Situation-specific Development of Business Models for Service Providers in Software Ecosystems*

Sebastian Gottschalk

Software Innovation Lab, Paderborn University, Paderborn, Germany
sebastian.gottschalk@uni-paderborn.de

Abstract. Effective business models are essential for a service provider in a software ecosystem to stay successful. To support those business models' development, different domain experts propose various methods to develop such business models and provide knowledge about successful business models. However, both the methods and the knowledge need to fit the service provider's situation and the actual service. Otherwise, the development of an ineffective business model can lead to poor market penetration of the services or even the service provider's bankruptcy. Currently, no existing business model development approach provides fully-fledged tailoring to the service provider's current situation. In this thesis, we address this problem by introducing a holistic approach to support the business model's situation-specific development for a service provider. The approach introduces the role of a domain expert, a method engineer, and a business developer together with a repository of methods for developing business models and a repository of knowledge for supporting the development. Both repositories are based on the experience of domain experts. Out of these repositories, situation-specific process models for developing business models can be tailored by the method engineer and enacted by the business developer. We demonstrate our approach's feasibility with an open-source implementation and evaluate it with a case study of developing business models for a mobile app.

Keywords: Business Model Development · Situational Method Engineering · Service Provider · Software Ecosystems · Tool-support

1 Introduction

The development of effective business models, defined by Osterwalder et al. as "the rationale of how the organization creates, delivers, and captures value" [24], is an essential task for a company to stay competitive. This is one of the results of the GE Innovation Barometer 2018 [13], a study with over 2000 business executives. In this study, 64% of these executives have the "difficulty to define an

* This work was partially supported by the German Research Foundation (DFG) within the Collaborative Research Center "On-The-Fly Computing" (CRC 901, Project Number: 160364472SFB901)

effective business model to support new ideas and make them profitable" [13]. An important reason for this is that customers want solutions for perceived needs rather than just product which affects that the business model can often be more important than the latest technology of the product [9]. Here, especially software ecosystems, tend to standardize more and more technology parts and make them available for all service providers within their software development kits. Therefore, service providers must focus increasingly on their business model to build successful services. To support the business model development, different domain experts propose various methods to develop such business models in the form of development processes (e.g. [23]) and method repositories (e.g. [4]). Moreover, these experts provide knowledge in the form of taxonomies of possible (e.g. [1]) and patterns of successful (e.g. [12]) companies. However, both the methods and the knowledge need to fit the situation of the service provider and its service to support the effective development of business models. Otherwise, the development of an ineffective business model can lead to a poor market penetration of the services or even a company bankruptcy. For example, in a situation with a high problem complexity the service aims to solve, conducting expert interviews instead of customer interviews could lead to an increased understanding of the problem domain. Moreover, in a situation with business users as the service's target group, data sovereignty and personal customer relationship may be more important than by targeting private users. Although various business model development approaches have been proposed, none of them provides fully-fledged tailoring to the service provider's situation. This corresponds to the general lack of it-support for business model development [27], current limitations of tools in practice [26], and the automation of the development [7].

In this thesis, we address this problem by introducing a holistic tool-supported approach to support the business model's situation-specific development for service providers. The approach introduces the role of the *Meta-Method Engineer*, the *Method Engineer*, the *Domain Expert*, the *Business Developer* and other *Stakeholder*. In contrast to other business model development approaches, our approach points out the importance of the *Method Engineer* who formalizes the methods and knowledge to make them useable for the *Business Developer*. In our approach, the *Meta-Method Engineer* needs to create the meta-model to handle methods and knowledge once. Based on that, the *Method-Engineer* models existing methods and knowledge in repositories based on the experience of the *Domain Expert*. After that, the *Method-Engineer* constructs the development methodology out of both repositories based on the described situation of the *Business Developer*. Finally, the *Business Developer* enacts the methodology to develop his business model together with the *Stakeholder*. We implement the whole approach as a web-based open-source tool and evaluate it with a case study of creating a business model for a local event platform's mobile app.

1.1 Problem Domain and Research Question

One of the most grown areas in the last years is software ecosystems (SECOs). SECOs are defined by Bosch et al. as "a software platform, a set of internal and external developers and a community of domain experts in service to a community of users that compose relevant solution elements to satisfy their needs" [5]. Here, especially external developers (i.e., service providers) can profit from the existing community of potential users together with the scalability of providing software services in general. Nevertheless, these developers have the disadvantage of high competition inside those markets together with short update/lifetime cycles of those software services. Consequently, these service providers are in need of building effective business models for their services to satisfy the user's needs. Here, the building of business models can be supported by the knowledge of domain experts inside those software ecosystems. Therefore, this thesis aims to support service providers with a software-based situation-specific business model development approach by providing an answer to the following research question:

- **RQ:** How to enable the tool-supported, situation-specific development of business models for service providers within software ecosystems?

We apply our approach to the development of business models for apps in mobile ecosystems. We choose this application area because, with over 218 billion downloads and 142 billion of revenue in 2020, mobile ecosystems are the biggest markets for service [2]. Moreover, developers are getting more and more restricted by the ecosystem providers like with the usage of regulated payment systems and standardized software development kits.

1.2 Research Method

This thesis uses a design science research (DSR) process to build an approach for the situation-specific development of business models for service providers. We use DSR because it focuses on developing an artifact with the intention of a stepwise improvement using cycles. Here, the output in the form of evaluated results of a cycle is used as the next cycle's input. As a concrete method, we choose the DSR cycle of Kuchler and Vaishnavi [21]. The process is shown in Fig. 1 and consists of three cycles with the five steps of taking *Awareness of [the] Problem*, making *Suggestion* for the solution, the *Development* of a corresponding artifact, the *Evaluation* of our solution, and the drawing of a *Conclusion*.

In the *First Cycle* (2019-2020), we analyzed the current literature and software tools to understand the problem of software-based business model development. Based on that, we created conceptional parts for the situation-specific development of business models, implement them in software fragments, and evaluate the technical feasibility. Moreover, we have provided a tool review of decision support within the so-called Business Model Development Tools (BMDTs).

Currently, in the *Second Cycle* (2020-2021), we take the lessons learned from the last cycle and the tool review to create an integrated concept for the situation-specific development of business models. For this, we implement a software tool and evaluate it with a case study on creating the business model for the mobile app of a local event platform. Moreover, we use our existing tool review to create a reference architecture for BMDTs that researchers can use.

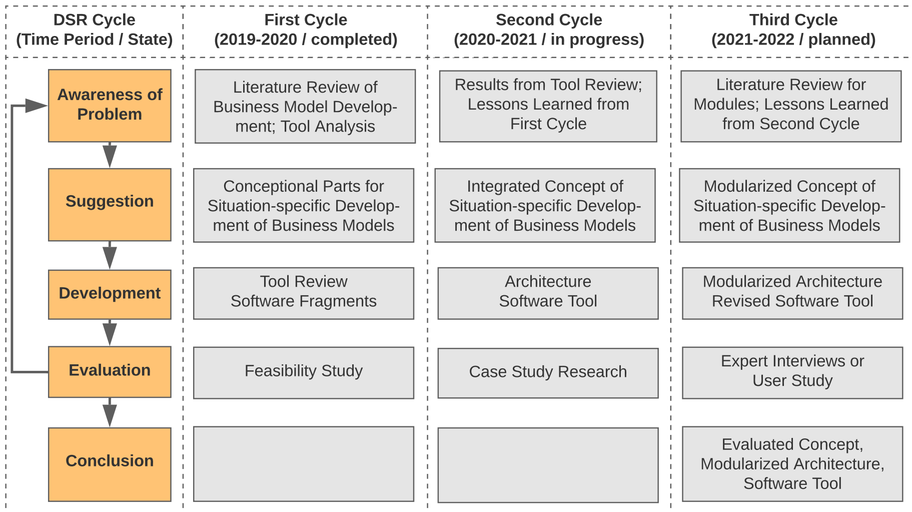

Fig. 1. Design Science Research Process based on Kuechler and Vaishnavi [21]

In the *Third Cycle* (2021-2022), we will take the lessons learned from the second cycle to create a modularized concept of situation-specific development of business models. The modularization will consist of a core architecture that different modules can extend to provide decision-support in various enacted process parts. Here, we will also review additional literature for modules we are developing. After implementing the core architecture and the modules, we will evaluate the approach with expert interviews or a user study. Finally, we conclude with an evaluated concept, a modularized architecture, and a software tool.

2 Background & Related Work

2.1 Business Model Development

The process of Business Model Development (BMD) is a continuous and challenging task, which often uses creativity and collaboration between different stakeholders. It consists of several phases (e.g., ideation, implementation) where different possible business models have to be created and validated within the market. This, in turn, can be done by conducting experiments with the potential customers of the product/service [23]. To provide flexibility in the method construction, a repository of experiments with different experiment sequences based on the type of business is introduced [4]. Moreover, the flexibility can be supported by alternative choices for process steps inside the method [25]. Nevertheless, those approaches focus on high abstraction levels and one-size-fits-all methods that cannot cover all relevant contextual factors of the situation. Different artifacts and tools can support this process. One group of artifacts are canvas models like the Value Proposition Canvas or the Business Model Canvas (BMC). The BMC [24] divides the business model into nine building blocks where each block consists of different elements. Moreover, tools in the form of repositories like pattern databases [12] or software-based tools [26] can be used. These software-based tools are often called Business Model Development Tools (BMDTs) and provide different guidance levels to develop new and improve existing business models [26]. An analysis of tools in practice [26] shows that these tool focus on the design of business modeling but not on the actual decision support. BMDTs already introduce possible parts of decision support in research. For example, BMDTs can support different phases (i.e., analysis, design, implementation, management) to guide the development process [11]. Moreover, the knowledge of business models can be supported by a reference database of existing business models [10], the usage of pattern repositories for guiding the development process [22] or domain-specific modeling based on a shared vocabulary [6]. Nevertheless, those approaches focus either on the process or the structure of business models and do not consider the situation of the provider.

2.2 Situational Method Engineering

Situational Method Engineering (SME) [20] has its origin in creating software development methods and typically consists of the two roles of a method engineer and a project manager. Here, the method engineer analyzes various methods and stores them in a method base. After that, the method engineer identifies the project's situational factors and constructs a suitable method of the method base. This method, in turn, is then enacted by the project manager to manage the project. The underlying method base can consist of method fragments and method components. A method fragment is a reusable atomic block of a method that can have a process (called work unit), a product (called work product), or a producer focus [8]. A method component consists of inputs and outputs of work products together with a process to transform the input into the output [20]. We will use the naming method elements for a method fragment and method building block for a method component to stick to the business model terminology. Moreover, we use the term of method pattern to note sequences of method building blocks.

While most of the existing approaches focus on developing software products, some also include business-related parts to their methods base. Here, a case study [3] identities different situational factors for the business (e.g., size of the business unit team), the customer (e.g., number of customers), market characteristics (e.g., market size), product characteristics (e.g., product lifetime), and stakeholder involvement (e.g., partner involvement) for phases in product management. An SME approach of IoT development methods [14] also includes business-related (e.g., regulations) and customer-related (e.g., domain experience) situational factors together with business-related (e.g., IoT Canvas) work products. Nevertheless, those approaches cover the business aspect as one side aspect of the product development process. With this, they do not consider the BMD as a separate continuous process with its characteristics.

3 Proposed Solution

Based on *DSR*, we propose the situation-specific development of business models for service providers. For this, we analyzed the business models of various software ecosystems to create a variability model of the ecosystem provider's business model and its dependencies to the service providers and the users [16]. After that, in the *First Cycle*, we have decided to focus on the mobile ecosystem's application domain by creating a knowledge model for mobile app providers' business models based on feature models [17]. Moreover, we created a process to change to create and adapt those knowledge bases [18]. Based on that, in the *Second Cycle*, we have worked on creating development methods out of a method repository [19] and using knowledge out of a knowledge repository [15], both based on the proposed case study. For that, we have developed the BMDL Method Modeler[1] and the BMDL Feature Modeler[2]. We currently integrate both solutions into each other to enact the development methodology and conduct the case study. The proposed solution is the expected outcome of the *Third Cycle* and should be evaluated based on a user study or expert interviews.

An overview of the approach can be seen in Fig. 2. It consists of the roles of the *Meta-Method Engineer*, the *Method Engineer*, the *Domain Expert*, the *Business Developer*, and other *Stakeholder* together with the stages of *A) Business Model Development Method Engineering*, *B) Business Model Knowledge Model Engineering*, *C) Business Model Development Method Construction* and *D) Business Model Development Method Enaction*.

The *A) Business Model Development Method Engineering* is used to provide the creation of method parts for the situation-specific development. For this, the *Meta-Method Engineer* creates a meta-model for the *Business Method Repository* and the *Business Development Method Patterns*. Here, the repository stores method building blocks that are configured out of atomic method elements. These elements can be the possible situational factors (e.g., market size), the different method types (e.g., discovery), the performed tasks (e.g., conduct customer interview), the involved stakeholder (e.g., designer), the created internal (e.g., Business Model Canvas) or external artifacts (e.g., prototype) and the used internal (e.g., canvas comparison) or external tools (e.g., prototyping tool). These building blocks are arranged within nested patterns by considering their type, which can themselves have situational factors and types together with a notation pattern based on BPMN. The repository and patterns are created by the *Method Engineer* based on the experience of the *Domain Expert*. By considering the SME terminology, this part focuses on process method fragments.

The *B) Business Model Knowledge Model Engineering* is used to provide the creation of knowledge parts for the situation-specific development. For this, the *Meta-Method Engineer* creates a meta-model for the *Canvas Knowledge Repository* and the *Canvas Models*. Here, the repository stores knowledge building blocks that are configured out of knowledge elements. These elements are mod-

[1] Method Modeler: `https://sebastiangtts.github.io/bmdl-method-modeler/`
[2] Feature Modeler: `https://sebastiangtts.github.io/bmdl-feature-modeler/`

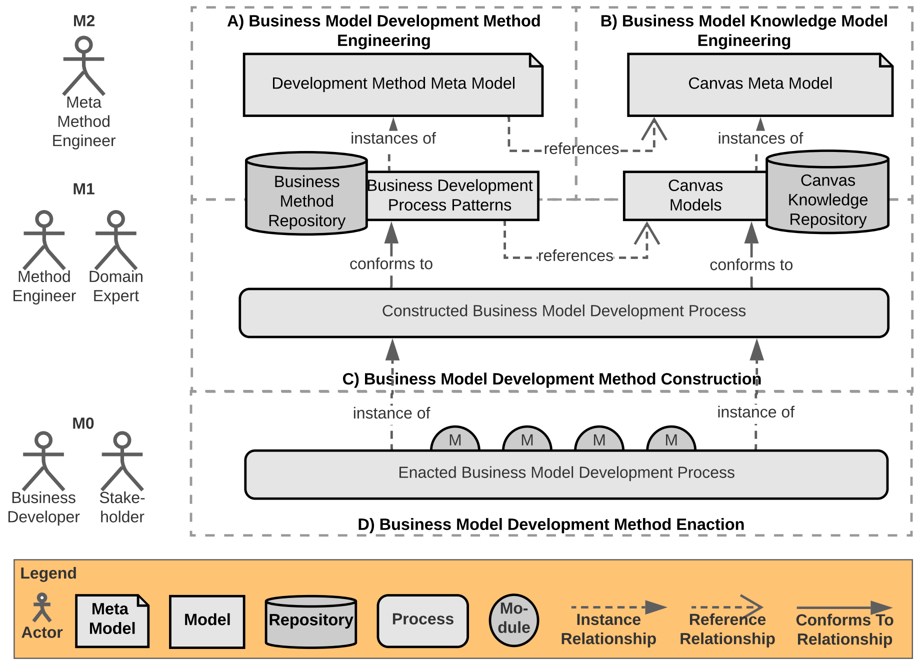

Fig. 2. Overview of Situation-specific Business Model Development

eled into hierarchies (e.g., subscription to monthly subscription) based on the concept of feature models with additional relationships for supporting (e.g., professional user supports subscription) and hurting (e.g., advertisement hurts privacy) elements. These building blocks are arranged within different canvas models (e.g., Value Proposition Canvas, Business Model Canvas) to support different phases of the business model development with existing knowledge. For each canvas model, we store the existing knowledge (e.g., modeled as feature model) that can be used as a library for recommendations together with concrete canvas models (e.g., instances of the feature model) to show existing examples and possible patterns. The repository and canvas models are created by the *Method Engineer* based on the experience of the *Domain Expert*. By considering the SME terminology, this part focuses on product method fragments.

The *C) Business Model Development Method Construction* is used to construct the development method out of the development method engineering and the knowledge model engineering. For this, the *Method-Engineer* models the described situation of the *Business Developer* with the existing situational factors. These situational factors are used to recommend specific method patterns out of the *Business Development Method Patterns*. These method patterns are nested into each other and, finally, filled with the method building blocks of the *Business Method Repository*. During this filling, he chooses the corresponding knowledge building blocks from the *Canvas Knowledge Repository* based on the linkage between the internal artifacts and the *Canvas Models*. This should ensure knowledge recommendations in the different development steps.

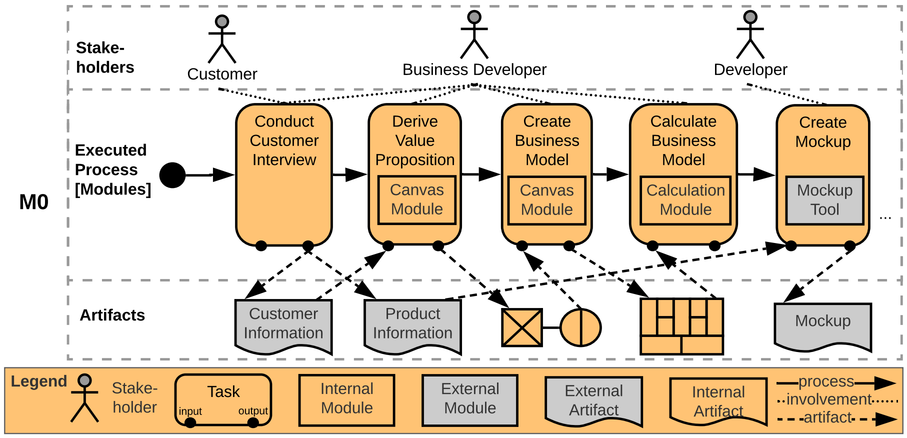

Fig. 3. Enaction of the Development Methodology

The *D) Business Model Development Method Enaction* is used to enact the constructed development method. An overview of the methodology enaction by the *Business Developer* can be seen in Fig. 3. Here, the methodology consists of the *Executed Process* in the form of tasks (e.g., calculate business model) with corresponding *Stakeholders* (e.g., business developer). The process steps can be linked to *Internal Modules* (e.g., calculation module) or *External Modules* (e.g., mockup tool). While the *Internal Modules* can directly interact with the *Internal Artifacts* (see input/output of artifacts) in the form of different enhanced canvas models, and, therefore, be used inside the software-tool, the *External Modules* provide more flexibility by also using *External Artifacts* outside the software-tool. During the enaction, the *Business Developer* will see a visualization of the process in the form of a Kanban-board, which also allows the collaboration with other stakeholders that are mentioned in the specific task. Moreover, during the enaction, the method might be adapted due to the service provider's changing situation or results of process steps.

4 Conclusion and Expected Contributions

This paper has presented our DSR-based approach's current state for the situation-specific business model development approach for service providers within software ecosystems. The method-engineer constructs the development process out of a method repository and a knowledge repository that domain experts fill in the approach. The business developer then enacts this development method to create a business model for his service. In the past, we conducted the first cycle by creating our solution's conceptual fragments and evaluating their technical feasibility. Currently, in the second cycle, we integrated those fragments into each other and planned the evaluation with a case study for the business model development for a local event app. In the future, we will modularize our approach and provide a final evaluation in the form of a user study or expert interviews.

By using DSR, we expected the evaluated concept, the modularized architecture, and the software tool as an output. With this output, we also contribute to research and practice in the following way: First, the evaluated concept provides a new view of how SME could be applied to the business modeling domain. Second, the modularized architecture supports researchers with a reference model to build new BMDTs. Third, the software tool supports practitioners in developing effective business models for their services.

Acknowledgements

I want to thank Prof. Dr. Gregor Engels (engels@uni-paderborn.de) from the database and information systems group (`https://cs.uni-paderborn.de/dbis/`) of Paderborn University for the supervision of this thesis.

References

1. Altmann, J., Ion, M., Bany Mohammed, A.A.: Taxonomy of Grid Business Models. In: Grid Economics and Business Models, vol. 4685, pp. 29–43. Springer (2007)
2. App Annie Inc: The State of Mobile 2021, `https://www.appannie.com/en/go/state-of-mobile-2021/`
3. Bekkers, W., van de Weerd, I., Brinkkemper, S., Mahieu, A.: The Influence of Situational Factors in Software Product Management: An Empirical Study. In: International Workshop on Software Product Management. pp. 41–48. IEEE (2008)
4. Bland, D.J., Osterwalder, A.: Testing business ideas. John Wiley & Sons, Hoboken (2020)
5. Bosch, J., Bosch-Sijtsema, P.: From integration to composition: On the impact of software product lines, global development and ecosystems. Journal of Systems and Software **83**(1), 67–76 (2010)
6. Boßelmann, S., Margaria, T.: Guided Business Modeling and Analysis for Business Professionals. In: Service Business Model Innovation in Healthcare and Hospital Management, pp. 195–211. Springer (2017)
7. Bouwman, H., de Reuver, M., Heikkilä, M., Fielt, E.: Business model tooling: where research and practice meet. Electronic Markets **30**(3), 413–419 (2020)
8. Brinkkemper, S.: Method engineering: engineering of information systems development methods and tools. Information and Software Technology **38**(4), 275–280 (1996)
9. Chesbrough, H.: Business model innovation: it's not just about technology anymore. Strategy & Leadership **35**(6), 12–17 (2007)
10. Di Valentin, C., Emrich, A., Werth, D., Loos, P.: Business Modeling in the Software Industry: Conceptual Design of an Assistance System. In: Practice-Driven Research on Enterprise Transformation, vol. 151, pp. 34–45. Springer (2013)

11. Ebel, P., Bretschneider, U., Leimeister, J.M.: Leveraging virtual business model innovation: a framework for designing business model development tools. Information Systems Journal **26**(5), 519–550 (2016)
12. Gassmann, O., Frankenberger, K., Csik, M.: The business model navigator: 55 models that will revolutionise your business. Pearson, Harlow (2014)
13. General Electric Inc: GE Global Innovation Barometer 2018, `https://www.ge.com/reports/innovation-barometer-2018/`
14. Giray, G., Tekinerdogan, B.: Situational Method Engineering for Constructing Internet of Things Development Methods. In: Business Modeling and Software Design, vol. 319, pp. 221–239. Springer (2018)
15. Gottschalk, S., Kirchhoff, J., Engels, G.: Extending Business Model Development Tools with Consolidated Expert Knowledge (accepted). In: Business Modeling and Software Design. Springer (2021)
16. Gottschalk, S., Rittmeier, F., Engels, G.: Business Models of Store-Oriented Software Ecosystems: A Variability Modeling Approach. In: Business Modeling and Software Design, vol. 356, pp. 153–169. Springer (2019)
17. Gottschalk, S., Rittmeier, F., Engels, G.: Intertwined Development of Business Model and Product Functions for Mobile Applications: A Twin Peak Feature Modeling Approach. In: Software Business, pp. 192–207. Springer (2019)
18. Gottschalk, S., Rittmeier, F., Engels, G.: Hypothesis-driven Adaptation of Business Models based on Product Line Engineering. In: International Conference on Business Informatics (CBI). IEEE (2020)
19. Gottschalk, S., Yigitbas, E., Nowosad, A., Engels, G.: Situation-specific Business Model Development Methods for Mobile App Developers (accepted). In: Enterprise, Business-Process and Information Systems Modeling. Springer (2021)
20. Henderson-Sellers, B., Ralyté, J., Ågerfalk, P.J., Rossi, M.: Situational Method Engineering. Springer, Heidelberg (2014)
21. Kuechler, B., Vaishnavi, V.: On theory development in design science research: anatomy of a research project. Eur. J. Inf. Syst. **17**(5), 489–504 (2008)
22. Lüdeke-Freund, F., Bohnsack, R., Breuer, H., Massa, L.: Research on Sustainable Business Model Patterns: Status quo, Methodological Issues, and a Research Agenda. In: Sustainable Business Models, pp. 25–60. Springer (2019)
23. McGrath, R.G.: Business Models: A Discovery Driven Approach. Long Range Planning (43), 247–261 (2010)
24. Osterwalder, A., Pigneur, Y.: Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. John Wiley & Sons, Hoboken (2010)
25. Simmert, B., Ebel, P.A., Peters, C., Bittner, E.A.C., Leimeister, J.M.: Conquering the Challenge of Continuous Business Model Improvement. Business & Information Systems Engineering **61**(4), 451–468 (2019)
26. Szopinski, D., Schoormann, T., John, T., Knackstedt, R., Kundisch, D.: Software tools for business model innovation: current state and future challenges. Electronic Markets **60**(11), 2794 (2019)
27. Veit, D., Clemons, E., Benlian, A., Buxmann, P., Hess, T., Kundisch, D., Leimeister, J.M., Loos, P., Spann, M.: Business Models. Business & Information Systems Engineering **6**(1), 45–53 (2014)