & Ebel, B. (2011). Dimensions of objectives in interdisciplinary product development projects. In S. J. Culley, B. J. Hicks, T. C. McAloone, T. J. Howard, & U. Lindemann (Eds.), Proceedings of the 18th International Conference on Engineering Design (ICED) 2011: Impacting society through engineering design. Konferenz, Lyngby/Copenhagen, Dänemark (pp. 256–265).

Albers, A., Reiß, N., Bursac, N., Walter, B., & Gladysz, B. (2015). InnoFox-Situationsspezifische Methodenempfehlung im Produktentstehungsprozess. In H. Binz, B. Bertsche, W. Bauer, & D. Roth (Chairs), Stuttgarter Symposium für Produktentwicklung (SSP), Stuttgart.

Atzberger, A., Nicklas, S. J., Schrof, J., Weiss, S., & Paetzold, K. (2020). Agile Entwicklung physischer Produkte: Eine Studie zum aktuellen Stand in der industriellen Praxis. Universität der Bundeswehr München. https://doi.org/10.18726/2020_5

Badke-Schaub, P., Daalhuizen, J., & Roozenburg, N. (2011). Towards a Designer-Centered Methodology: Descriptive Considerations and Prescriptive Reflections. In The Future of Design Methodology (pp. 181–197). Birkhofer, Herbert.

Bavendiek, A.-K. (2018). Adaptation of engineering design methods for multidisciplinary development processes considering heterogeneous teams. Dr. Hut. https://doi.org/10.24355/dbbs.084-201810111220-0

Bavendiek, A.-K., Inkermann, D., & Vietor, T. (2018). Interrelations Between Processes, Methods, and Tools in Collaboratve Design – A Framework. In A. Maier, H. Kim, J. Oehmen, F. Salustri, S. Škec, & M. Kokkolaras (Eds.), DS: 87, 8. Human behaviour in design (pp. 349–358). Red Hook, NY: Curran Associates Inc.

Blessing, L. T., & Chakrabarti, A. (2009). DRM, a Design Research Methodology. London: Springer. Retrieved from http://dx.doi.org/10.1007/978-1-84882-587-1

Dumitrescu, R., Albers, A., Riedel, O., Stark, R., & Gausemeier, J. (2021). Engineering in Deutschland – Status quo in Wirtschaft und Wissenschaft: Ein Beitrag zum Advanced Systems Engineering. Paderborn.

Gericke, K., Eckert, C., Campean, F., Clarkson, P. J., Flening, E., Isaksson, O., . . . Wilmsen, M. (2020). Supporting designers: moving from method menagerie to method ecosystem. In Design Science, Vol. 6 (Vol. 6, p. 49). https://doi.org/10.1017/dsj.2020.21

Gericke, K., Eckert, C. M., & Stacey, M. K. (2017). What do we need to say about a Design Method? 21st Int. Conf. on Eng. Design, ICED17.

Gericke, K., Meißner, M., & Paetzold, K. (2013). Understanding the context of product development. In DS 75-3: Proceedings of the 19th International Conference on Engineering Design (ICED13) Design For Harmonies, Vol. 3: Design Organisation and Management, Seoul, Korea 19-22.08. 2013. (Vol. 3, pp. 191–200).

Heimicke, J., Duehr, K., Krüger, M., Ng, G.-L., & Albers, A. (2021). A framework for generating agile methods for product development. Procedia CIRP, 100, 786–791. https://doi.org/10.1016/j.procir.2021.05.043

Heimicke, J., Ng, G.-L., Krüger, M., & Albers, A. (2021). A systematic for realizing agile principles in the process of mechatronic systems development through individual selection of suitable process models, methods and practices. Procedia CIRP, 100, 619–624. https://doi.org/10.1016/j.procir.2021.05.133

Heimicke, J., Roebenack, S., Frobieter, C., Tihlarik, A., Albert, B., Bramato, L., . . . Albers, A. (2021). Evaluation of Challenges in the Implementation of Scrum in a large German Plant Engineering Company: Derivation of Hypotheses for an Improved Introduction of Agile Approaches into the Processes of Physical Product Development. In R&D Management Conference 2021 - Innovation in an Era of Disruption (n.p.). online.

Hjartarson, B., Daalhuizen, J., & Gustafsson, K. F. (2021). The Dark Side of Methods – An Exploration of the Negative Effects of Methods Use and Method Reflection in Design. Proceedings of the Design Society, 1, 3001–3010. https://doi.org/10.1017/pds.2021.561

Inkermann, D. (2021). Shaping Method Ecosystems – Structuted Implementation of Systems Engineering in Industrial Practice. In Proc. of Int. Conf. on Eng. Design (ICED21) (Vol. 1, pp. 2641–2650). https://doi.org/10.1017/pds.2021.525

ISO15288 (2015). ISO15288:2015 Systems and software engineering. System life cycle processes.

Lindemann, U. (2009). Methodische Entwicklung technischer Produkte: Methoden flexibel und situationsgerecht anwenden. VDI-Buch. Berlin Heidelberg: Springer-Verlag Berlin Heidelberg.

Lutters, E., van Houten, F. J., Bernard, A., Mermoz, E., & Schutte, C. S. (2014). Tools and techniques for product design. CIRP Annals, 63(2), 607–630. https://doi.org/10.1016/j.cirp.2014.05.010

Ropohl, G. (1975). Einleitung in die Systemtechnik. In G. Ropohl (Ed.), Systemtechnik, Grundlagen und Anwendung (pp. 1–77). München, Wien: Carl Hanser Verlag.

Roth, D. & Binz, H. (2022). Context-Adapted Methods of Modern Product Development: Recommendations and Best Practice Examples. In D. Krause & E. Heyden (Eds.), Design Methodology for Future Products: Data Driven, Agile and Flexible (pp. 167–182). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-78368-6_9

VDI. Entwicklung technischer Produkte und Systeme. (VDI-Richtlinie 2221 Blatt 1 und Blatt 2). Berlin: Beuth Verlag: Beuth Verlag.

Walden, D. D., Roedler, G. J., Forsberg, K. J., Hamelin, R. D., Shortell, T. M., & Kaffenberger, R. (Eds.) (2017). INCOSE Systems Engineering Handbuch: Ein Leitfaden für Systemlebenszyklus-Prozesse und -Aktivitäten : INCOSE-TP-2003-002-04 2015 (Deutsche Übersetzung der vierten Ausgabe). München: GfSE e.V.

Wallace, K. (2011). Transferring Design Methods into Practice. In H. Birkhofer (Ed.), The Future of Design Methodology (pp. 239–248). London: Springer London. https://doi.org/10.1007/978-0-85729-615-3_21