Designing a Reference Model for Digital Product Configurators

Sarah Hönigsberg[1], Christoph Kollwitz[1], and Barbara Dinter[1]

[1] Chemnitz University of Technology, Business Information Systems I, Chemnitz, Germany
{sarah.hoenigsberg, christoph.kollwitz, barbara.dinter}@wirtschaft.tu-chemnitz.de

Abstract. Since the manufacturing industry is nowadays facing increasingly heterogeneous customer requirements, digital product configurators (DPC) have become a popular means for integrating customers into organizational value creation. DPC are information systems, which serve as a frontend to the customer and enable the individualization of products. The design of such a DPC is time consuming, expensive and lacks appropriate models offering guidance for its development. The paper at hand addresses these issues by providing a reference model (RM) for DPC development. The model has been constructed by means of an extensive literature review and was subsequently demonstrated and evaluated in a real world scenario. In order to ensure a flexible and individual development of company specific DPC the RM includes adaptation mechanisms. Therefore, our research provides a first building block to the endeavor of facilitating or even automating DPC development.

Keywords: Reference Model, Product Configurator, Information System Modeling, Software Engineering.

1 Introduction

Nowadays organizations experience a business environment, where the focus has shifted from mass produced goods to individual products tailored to customer needs [1]. This trend towards personalization was mainly driven by the development of information and production technology [2]. The customer's active involvement in the product design serves as an example, whereby so-called digital product configurators (DPC) enable user-friendly and fast communication and the collection of customer preferences regarding the product customization [3]. Well-known examples are car configurators enabling customers to personalize their car by choosing e.g. interior, engine types, or color. Although DPC promise many advantages, such as reducing process cycle times through automation, decreasing faulty orders, or increasing sales [4], they exhibit downsides, too. High development efforts and costs as well as the difficult maintainability of DPC have to be emphasized [5, 6]. Especially for small organizations, developing and deploying a DPC is challenging and time-consuming. At the same time the DPC has a huge impact on their business and becomes a key system that centralizes the exchange of product, tender, and customer data [7]. This effect

14th International Conference on Wirtschaftsinformatik,
February 24-27, 2019, Siegen, Germany

occurs because the DPC supports one of the company's core processes (selling the product in combination with customer contact) and thereby requires access to adjacent systems such as customer relationship management, enterprise resource planning, or product data management systems [8]. In small companies without such information systems (IS), parts of these IS must be integrated into the DPC, e.g. to store the product specification necessary for the configuration task. This leads to very complex DPC and highly individual company requirements towards it. Missing guidelines for a sound development of DPC constitute a major challenge often resulting in an ad hoc development of these complex IS without conceptual planning and in problems such as the entanglement of software layers, e.g. the graphical user interface (GUI) in the frontend and the business logic in the backend [6].

While researchers have addressed many different aspects of DPC development in an isolated way such as product knowledge modeling, principles for designing GUI, or reasoning algorithms [9] and studies have been conducted examining several hundred DPC [10], a consolidation of the results from a variety of research fields such as economics, innovation or production research is missing. Therefore, a transformation of the acquired knowledge from the different research fields to the IS research as a holistic conceptual view for the development of DPC is absent. On the one hand, the motivation to build a guidance supporting the DPC software developers lies in the importance of DPC in the modern business environment and on the other hand in the challenges of their development. To guarantee that this guidance is suitable for as many organizations as possible, an artifact representing the best practices for DPC is needed. This artifact should be reusable and universally applicable for many cases. Reference models (RM) are particularly suitable for such purposes. They support system design and development since they streamline the design process and facilitate the implementation by providing best practices, recommendations and represent the knowledge in a particular field [11].

Taking the mature research field of DPC into account, the paper at hand aims to consolidate prior research findings from the last decades to this novel, overarching artifact. The authors develop a RM for DPC as an IS on a conceptual level, based on a secondary literature study including prior literature reviews and studies. This approach leads to the following research question (RQ): *How does a DPC RM need to be built in terms of design decisions and scope to make existing knowledge accessible to IS scholars and practitioners?* To answer this RQ, the remainder of the paper is organized as follows: The (1) introduction with the problem definition is followed by the (2) DPC foundations, and then the (3) research approach is described. Afterwards, the (4.1) DPC characteristics are derived from literature. Using this foundation the RM is (4.2) built, (5) applied, and evaluated. The paper ends with the (6) discussion of the findings and an (7) concluding section.

2 Foundations

Product configuration as a sub-class of configuration in general, constitutes a particular type of design activity. The product to be designed is composed of a set of predefined components that can only be combined in a certain set of combinations, which are called solution space [12]. In order to address and facilitate this often complex task, IS are used, which contain the knowledge about the components and the rules how these can be combined [13]. DPC can support quotation, marketing, and rapid manufacturing of customized products in various industries from car manufacturers to software vendors [9, 13]. The application of DPC in this context has already been discussed in academia and practice since the late 1970s [14]. In DPC research, topics ranging from system design to the impact of configuration have been discussed [9]. Research on this topic, however, often revolved around user or customer interaction. Aspects such as complexity perception, customer satisfaction, and user interaction patterns were addressed [5] in order to optimize the experience with DPC. Not only the experience or fun of configuring products was discussed, but also how DPC as an advisory systems can help customers find the right product to meet their needs [8]. In contrast, a study in which 500 online business-to-customer DPC were evaluated and categorized by features, indicates that they are often still parameter-based, decision-focused, and technology-oriented [10]. This leads to the conclusion that the customer experience is widely discussed in theory, however not properly addressed in the real world. In addition, good and bad practices of development were presented, e.g. a study of 111 web sales DPC raised questions about the presentation of configuration options, the handling of restrictions, and the support of configuration processes [6].

A more comprehensive view of DPC is provided in Subsection 4.1, where a systematic literature review is carried out to analyze the domain as an integral part of reference modeling. The reference modeling itself is described in Section 3 below.

3 Research Approach

To design the RM for DPC, the development process according to [15] has been used in conjunction with an evaluation framework for RM [16] and the design science research approach [17]. In contrast to a (project or enterprise) specific model that would describe a specific DPC, a RM for DPC describes this class of IS in an idealized way [15]. The method of reference modeling can be divided into two phases: (A) construction of the RM and (B) usage of the resulting RM [15]. Both phases are each divided into four individual steps. Common construction steps are (A.1) problem definition, (A.2) conceptualization, (A.3) evaluation, and (A.4) maintenance of a RM. Typical usage steps include (B.1) selection, (B.2) customization, (B.3) integration, and (B.4) application of the RM to solve a problem [15]. The paper at hand addresses the steps (A.1) to (A.3) and (B.2). For (A.1) a systematic literature review was conducted [18, 19] (cf. Subsection 4.1). We have focused on existing research results and DPC studies to consolidate the previous extensive research into a secondary study [20]. After the DPC characteristics have been derived, the (A.2) conceptualization follows, in which adequate views, such as a functional view and the corresponding modeling language, are defined (cf. Subsection 4.2). The multiperspectivity constitutes an important modeling concept, meaning that the same IS is described from multiple viewpoints. In addition to the views, the adaptation mechanisms are developed in the conceptualization step. In general, adaptive RM are models that can be adjusted to a particular application and/or condition. Adaptation mechanisms are needed to guarantee that the RM is suitable for different situations and to provide a solution to the reference modeling dilemma, which states: the more specific a RM addresses an organizational problem the more likely the company will choose it, but the more specific a RM is the fewer companies are addressed by it [21]. The focus on adaptive reference modeling in the paper at hand is justified by the wide range of DPC that need to be accommodated (cf. Subsection 4.1). There are two kinds of adaptive RM: the configurative and the generic adaptive RM [21]. Configurative models contain rules on how the model is to be adjusted to a specific situation via configuration parameters, hereby they are mostly easier to use but harder to construct. Generic adaptable models provide a framework but can only be adjusted manually and thereby are easier to construct, however require more effort in use [21]. Since the goal of the RM for DPC is to reduce development efforts and to imprint obtained knowledge into the RM, we have opted for a configurative RM. To guarantee the best possible fit of the generated specific model we have additionally included generic adaptation mechanisms and thereby keep the possibility of altering the specific model after the model configuration.

After the construction, the RM has been demonstrated by (B.2) customizing it for a real world scenario (cf. Subsection 5.1). The final step of reference modeling presented in this paper encompasses the (A.3) evaluation (cf. Subsection 5.2). On the one hand, for the analytical part we have used the Guidelines of Modeling (GoM) [22] and on the other hand, for the empirical part, we have conducted expert interviews. In the second round of the empirical evaluation, the RM has been used in an organizational setting in the ongoing research project PROFUND to build and deploy a cross-organizational DPC for companies without pre-existing DPC.

4 Construction of the RM

In this section, the paper at hand addresses the first two steps of the construction phase of reference modeling. The following literature review defines the problem and analyzes the domain of the RM. By deriving the DPC characteristics, it builds the foundation for the conceptualization in Subsection 4.2. During the conceptualization, the views and adaptation mechanisms are built.

4.1 Deriving the DPC Characteristics from Literature

Although some reviews and studies partly focus on DPC as an IS from a design and development perspective, to our best knowledge the holistic conceptual view for the development of DPC is neglected. Our secondary literature study based on [18, 19] aims to fill this gap. It includes prior literature reviews and studies describing an extensive amount of DPC in order to gain a comprehensive overview (cf. Table 1 for a fact sheet) and addresses the questions of how the DPC is described as an IS, which perspectives exist, and which overall picture emerges. Through the analysis of the papers we have developed ten different concepts in several rounds by axial coding, in order to summarize and aggregate the DPC characteristics. The morphological box is used in accordance with the general morphological analysis [23] to break down the complex properties of DPC into individual concepts and aspects. This creates a clear structure and a solution space that is easy to interpret. The disadvantage of the morphological box for this review - that common or valid combinations of aspects are not recognizable - is disregarded in favor of concept centricity and easy readability of the results. The percentages for each aspect in the morphological box indicate in how many documents of the literature corpus the specific aspect has been described as shown in Table 1.

Table 1. Morphological Box with DPC Characteristics

Concepts	Aspects					
(1) **Purpose of configurator**	Product development (25,93%)	Product configuration & sales (92,59%)	Customer preference formation (25,93)	Data collection & data analysis (33,33%)		
(2) **Product type**	Tangible (74,07%)	Intangible (40,74%)				
(3) **Addressee of configurator**	External (81,48%)	Internal (33,33%)				
(4) **Production concept**	Development- / Engineer-to-Order (29,63%)	Make-to-Order & Assemble-to-Order (55,56%)	Match-to-Order / Locate-to-Order (7,41%)			
(5) **Type of configuration**	Parameter-based & component-based (96,30%)	Requirement-based (55,56%)	CAD (25,93%)			
(6) **Standard functions (GUI)**	Active input (92,59%)	Product information (77,78%)	Visualization (70,37%)	Videos & animations (33,33%)	Sound (3,70%)	
	Price calculation & display (51,85%)	Guided process & navigation (51,85%)	Save & share (29,63%)	Feedback (62,96%)		
(7) **Support functions**	Validation (70,37%)	Recommendation (59,26%)	Basic configuration (51,85%)	Comparison (22,22%)		
	Automated solution generation (44,44%)	Simulation (11,11%)	Prototype & 3D print (11,11%)	Community & expert feedback (40,74%)		
(8) **Knowledge repres. / logic**	Rule-based-reasoning (33,33%)	Case-based-reasoning (25,93%)	Model-based-reasoning (37,04%)			
(9) **Data models / management**	Customer / session data (44,44%)	Configuration rules (62,96%)	Product model (74,07%)	Configuration (81,48%)	User interface data (22,22%)	Administration data (14,81%)
(10) **System design**	Structure (51,85%)	Function (59,26%)	Data / function integration (59,26%)			

Fact sheet - Search Terms: Produktkonfigurator, Konfigurator, Produktkonfiguration, Sales Configurator, Configurator AND Product, Configurator AND Mass Customization, Toolkit AND Mass Customization, Frontend AND Mass Customization, Customer-Company Interaction System. Databases and Search Engines: IEEE Xplore, AIS eLibrary, Wirtschaftsinformatik-Genealogie-Datenbank (wige.net), Web of Science and Google Scholar; searched in titels only. Number of Articles found: 101 initially, 76 relevant after titel-check; 136 additional paper after forward-backward search; total 214, 68 relevant after abstract-check; 27 relevant after full paper check. Definition of relevanc: the DPC is discussed as a IS from a software engeneering / design perspective.

Existing literature addresses various (1) purposes of DPC ranging from product development over sales and customer preference formation to data collection and analysis [8, 24]. The most discussed purpose is product configuration with the intention of selling the product afterwards. The (2) product type a DPC is used for can be tangible, like cars or intangible, like software [8]. The (3) addressee of a DPC ranges from engineer or salesman to customer, including companies as a customer or end customers. To simplify the fact of user group variation, the different users were divided into two groups: external, which means customers or the public, and internal representing employees. DPC for tangible products and those which are used externally are examined most frequently, respectively. The focus of customer integration by DPC lies on Made-to-Order (MTO) and Assemble-to-Order (ATO), but the entire spectrum from Engineer-to-Order (ETO) to Match-to-Order (MCHTO) or Locate-to-Order

(LTO) [25] is represented [26–28], as shown in the (4) production concept. The different production concepts relate to the grade of the customizability of the product ranging from pure customization (ETO) to mass production (MCHTO/LTO). The (5) type of configuration describes, how the product can be altered, e.g. by changing parameters or components or by stating a requirement which presets the parameters and components needed to satisfy this requirement [29]. As our literature corpus indicates, the configuration via parameters and components is by far the most common way to realize product customization with DPC. Many (6) standard and (7) support functions of a DPC, e.g. updating the product visualization or automatic validation [30] could be identified. Some functions such as active input, which means the act of entering, e.g. a configuration parameter, are almost always mentioned but others such as (listening to product) sounds are rarely mentioned. To realize the different customization approaches and to guide the user through the configuration task, DPC need to include the configuration rules and algorithms, which can be multi-tiered and complex [29, 31]. This is addressed by (8) knowledge representation and reasoning for DPC, which typically distinguishes the three categories: rule-based, case-based, and model based. Each approach represents the knowledge in different ways, either by if-then-rules or by case similarity or by using constraints [8]. In addition, various (9) data models and (10) system designs have been discussed in literature, e.g. the distinction of user frontends and technician frontends with their respective corresponding logic [32]. It is notable that researchers discuss the system design concepts less distinctively, which forms a strong contrast in regards to the intensively discussed distinct functions of DPC.

All in all, DPC is an umbrella term encompassing wildly different IS, which only have in common that they are used for product configuration tasks in a broad variety of domains. Due to this, the (cost-) efficient development of DPC remains a challenge, as there is no standard procedure to guide such development. No single perfect DPC design is available because of the variety concerning: the industries they are used in, the products they can be used for, the customer integration point they can be used at, the customization approach they can realize, and even the user they can address. This leads to the conclusion that the RM needs to be specific enough for software developers to guide the development of a specific DPC but at the same time generic enough to be applicable for ETO, MTO, or MCHTO settings in different industries. Therefore, a preferable solution for the aforementioned problem is a configurative RM addressing the variety in this class of IS [cf. 21].

4.2 Building the Views

The Approach. Based on the findings from the literature review the requirements for the RM for DPC were derived and led to some major design decisions. The views of the RM could be determined, since from a development perspective on the results, there is a clear emphasis on DPC functions, architecture, and data models with a strong focus on functionality. Accordingly, the RM needs at least two different views to represent the different DPC aspects: a functional view describing user interaction with the DPC and an architectural view describing the system architecture including the data models in the backend. The literature review indicates that the RM must be adaptable in order to cover the broad field of DPC applications. Figure 1 illustrates how the morphological box was used to build the foundation of the RM. There are three different sections the concepts can be divided into. The first five concepts build the setting options for the modeler represented by a choice board. The standard and support functions can be converted to a functional view and the last three concepts can be converted to an architectural view. The resulting three artifact types are interconnected via RM configuration rules. The components of the artifact are described in detail below.

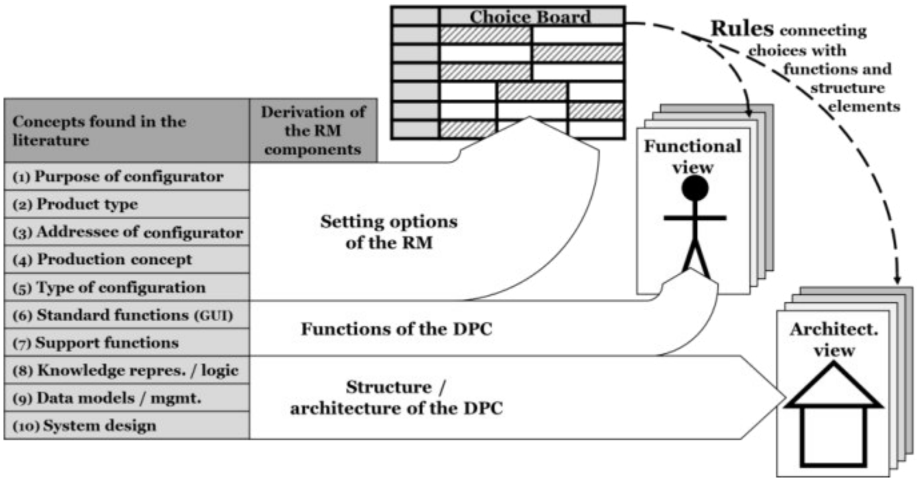

Figure 1. Derivation of the RM Components

Choice Board. What is here called a choice board, is again a morphological box. It is well suited to show configuration dimensions (concepts) of the RM as well as selection options (aspects) in the dimensions. This is a common representation of the multidimensional solution space that illustrates the multitude of variations of RM [33]. There are six questions asked representing the dimensions on the left side of the choice board such as: “Which purpose will the configurator have?”. On the other side, there are predefined answers as selection options like: “Product development” or “Product configuration & sales” (cf. Table 2 in Subsection 5.1). Two kinds of rules were defined in the RM construction. The first one describes the effects of the choice board selections on the two views (cf. rules below) and the second one describes valid combinations in the choice board. It is possible to configure over 270 different versions of the model. An example of a rule is: IF product development THEN NOT Match-to-Order AS production concept. It is suggested to use the choice board by answering the questions in a top-down order, which makes it easier to use.

Functional View and Architectural View. Once the views were identified, we chose adequate modeling languages. In our context, we apply the unified modeling language (UML) Use Case diagram to build the functional view due to two reasons:

First, UML is commonly used in this specific context (describing DPC) and therefore already used in this domain and second, UML is widely accepted and a well-known standard in the field of software engineering [31, 34]. The application of an established modeling standard leads to RM, which are easier to use and avoids starting barriers. Applying a use case diagram has multiple advantages. First, it is possible to build hierarchies of DPC functions like "View product information" as a generalization of "View pictures", "Read description", and so on. Second, the actors included in this diagram type are representing the different DPC users showing the typical interaction with the DPC. During the design process the frequency of mentioning the function in the literature review was taken into account, resulting in mandatory DPC functions in the RM. The full version of the diagram is shown in Figure 2 in Section 5.

For the architectural view an architecture diagram also based on UML was chosen, taking the principle of systematic design into account. This modeling principle demands the inter-model consistency between different views [22]. The condition is fulfilled by using a specified modeling standard such as UML for both views. The architectural view is based on the common three-layer architecture with GUI, application layer, and persistence layer. The view contains components, data elements, and their relations corresponding to the functional view. Both views are linked by a set of rules, so that a selection in the choice board results in changes in both views.

Adaptation Mechanisms and Rules. After determining the representation (modeling language) and the model elements for the views (functions and architectural components), the adaptation mechanisms for the RM were defined. By analyzing the literature, it became clear that dependencies between the DPC functions as well as the architecture and the application field of DPC exist. The RM constitutes a logical representation of these dependencies and therefore becomes a configurator for the DPC as an IS on a conceptual level. The rules for the RM were fully elaborated, resulting in over 100 rules for the configuration and about 40 rules for the generic adaptation. The software developer configures the model by using the choice board. In the first step (the configuration) both views are generated and in the second step (the generic adaptation) individual components are modified if necessary. The result is the recommended functional and architectural scope for a specific DPC. An example of two configuration rules combined is as follows: IF product configuration & sales AS purpose of DPC THEN calculate price information AS function AND price calculation service AS architectural component. Several rules have been defined for each selection option in the choice board. That is, when a selection is made in the choice board, a corresponding set of functions and components is added to the diagrams. In the case that the software developer does not agree with these recommendations, he/she can use the configured model as a starting point and delete or add elements in the model using the generic adaptation rules. Removing, e.g. the price function triggers the generic adaptation rule: IF REMOVE calculate price information AS function THEN REMOVE price calculation service AS architectural component, vice versa. Adding functions and components is limited to the maximum scope of the RM, which is shown in the following section. The demonstration will point out how this RM can be used for a real world scenario.

5 Usage and Evaluation of the RM

In this section, we address the second step of the usage phase and the third step of the construction phase of reference modeling. To perform the customization step of reference modeling, both adaptation variants including the configuration and generic adaptation mechanisms were demonstrated in Subsection 5.1. After this demonstration of the usage, the analytical evaluation to assess the quality and expressiveness of the RM diagrams and the empirical evaluation to assess the suitability and applicability of the RM in real world cases follows in Subsection 5.2.

5.1 Using the RM for DPC

The procedure to use this DPC RM is as follows: First, the DPC developer gathers the initial requirements, second, the DPC developer fills in the choice board with these requirements, third, the RM generates a recommended solution diagram, fourth, the DPC developer uses the generic adaptation to get a better fit of the solution, and finally fifth, the DPC developer uses the specific model to develop a DPC.

For the demonstration and the subsequent evaluation a software development company specialized in DPC development was involved. They described a real case of a prior DPC they had developed for an Austrian manufacturer of customizable front doors. The manufacturer wanted to be able to support its own sales and trading partners in the preparation of quotations. The DPC development company used our choice board to specify the requirements of the manufacturer. To avoid any bias, the developers did not know the rules and logic behind the RM. In Table 2, the selected configuration is shown in the choice board.

Table 2. Choice Board

Dimension / Question	Answer		
Which purpose will the configurator have?	Product development	**Product configuration & sales**	
Who will use the configurator?	**External (customer)**	Internal (employee)	
Which are additional pursued objectives?	**Customer preference formation**	Data collection & data analysis	
Which production concept is pursued?	ETO	**MTO / ATO**	MCHTO / LTO
What type of configuration is needed?	**Parameter / component**	Requirement	CAD
How will the system be integrated?	**Stand-alone**	Data exchange	System integration

The use case diagram shown in Figure 2 includes the maximum scope of functions in the RM, even though not all functions are included in the solution for the manufacturer. This is done for illustration purposes, the legend explains which functions are included and why.

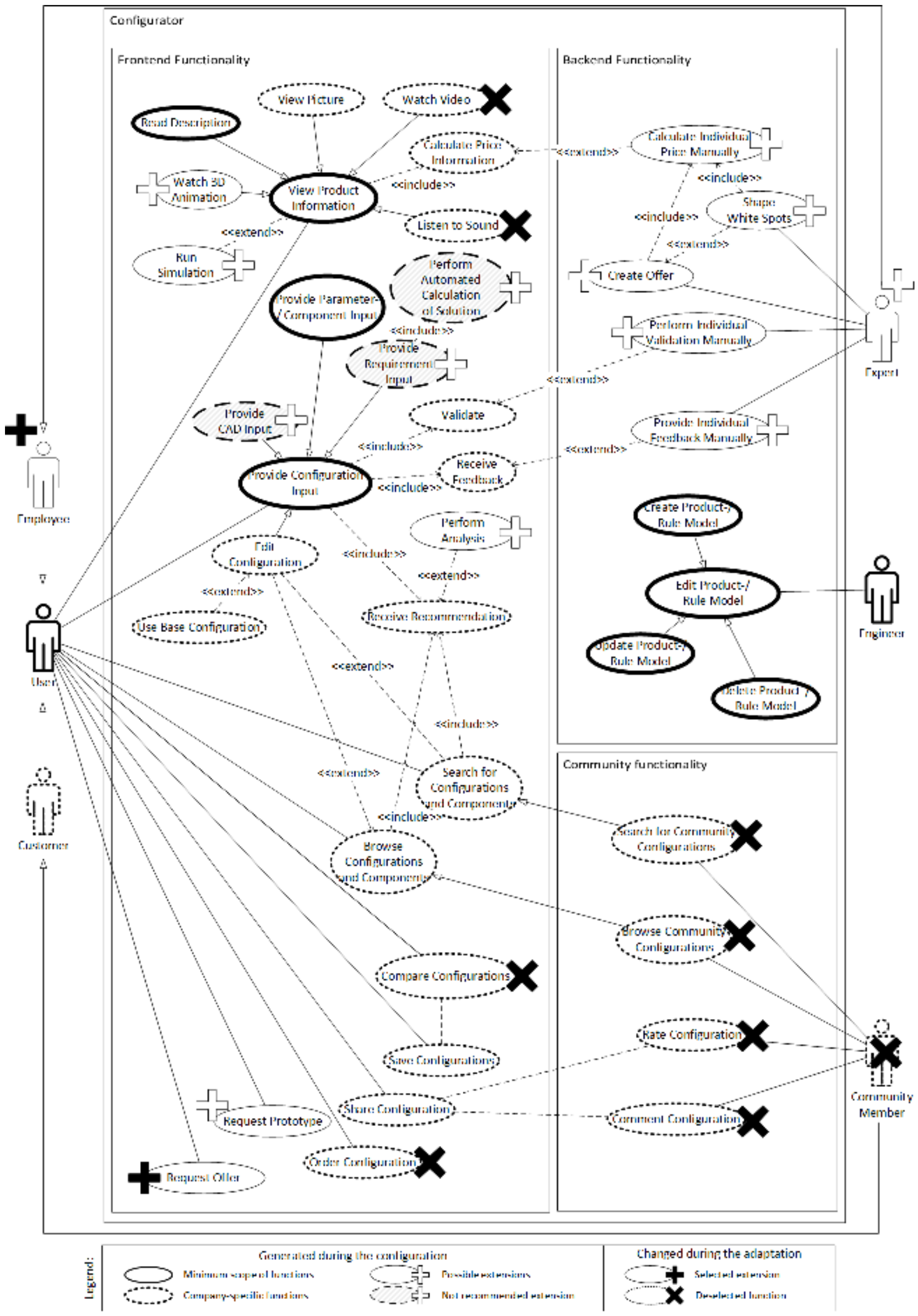

Figure 2. Demonstration of the Functional View

The four symbols on the left side of the legend describe the highlighting of the function explaining the recommendations of the RM after the configuration. The two symbols on the right illustrate the highlighting of the functions, which are added or removed during the general adaptation.

After the model configuration we presented the result to the DPC development company. The developer then had the chance to perform the generic adaptation step to have a closer match to the real DPC of the manufacturer. It is recognizable in Figure 2 that, besides a few other functions, those concerning community integration have been removed whereas the function "request offer" and the employee-actor have been added by the DPC development company. As mentioned before, the architectural view is interconnected with the functional view. Its components correspond with the functions. However, this diagram is not included in this section. Nevertheless, the whole demonstration including both views was performed with the DPC development company and was used for the evaluation afterwards.

5.2 Evaluating the RM for DPC

Analytical Evaluation. For the analytical evaluation, the GoM were used to assess the quality and expressiveness of the RM diagrams. The GoM consist of six principles: (1) construction adequacy, (2) language adequacy, (3) economic efficiency, (4) clarity, (5) systematic design, and (6) comparability [22]. The conclusions of the assessment were the following. (1) and (3) are completely fulfilled, since: first, the problem definition has proven that a RM is a suitable way to address the situation, second, the two views (functional and architectural) are a well-accepted way to describe IS, third, the RM does not include unnecessary content, and fourth, it is considered as economically efficient to build a RM to address a cross-industry problem. The assessment of the latter is based on the fact that in the empirical evaluation the development company estimated an acceleration of DPC development using the RM, which could also be confirmed in the research project PROFUND. Also (4) and (6) were classified as fully satisfied, since UML and UML-related modeling languages were used. However (2) and (5) were rated as partially satisfied, due to the semiformal language and the slightly altered diagram types, although a meta model integration for both views was carried out (though not included in this paper) to guarantee a systematic design.

Empirical Evaluation. At first, the usage perspective was evaluated with the question how suitable the choice board is and whether it is sufficient for the elicitation of requirements in the first stage of DPC development. This was realized by conducting an expert interview with a developer and a requirements engineer in the DPC development company mentioned above. The experts have considered the approach as promising and suitable, but a refinement of the questions and answers is desirable in order to obtain a higher degree of detail in the initial requirements engineering phase. They also identified three major shortcomings of the choice board: First, the external use of the DPC is not divided into the categories of business-to-business and business-to-customer, which can have implications for the frontend design or user handling. Second, the dimension / question for the addressed customization approaches is missing (fit, style, etc.), and third, the rules in the choice board were too strict. While the ruleloosening was realized during development process iterations, the other issues need to be addressed by a broader approach in the RM including, e.g. design suggestions for the frontend or the user handling, e.g. a user login in case of a business-to-business DPC.

At second, the result after using the RM in a real world scenario was assessed. Since the RM was applied to configure a specific model for an already implemented DPC, the resulting model could be compared with the functionality and the architecture of the real system backend of the Austrian front door manufacturer. The comparison was carried out by one of the developers and not by us. The feedback exhibited that a function in the functional view and respectively a component in the architectural view concerning "requesting offers" were missing. As it can be seen in Figure 2, the missing parts were added in the prior RM refinements and are already included in the paper at hand.

At third, the RM was used in the research project PROFUND. The RM was used to accelerate the requirements engineering and the conceptual design of the information technology landscape in the project including a DPC. This led to a prompt conceptual model of the DPC, which was generated by the RM and was used as a foundation for discussion. However, the usage has also shown that it is not facilitating the solution space development and knowledge representation of DPC so far. The usage of the RM in this context will lead to a broader evaluation concentrating on distributed configuration in value chains in an ETO environment and including the whole process of software development. Feedback rounds with to the suggested functionality of the DPC showed that searching for parts, configuring as well as saving the configuration related to the customer data were seen as the most important functionalities, whereby generating offers or sending messages to the customer were deemed irrelevant by the textile companies. The first prototype of the DPC based on the RM has been developed and first feedback indicates that the prototype constitutes a promising approach.

6 Discussion

This paper presents a RM for DPC, which consolidates prior research findings and integrates them into the views, the choice board, as well as the rules and thus forms a novel overarching artifact. The recommendations included in the RM are derived from a comprehensive sample of individual cases and therefore offer general validity. We addressed the missing holistic approach by consolidating prior research viewpoints on DPC such as functionality, architecture, usage, and point of customer integration. While studies like [10] examine a specific type of DPC in a comprehensive way, this paper specifically targets the wide range of DPC to identify the underlying structure and mechanisms of this IS type. Thereby, the presented approach of reference modeling is an example on how to deduce specific features of a RM via a systematic literature review in a mature research field, instead of abstracting from specific models in an inductive way [35].

The evaluation results indicate that the RM is able to accommodate real world requirements successfully and that the RM is applicable across different industries, such as the discrete manufacturing industry (e.g. front doors) as well as the process industry (e.g. technical textiles). The application of the RM in a software development project for small and medium sized enterprises in the textile industry has confirmed the potential benefits of the RM, but has also revealed its limitations. The current RM only allows manual deriving of a specific model. Instead, the model could be automatically generated by a software tool with the option to analyze the usage of the RM for subsequent optimization. In addition, the current RM provides abstract recommendations and no specific support for the actual design and implementation.

Further development could break the RM down to more specific models in order to support the development efforts not only in the initial stages, but also in the subsequent development stages. This purpose could be achieved by, e.g. refining the modules with class diagrams.

7 Conclusion

The paper at hand aims to support organizations in the development of DPC. In a first step we have deduced the characteristics and functions of DPC by conducting a systematic literature review. Considering the wide variety of DPC applications, we have made the design decision that the RM must be adaptive to ensure universality but also specificity. As shown in the demonstration, the adaptive RM constitutes a tool, which can support the development of DPC starting with the requirements engineering stage. The RQ on how to build a DPC RM in terms of design decisions and scope to encompass existing knowledge was answered successfully. The resulting RM is designed to address the problem of high expenses related to the development of DPC, since recommendations for the functional scope and the architecture of the DPC are provided at a very early stage in the development process.

In summary, our research contributes to the knowledge base by applying reference modeling to the field of DPC by making knowledge from research and practice available and thereby usable for developers. Thus, it provides a first building block to the endeavor of facilitating DPC development.

Acknowledgement: *The research described in this paper was supported by a grant from the German Ministry for Research and Education (BMBF), project name: PROFUND grant number 03ZZ0618C.*

References

1. Pine, J., Victor, B., Boynton, A.C.: Making Mass Customization Work. Harv. Bus. Rev. 71, 108–116 (1993)
2. Fogliatto, F., da Silveira, G., Borenstein, D.: The mass customization decade: An updated review of the literature. Int. J. Prod. Econ. 138, 14–25 (2012)
3. Gandhi, A., Magar, C., Roberts, R.: How technology can drive the next wave of mass customization. (2013)

4. Zhang, L.L., Helo, P.T.: Empirical investigation on implications of configurator applications for mass customization. In: Modrak, V. (ed.) Mass Customized Manufacturing: Theoretical Concepts and Practical Approaches. pp. 217–233. Taylor & Francis, London (2016)
5. Franke, N., Piller, F.: Key research issues in user interaction with user toolkits in a mass customisation system. Int. J. Technol. Manag. 26, 578–599 (2003)
6. Abbasi, E.K., Hubaux, A., Acher, M., Boucher, Q., Heymans, P.: The Anatomy of a Sales Configurator: An Empirical Study of 111 Cases. In: Proceedings of the Advanced Information Systems Engineering Conference (CAiSE'2013). pp. 162–177. , Valencia (2013)
7. Forza, C., Salvador, F.: Product configuration and inter-firm co-ordination: An innovative solution from a small manufacturing enterprise. Comput. Ind. 49, 37–46 (2002)
8. Blecker, T., Abdelkafi, N., Kreutler, G., Friedrich, G.: Product Configuration Systems: State of the Art, Conceptualization and Extensions. In: Proceedings of the Maghrebian Conference on Software Engineering and Artificial Intelligence (MCSEAI'2004). pp. 25–36. , Tunis (2004)
9. Zhang, L.L.: Product configuration: a review of the state-of-the-art and future research. Int. J. Prod. Res. 52, 6381–6398 (2014)
10. Walcher, D., Piller, F.: The Customization 500: An International Benchmark Study on Mass Customization and Personalization in Consumer E-commerce. Lulu Marketplace (2011)
11. Rosemann, M., van der Aalst, W.M.P.: A configurable reference modelling language. Inf. Syst. 32, 1–23 (2007)
12. Mittal, S., Frayman, F.: Towards a Generic Model of Configuraton Tasks. In: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'1989). pp. 1395–1401. , Detroit (1989)
13. Helo, P.T.: Product configuration analysis with design structure matrix. Ind. Manag. Data Syst. 106, 997–1011 (2006)
14. McDermott, J.: R1: A rule-based configurer of computer systems. Artif. Intell. 19, 39–88 (1982)
15. Fettke, P., Loos, P.: Referenzmodellierungsforschung. Wirtschaftsinformatik. 46, 331–340 (2004)
16. Fettke, P., Loos, P.: Multiperspective Evaluation of Reference Models – Towards a Framework. In: Jeusfeld, M.A. and Pastor, Ó. (eds.) Conceptual Modeling for Novel Application Domains: ER 2003, Lecture Notes in Computer Science (LNCS). pp. 80–91. Springer, Berlin (2003)
17. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Q. 28, 75–105 (2004)
18. Webster, J., Watson, R.T.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. MIS Q. 26, xiii–xxiii (2002)
19. Fettke, P.: State-of-the-Art des State-of-the-Art: Eine Untersuchung der Forschungsmethode "Review" innerhalb der Wirtschaftsinformatik. Wirtschaftsinformatik. 48, 257–266 (2006)
20. Kitchenham, B., Charters, S.: Guidelines for performing Systematic Literature reviews in Software Engineering. (2007)
21. Becker, J., Delfmann, P., Knackstedt, R.: Adaptive reference modeling: Integrating configurative and generic adaptation techniques for information models. In: Becker, J. and Delfmann, P. (eds.) Reference Modeling Efficient Information Systems Design Through Reuse of Information Models. pp. 27–58. Physica, Heidelberg (2007)
22. Schütte, R., Rotthowe, T.: The Guidelines of Modeling - an approach to enhance the quality in information models Introduction : Theoretical Aspects in Information Modeling. In:

Proceedings of the International Conference on Conceptual Modeling (ER'1998). pp. 240–254. , Singapore (1998)
23. Zwicky, F.: Discovery, invention, research through the morphological approach. MacMillian, New York (1969)
24. Redlein, A.: Fulfilling customer's needs by the use of a variant configurator for dynamic product definition. In: Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'1999). pp. 735–742. , Barcelona (1999)
25. Piller, F., Möslein, K., Stotko, C.: Does mass customization pay? An economic approach to evaluate customer integration. Prod. Plan. Control. 15, 435–444 (2004)
26. Kristianto, Y., Helo, P., Jiao, R.J.: A system level product configurator for engineer-to-order supply chains. Comput. Ind. 72, 82–91 (2015)
27. von Hippel, E., Katz, R.: Shifting Innovation to Users via Toolkits. Manage. Sci. 48, 821–833 (2002)
28. Forza, C., Salvador, F.: Product Information Management for Mass Customization - Connecting Customer, Front-office and Back-office for Fast and Efficient Customization. Palgrave Macmillan, Houndmills (2006)
29. Randall, T., Terwiesch, C., Ulrich, K.T.: Principles for User Design of Customized Products. Calif. Manage. Rev. 47, 68–85 (2005)
30. Füller, K., Böhm, M., Krcmar, H.: Designing for positive user experience in product design: A qualitative analysis of toolkit design elements and their implications on emotional reactions and perceptions. In: Proceedings of the Hawaii International Conference on System Sciences (HICSS'2016). pp. 1810–1819. , Koloa (2016)
31. Roth, M., Nerb, A., Kasperek, D., Lindemann, U.: A Tool to Bridge the Gap from Functional Dependencies to Configuration Rules - Translating Knowledge on Functional Restrictions from Systems Engineers to a Configurator for Sales and Decision Makers. In: Proceedings of the IEEE International Systems Conference (SysCon'2015). pp. 158–163. , Vancouver (2015)
32. Tiihonen, J.: Support for configuration of physical products and services. Aalto University | Doctoral Dissertations, Helsinki (2014)
33. Winter, R.: BEN-Gestaltungsmethodik für Geschäftslösungen. In: Österle, H., Winter, R., and Brenner, W. (eds.) Business Engineering Navigator - Gestaltung und Analyse von Geschäftslösungen "Business-to-IT." pp. 55–84. Springer, Berlin (2011)
34. Yip, A.L.K., Corney, J.R., Jagadeesan, A.P., Qin, Y.: A Product Configurator for Cloud Manufacturing. In: Proceedings of the ASME 2013 International Manufacturing Science and Engineering Conference (MSEC'2013). pp. 1–9. , Madison (2013)
35. Rehse, J.-R., Hake, P., Fettke, P., Loos, P.: Inductive Reference Model Development: Recent Results and Current Challenges. In: INFORMATIK 2016. pp. 739–752. , Klagenfurth (2016)