

Application of design thinking to product-configuration projects

Shafiee, Sara; Haug, Anders; Shafiee Kristensen, Saeedeh; Hvam, Lars

Published in:
Journal of Manufacturing Technology Management

DOI:
10.1108/JMTM-04-2020-0137

Publication date:
2020

Document version:
Accepted manuscript

Citation for pulished version (APA):
Shafiee, S., Haug, A., Shafiee Kristensen, S., & Hvam, L. (2020). Application of design thinking to product-configuration projects. *Journal of Manufacturing Technology Management*, 32(1), 219-241.
<https://doi.org/10.1108/JMTM-04-2020-0137>

Go to publication entry in University of Southern Denmark's Research Portal

Terms of use

This work is brought to you by the University of Southern Denmark.
Unless otherwise specified it has been shared according to the terms for self-archiving.
If no other license is stated, these terms apply:

- You may download this work for personal use only.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying this open access version

If you believe that this document breaches copyright please contact us providing details and we will investigate your claim.
Please direct all enquiries to puresupport@bib.sdu.dk

Application of Design Thinking to Product-Configuration Projects

Abstract

Purpose: Product configurators are expert systems that support product customization by defining how predefined entities and their properties may be combined. Developers of configuration systems act as designers, although they do not often recognize that they are performing as such. Moreover, exploring solution spaces is typically not integral to configuration projects, as this task is typically perceived as mapping existing knowledge to the configurator. This article argues that developing configurators may be understood by distinguishing between the problem and solution spaces using design thinking (DT).

Design/methodology/approach: A multiple-case-study approach with four configuration projects is adopted to study two projects involving DT and compare them to two similar projects not involving DT. Data collection depended on multiple data sources via workshops and semi-structured interviews.

Findings: First, DT methods and concept–knowledge (C-K) theory are integrated into configuration projects. Second, the application of DT during configurator development is presented through workshops and interviews, which demonstrates the benefits of DT in overcoming existing challenges.

Research limitations/implications: The case studies demonstrate the successful implementation of DT in developing configurators. However, a limited number of cases in only one company limits the generalizability of the results.

Practical implications: The framework's individual steps create a structured approach to supporting industrial companies with a toolbox of DT techniques and methods for configuration projects.

Originality/value: The results show that the application of DT to configuration projects can improve user motivation, stakeholder satisfaction, and knowledge acquisition.

Keywords: product configuration, configurator, design thinking, knowledge management, problem, and solution space

Paper type: Research paper

1. Introduction

Product configurators have been attracting attention in different industries because of their support of product specification in sales and engineering (Haug et al., 2019a). Widely used in various industries, configurators have substantial benefits, such as shorter lead times for generating quotations, fewer errors, increased ability to meet customers' product-functionality requirements, fewer resources, optimized product designs, less routine work, and improved on-time delivery (Petersen, 2007; Salvador and Forza, 2004). Based on the literature, configuration projects face various challenges, such as stakeholder communication and collaboration (Felfernig et al., 2014; Haug et al., 2019b; Zhang, 2014), and companies face challenges developing implementing configuration projects (Kristjansdottir et al., 2018; Shafiee, 2017; Zhang, 2014) such as: resource management (Shafiee et al., 2014), product complexity (Forza et al., 2006; Haug, 2013; Heiskala et al., 2007), technical IT (Heiskala et al., 2007; Shafiee et al., 2017), knowledge acquisition (Heiskala et al., 2007; Shafiee et al., 2018; Zhang et al., 2020), and organisation (Haug, 2013; Heiskala et al., 2007).

The knowledge required for configuration projects, a highly reported challenge, is normally specialized product knowledge beyond the configuration team's expertise (Haug, 2008; Nonaka, 1994; Studer et al., 1998); some is not even articulable but only tacit. In this context, we lean upon the work of Polanyi (Polanyi, 1969; Polanyi and Nye, 2015), who argues that all knowledge has a tacit dimension. "Tacit knowledge" should not be confused with "knowledge not yet articulated" (Tsoukas, 2003). The knowledge-acquisition literature reflects this distinction; knowledge elicitation is typically perceived as a modeling activity (Speel et al., 2001) in which a configuration team acting as knowledge elicitors and domain experts work together to create a model of an expert's knowledge including its tacit dimension.

Knowledge formalization and communication in configuration projects can be perceived as a modeling activity in which product knowledge is described on a relatively visual, abstract level to be understandable to all persons concerned (Shafiee et al., 2017). The knowledge in configuration projects is extensive and must be continually validated by domain experts (Basili and Weiss, 1984). Strong communication between the configuration team and domain experts in configuration projects is vital, and specific modeling techniques tend to accomplish this challenge (Forza and Salvador, 2002). Developing and implementing a configurator is a collective task that involves internal stakeholders from various divisions within the organization. Hence, the social dimension of design relies on stakeholders understanding the system, research and development, marketing, IT, and domain experts interacting, and assembling multidisciplinary teams.

Design may be understood as a mapping process between functions and design parameters (Yoshikawa, 1981). Design problems are everywhere; for example, what does a backache-reducing office chair look like? What form should a computer interface have to be accessible for elderly people (Lindberg et al., 2011)? A recent design theory, concept-knowledge (C-K) theory, reflects the assumption that design can be modeled as the interplay between two interdependent spaces with different structures and logics: the spaces of concepts (C) and knowledge (K) (Hatchuel and Weil, 2009; Sharif Ullah et al., 2012). The propositions of the C-space focus on objects whose existence is still undecidable on the basis of the propositions available in the K-space (Le Masson et al., 2017). C-K theory proposes as unified a language as possible to facilitate dialogue between the major design professions—designers, engineers, and architects—independently of the specific objects they design and handle (Le Masson et al., 2017). C-K theory accounts for creative design, models the design process using existing knowledge, expands knowledge bases, and explores unknown concepts, so it extends design thinking (DT) to further support innovative design education (Hatchuel et al., 2008).

DT is strongly connected to exploration and learning (Beckman and Barry, 2007) and is mainly used in the fuzzy front end of innovation and product development (e.g., Lindberg et al., 2010). Although DT was at first only explored and developed by professional designers, strategies have been identified that are relevant to all disciplines and professions. According to Braha and Reich (2003), the design process is characterized by being iterative, exploratory, and sometimes chaotic. It starts with abstract specifications, what Hatchuel and Weil (2009, p. 182) call a "brief," and it ends with a product description while gradually refining the product specifications. DT has received attention in product and service design as the major component of business competitiveness; many companies have committed themselves to becoming design leaders (Dunne and Martin, 2006; Razzouk and Shute, 2012). Moreover, DT has become an integral part of the design and engineering fields as well as business because it involves creative thinking to solve problems.

Despite the broad range of DT applications, it has yet to be applied to product configuration. Given that a lack of user acceptance can frustrate configuration projects (Haug et al., 2019b), the integration of DT methods appears relevant, since such methods have been shown to strengthen user involvement. Furthermore, as DT methods allow deeper user understanding, DT integrating should increase the system quality. Thus, in this study, we explore the application of DT to broaden problem understanding and problem solving in developing configuration projects.

Comprehensively understanding what solutions will work or not and how the interactions between users and software can be shaped generally depends on communication with domain experts (Lindberg et al., 2011). Nowadays, IT engineers develop for highly competitive consumer markets in which successful innovations are

defined by users' points of view rather than technical perfection. The scant literature on the application of DT to IT projects has nevertheless demonstrated that DT has practical benefits for both waterfall-based and agile-based approaches (Lindberg et al., 2012). Proponents of DT claim it is an all-purpose problem-solving approach, a creativity booster for organizations and their employees, and a way to be more innovative and produce groundbreaking ideas or install a (managerial) mindset that promotes better balanced analytical and creative thinking in the organization (Brown, 2008; Dunne and Martin, 2006; Seidel and Fixson, 2013).

Few empirical studies have researched the actual use of DT in companies, so the understanding of DT's potential benefits for organizations is limited (Carlgren et al., 2014). As mentioned above, configuration projects have several main challenges: knowledge acquisition, communication, and organization (Kristjansdottir et al., 2018). DT can address these challenges (Shafiee et al., 2019). Moreover, configuration development requires highly trained professionals for complex technical issues, such as programming languages or software and hardware architecture. Moreover, many features from users' perspectives must be considered. This paper argues that the analysis and design phase of product-configuration projects may be understood by distinguishing between the problem and solution spaces in DT. Exploring the solution space is not typically integral to configuration projects, which normally map existing knowledge to the configurator. The solutions and frameworks that support configuration projects are dominated by a technical perspective; however, such challenges seem more social and organizational. According to Lindberg et al. (2011), the days when IT-based products pushed technological dynamics are gone; now, they are deeply dependent on social dynamics and team-based collaboration as a core feature of agile development.

The application of DT to configuration projects requires a complementary thinking style that extends the collaboration and communication abilities of domain experts and configuration teams to make their outcomes more innovative. Verbal and written communication alone do not suffice to support scoping, dialogue, and knowledge acquisition in these projects; users have requirements that should be considered during configurator development. DT provides tools such as journey mapping, sketches, prototypes, diagrams, and models as visual and tactile representations of eliciting, communicating, evaluating, and acquiring knowledge and ideas. This paper focuses on relating DT to the structures, cultures, and development of configuration projects, particularly its social aspects. The main research question is the following:

How can the principles of DT support the development of product configurators?

The remainder of the paper is structured as follows. First, the paper reviews the DT literature and discusses its potential applications in configuration projects. Second, the paper studies two configuration projects in which DT was applied. Next, these are compared to two similar cases that did not involve DT. Finally, conclusions are drawn.

2. Literature review

2.1. Design thinking for IT development

DT can solve complex problems by understanding customers' needs, redefining the problem in a human-centric manner, creating new ideas, and adopting a hands-on approach in prototyping and testing. A DT team combines insights from interviews, represents a target group for the innovation team, ideates possible solutions, and then prototypes them. To initiate ideation, a team brainstorms a so-called "How Might We" questions, which can be derived from the team's "Point of View" (Vetterli et al., 2013). Each prototype undergoes tests with target users, which yields information that must be synthesized again. Depending on the outcome of this synthesis phase, the team can start iterating to refine the idea or return to the understand and observe phases to answer open questions and investigate new aspects of the problem (Dorst, 2006; Meinel and Leifer, 2015; Vetterli et al., 2013). DT thus has three basic characteristics (Lindberg et al., 2011):

1. Exploring the problem space: DT acquires an intuitive (not fully verbalized) understanding mainly by observing exemplary use cases or scenarios, not by formulating general hypotheses or theories regarding the problem and synthesizing this knowledge for different stakeholders and their points of view.
2. Exploring the solution space: DT seeks many alternative ideas in parallel and elaborates on them by sketching and prototyping, while ideas being consciously transformed into tangible representatives.
3. Iteratively aligning both spaces: These tangible representatives of ideas facilitate communication not only within the design team but also with users, clients, and experts. Thus, DT helps to stay connected with the problem-relevant environment and can use this information to refine the chosen solution path(s).

Figure 1 illustrates these three processes.

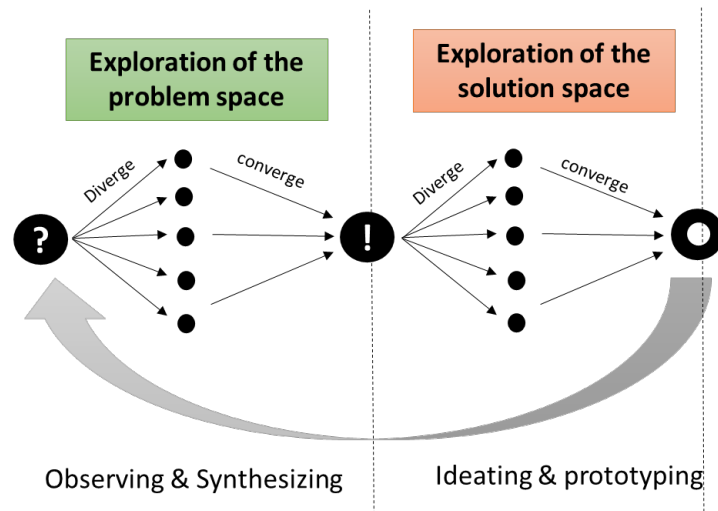


Figure 1. Exploration of problem and solution spaces (Lindberg et al., 2011)

2.2. The potential of DT in configurator projects

As mentioned above, the literature has reported the significant benefits of product configurators; however, such projects often fail (Blazek and Pilsle, 2017; Haug et al., 2019b; Walcher and Werger, 2012). Many features from users' perspectives must be considered during the scoping and development phases of configuration projects, and neglecting or misinterpreting these features can lead to such failures (Shafiee et al., 2020). An isolated technical perspective relying on analytical thinking can thus create an innovation trap of spending too much effort on developing technically novel solutions of which clients cannot see the value (Lindberg et al., 2011; Song et al., 2012). Applying DT to IT development requires a complementary thinking style that extends the problem-solving abilities of IT-development teams to make their outcomes more innovative (Lindberg et al., 2011).

To understand how DT can contribute to configurator projects, Shafiee et al.'s (2018) iterative framework of scoping and managing knowledge in configuration projects serves as a basis. Their knowledge-management (KM) framework comprises four steps: (1) determining the scope of the system to establish the project goal based on stakeholders' requirements and prioritizing the required functionalities; (2) acquiring knowledge to group it according to the desired output and identify knowledge sources; (3) modeling and validating knowledge; and (4) documenting and maintaining knowledge to ensure the KM system can be updated.

Several articles have discussed the challenges of configuration projects (e.g. Forza and Salvador, 2002; Heiskala et al., 2007). Kristjansdottir et al. (2018) summarized the configuration development challenges (Table 1), which were used to interview the IT team at the case company to measure the DT influence on configurator challenges.

Table 1. Challenges reported for configurator-development projects (Kristjansdottir et al., 2018)

Categories of configurator-management challenges	Specific challenges
Resource constraints	<ul style="list-style-type: none"> • Lack of resources • Vulnerability if key personnel leave
Product-related challenges	<ul style="list-style-type: none"> • Complexity of product structures • Continuous change in product offerings
IT (technical) challenges	<ul style="list-style-type: none"> • Software development • Systems design for user-friendliness
Knowledge-acquisition challenges	<ul style="list-style-type: none"> • Difficulty acquiring the correct knowledge • Lack of requisite knowledge to meet users' and customers' needs • Failure to communicate knowledge in the maintenance phase
Product modeling (knowledge representation) challenges	<ul style="list-style-type: none"> • Complexity due to a lack of product range overview • Correctness of specifications generated by the configurator according to the product model • Lack of knowledge related to product modeling
Organizational challenges	<ul style="list-style-type: none"> • Lack of support from top management • Resistance to using the configurator • Disagreements about the scope of the configurator

The literature primarily presents DT as an approach to develop more creative ideas or more empathetic designs to meet customer needs (Brown, 2008; Carlgren et al., 2014; Dunne and Martin, 2006). Much perceived value and many effects of DT indeed reflect user-centeredness and co-creation with users (Wang et al., 2020); however, DT brings something more than a human-centered method: prototyping, an initially difficult but surprisingly useful aspect of design (Carlgren et al., 2014). Prototyping is not only a way to learn and expand ideas but also to communicate with decision-makers and new team members. The material and visual practices commonly used by designers are the results of DT (Ravasi and Stigliani, 2012).

Carlgren et al. (2014) argued that the use of DT can increase long-term innovation in a company by contributing to the three dimensions of innovation capabilities that govern decision-making in organizations: resources, processes, and mindsets. We use the same categories to frame our interview questions for stakeholders to investigate DT influence.

3. Conceptual framework for using DT in configuration projects

3.1. DT in configuration projects

DT is a human-centered approach that includes the perspectives of many stakeholders, both internal (within the team and firm) and external, and it aims to enhance the human experience and solve complicated problems (Kolko, 2011). According to Lockwood (2010), the focus of DT is “observation, collaboration, fast learning, visualization of ideas, rapid concept prototyping, and concurrent business analysis.” The distinction between the problem and solution spaces elucidates the dualistic approach of DT, which systematically and iteratively seeks novel solutions for social and/or technical systems (Lindberg et al., 2011). Therefore, it may contribute to solving KM-related challenges in configuration projects, which are mainly related to intra-organizational communication.

Space C contains “concepts,” which are undecidable propositions in K (neither true nor false in K) about partially unknown objects x. A proposition is qualified as “undecidable” relative to the content of a space K if it cannot be proven true or false in K (Hatchuel and Weil, 2009; Le Masson et al., 2017). In other words, in C-K, the generative model is a sequence of operators and the conceptual model a set of knowledge items, so the theory illuminates a profound difference between the two (Hatchuel and Weil, 2009; Le Masson et al., 2017). Moreover, C-K theory takes design beyond the scope of classical abduction-driven logical inference.

The term “software design” is, in fact, one of the few design terms almost exclusively associated with technical issues (Lindberg et al., 2011). Configuration-project development can be defined as achieving a comprehensive understanding of (1) what the product will look like, (2) if solutions will work or not, and (3) how the conditions of interaction between user and software can be shaped presupposing the ability to communicate about these questions in technical terms (Lindberg et al., 2011). One basic problem, for instance, is that functionalities and user interfaces, albeit technically perfect, may be incomprehensible or inappropriate from the user’s point of view (Lindberg et al., 2011). An isolated technical perspective entailing isolated analytical thinking can thus generate an innovation trap of spending too much effort on the development of technically novel solutions of which clients cannot see the value (Lindberg et al., 2011; Song et al., 2012). Applying DT to IT development requires a complementary thinking style that extends the problem-solving abilities of IT-development teams to make their outcomes more innovative (Lindberg et al., 2011).

As stated above, this paper aims to integrate DT into configuration projects. With a basis in the DT perspective illustrated in Figure 1, a similar logic may be used to illustrate configuration projects, as shown in Figure 2. The basis of the process is an empty configurator shell, that is, configurator software in which product knowledge has not yet been modeled. In this configurator shell, configurator developers define parts, modules, attributes, and their relationships, thereby constructing the solution space of the configurator. When the configurator is put into operation, users make product choices inside this solution space until they arrive at the customized product of their choice, that is, a particular product instance within this space.

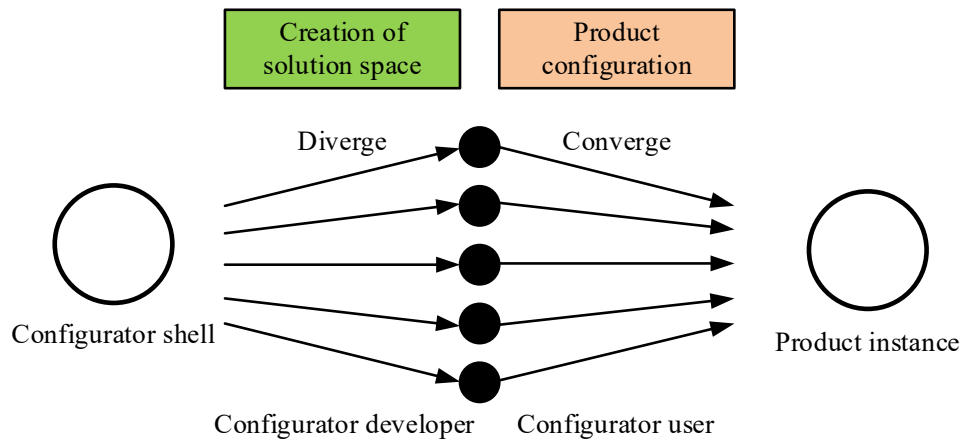


Figure 2. Divergence and convergence in configurator projects

In configuration projects, regardless of the management methods used (waterfall, Scrum, etc.), the configuration team seeks to limit divergent thinking to control the process and ensure progress, so they often limit solutions to previous ones instead of considering new ideas. In this paper, we introduce DT to focus on the expert's world and integrate perspectives on problem understanding and solution finding. The idea behind applying DT to configuration projects is to establish a complementary, meta-disciplinary thinking style. One reason DT works is the multiple iterations in its initial stages, where the costs are much lower, focusing the project scope on feedback in each early iteration from all involved stakeholders.

As previously argued, DT elements in configuration projects may address challenges, as illustrated in Figure 3, which demonstrates the merger of the configuration framework and the DT method.

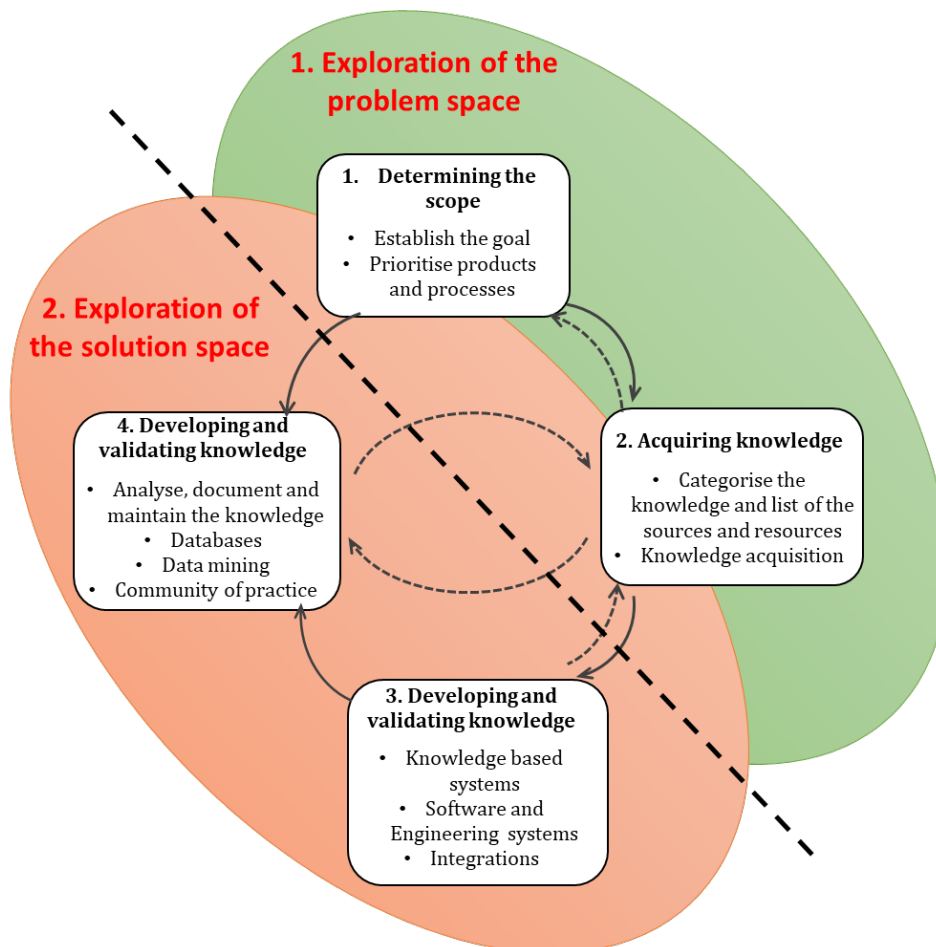


Figure 3. Applying design thinking to knowledge management in configuration projects (after Lindberg et al., 2011; Shafiee et al., 2018)

One step deeper, DT methods can be integrated into configuration projects. Table 2 illustrates this, showing DT phases relative to configuration projects, summarizing relevant design research, and suggesting tools and methods for each stage. Column 1 presents the DT phases in the literature—exploring the problem and solution spaces and iterating to align both—while column 2 contains the KM phases in Figure 3 relative to different DT phases. Column 3 has research design propositions and expectations for each phase and step. Finally, the last column highlights the suggested tools and methods for different phases. The procedure described in Table 2 represents a novel approach to product-configuration projects that requires further investigation by future research. Because of the emphasis on exploring problem and solution spaces, one expected outcome of the application of this process is configurators corresponding with the worldviews of their users and consequently seeing higher success rates.

Table 2. Applying design thinking methods to configuration projects

Design thinking phases	KM in configuration projects	Design research	Suggested methods and tools
<i>Exploring the problem space</i>	Step 1: Determining knowledge scope	Seek to understand culture, experiences, emotions, thoughts, and behavior: What are the user experiences? What will make them happy in achieving goals?	Applying qualitative research methods such as ethnography, participant observation, interviews, journey mapping, job-to-be-done analysis, an mind mapping to draw insights from ethnographic data and to create a “common mind” across team members, using collaborative ideation such as brainstorming and concept-development techniques
	Step 2: Acquiring knowledge	Primarily through semi-structured conversations between the researcher and the project team and definitions of integrations, architecture, and user interface	
<i>Exploring the solution space</i>	Step 3: Developing and validating knowledge	Represents samples qualitatively and seeks profiles of extreme users, as unusual observations may lead to novel ideas	Identifying assumptions: assumptions around value creation, execution, scalability, and defensibility that underlie the attractiveness of a new idea Prototyping: techniques that facilitate making abstract ideas tangible (storyboarding, user scenarios, metaphor, experience journeys, business concept illustrations, etc.)
	Step 4: Developing and validating knowledge	Investigation of behavior, objects, and words that people use to express the way they interact with the system	
<i>Iterative alignment of both spaces</i>		Exploring people’s opinions and behaviors regarding current situations or expectations of future contexts	Applying field experiments: testing the key underlying, value-generating assumptions of a hypothesis in the field with stakeholders

3.2. Measuring motivation

The framework constructed by DT can be further extended with C-K theory, which provides a means to monitor user motivation during a configuration project, which is a foundation for adjusting the level of user involvement throughout the project. A design process in C-K theory is driven by a motivation that can be quantified by information content (entropy) measured under epistemic uncertainty (Sharif Ullah et al., 2012). We assume an expandable knowledge space K, which contains true propositions characterizing partly known objects and partly known relations between these objects. In K, all propositions are true or false. The challenge of seeking new knowledge acts as the other motivation is hereinafter referred to as epistemic challenge. A design process in C-K theory is driven by a motivation that can be quantified by information content (entropy) measured under epistemic uncertainty (Sharif Ullah et al., 2012).

In this context, we consider two concepts: an ordinary concept C1 (developing platform-based configurator or designer-driven development) and a creative concept C2 (designing the configurator based on stakeholders’ ideas or user-driven development). Motivation drives the design process toward conceiving X (an undecided

solution, as in Figure 4). Given the knowledge in the current KM framework (K1) and DT and the KM framework (K2), perhaps C1 is suitable only for automation and not for innovation in developing configurators to accelerate the popularity of the system. The motivation behind pursuing C2 instead of C1 is innovation in design and development as a *compelling reason*, which leads to stakeholder satisfaction and faster, easier configurator implementation. However, the performance of C1 can be determined using the existing knowledge of the KM framework (K3), whereas the performance of C2 is not known; it is the *epistemic challenge*. This paper explores the performance of the new framework of DT and KM (K4).

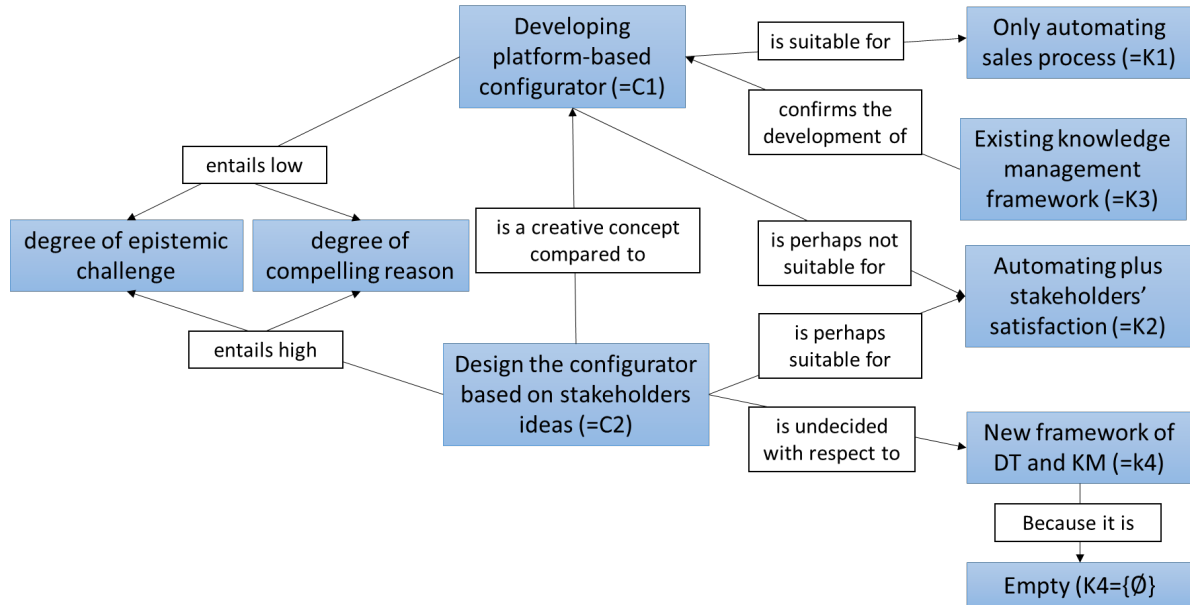


Figure 4. A concept map of a creative concept based on C-K theory (Sharif Ullah et al., 2012)

Two pieces of information are necessary to measure such information content: (1) system range and (2) design range (Sharif Ullah et al., 2012). In practice, the design range is expressed as a crisp range, and the system range is expressed as a probability distribution. Ullah (2005a, 2005b) has investigated calculating the information content of a design with an abundance or lack of knowledge and we use the same approach.

To measure the information content of C1 with respect to the C-K map in Figure 4 in terms of certainty entropy (CE) and requirement entropy (RE), two sets of propositions, {P11, P12} and {P13, P14}, are considered (Table 3) (Ullah and Harib, 2008). The linguistic truth values can be expressed by membership functions of fuzzy numbers and are the following: T1 = “mostly false (0.1),” T2 = “perhaps false (0.27),” T3 = “not sure (0.5),” T4 = “perhaps true (0.75),” and T5 = “mostly true (0.9). These are calculated as follows:

$$TV_{Ti} = \frac{\int_0^1 \mu_{Ti}(x) dx}{\int_0^1 \mu_{Ti}(x) dx} \quad (1)$$

These truth values are defined by membership functions following Ullah and Harib (2008):

$$CE = \frac{\sum_{j=1}^2 TV_{Qj}}{2} \quad (2)$$

$$I(TV_{Qj}) = \begin{cases} \frac{TV_{Qj}-0}{0.5-0}, & TV_{Qj} \in [0, 0.5] \\ \frac{1-TV_{Qj}}{1-0.5}, & otherwise \end{cases} \quad (3)$$

$$RE = \begin{cases} 1, & TV_R \leq b \\ \frac{a-TV_R}{a-b}, & TV_R \in (b, a] \\ 0, & TV > a \end{cases} \quad (4)$$

The values of CE and RE for {P11, P12} are 0.54 and 1, respectively, whereas the values of CE and RE for {P13, P14} are 0.54 and 0, respectively.

Table 3. Settings for determining the information content of C1

	Propositions	Truth values	Requirement
P11	C1 is suitable for stakeholders' requirements	T2 = 0.27	Stakeholders are satisfied
P12	C1 is not suitable for stakeholders' requirements	T4 = 0.73	
P13	Performance of C1 is satisfactory for KM and developing configuration projects	T4 = 0.73	KM and development are satisfactory
P14	Performance of C1 is not satisfactory for KM or developing configuration projects	T2 = 0.27	

We conducted the same calculation for C2 and the results are illustrated in Table 4. The values of CE and RE of {P21, P22} are 1 and 1, respectively, whereas the values of CE and RE of {P23, P24} are 0.2 and, respectively 0.

Table 4. Settings for determining the information content of C2

	Propositions	Truth values	Requirement
P21	C2 is suitable for stakeholders' requirement	T3 = 0.5	Stakeholders are satisfied
P22	C2 is not suitable for stakeholders' requirement	T3 = 0.5	
P23	Performance of C2 is satisfactory for KM and development of configuration projects	T5 = 0.9	KM and development are satisfactory
P24	Performance of C2 is not satisfactory for KM and development of configuration projects	T1 = 0.1	

When all certainty and requirement compliances are known for a material alternative, it is possible to measure its coherency, which quantifies its balance in terms all certainty and requirement compliances, using the following function:

$$\lambda = |c - d| \times |e - f| + c + d + e + f \quad (5)$$

where

$$C = \max (CC); d = \min (cc); e = \max (RC); \text{ and } f = \min (RC).$$

The overall information content of C2 is 3 (in terms of coherency measure), which is a higher value than C1 (2.08). The epistemic challenge thus has high information content (i.e., it is indeed a challenge), whereas the compelling reason has low information content (i.e., it is indeed a compelling reason). Moreover, based on previous research, the information content of design in the sense of epistemic uncertainty should be maximized to remain creative (Sharif Ullah et al., 2012). However, when new knowledge is available, the information content should significantly decline (Sharif Ullah et al., 2012).

4. Methodology

4.1. Research design

To investigate the proposed framework, we applied a multiple-case-study approach (Yin, 2009). According to Gummesson (2000), when collecting empirical data from large organizations, a qualitative approach can obtain enough detailed information. Case-study research enables comparisons of different theories and observations from empirical data (McCutcheon and Meredith, 1993; Van de Ven, 1989) and has considerable advantages over statistical methods (Finifter, 1993). However, it is important to consider that "case studies, like experiments, are generalizable to theoretical propositions and not to populations or universes" (Yin, 2009, p. 38) because cases are not sampling units and should be treated as experiments (Tsang, 2013).

4.2. Case selection

The study includes four configuration projects with similar characteristics in the same company. Two used DT in the developing process and two did not, so we could investigate the effects of DT in configuration development. The case company, which operates globally, specializes in process-plant technology. This study used the following company-selection criteria: (1) the company produces complex, highly engineered building elements

or components, (2) the project stakeholders have the potential and interest to use DT for KM, (3) potential access exists to senior experts and potential users, and (4) top managers in the case company sponsor and support all projects. All the configurators focus on sales to facilitate order fulfillment for engineers and managers at the company by providing high-speed, high-quality quotations for end customers. Hence, the stakeholders and end users of the system are the internal engineers and managers at the case company.

The selected projects involved the highly complicated engineering design of a chemical product and included stakeholders with different expertise. The company develops configurators to support the sales and production process internally. Hence, the end-users of the system are the engineers at the company. Their KM process also requires collaboration between all the involved resources with different backgrounds and expectations. Based on the experiences at the case company, a configuration system always risks rejection due to lack of satisfaction from stakeholders and end users. Table 5 shows the characteristics of the four projects.

Table 5. Project characteristics

	<i>New framework (DT and KM)</i>		<i>Existing KM framework</i>	
	<i>Case 1</i>	<i>Case 2</i>	<i>Case 3</i>	<i>Case 4</i>
Time frame (months)	11	8	9	12
Product complexity	Medium/high	Medium/high	Medium/high	Medium/high
Software platform	Sales configuration system		Sales configuration system	
Project participants (roles) in IT team	Project manager Data-management specialist Business analyst Development team (1 configuration engineer, 1 IT developer) Tester		Project manager Development team (1 configuration engineer, 1 IT developer) Tester	
Project participants (roles) in business	Process engineer Sales estimator Sales manager Department manager Mechanical engineer		Sales manager Department manager	

4.3. Data collection

Workshops were conducted to train the team to apply DT to KM, and feedback meetings and workshops were held to collect knowledge about the team's satisfaction (IT team and business stakeholders) and the challenges and drawbacks in applying DT to their configuration system KM. We formulated propositions as the basis for an agenda for further experimentation and empirical research combining KM and DT in innovative firms. After the workshops and observation of the DT process, semi-structured interviews with the IT team and stakeholders were conducted.

The steps in Table 2 were utilized to apply DT phases to the KM of configuration projects. We used preliminary prototyping and testing ideas as drawings and minor developments to illustrate the inputs and the user interface. In the next step, we prioritized the requirements from stakeholders. The case studies allowed us to evaluate the application of DT to configuration projects and compare the results with the current situations at the case company, where no one used DT for developing configuration projects.

Finally, interviews were conducted to deepen insight into DT performance from IT teams' and stakeholders' perspectives. Due to the lack of previous research, the study was designed as qualitative and exploratory, and open-ended data collection was applied (Bell and Bryman, 2007). We conducted two rounds of semi-structured qualitative interviews due to the multidisciplinary nature of the research. The first round explored the opinions of the IT team about DT's effect on configurators' main challenges, and the second focused on stakeholders (customers) and explored the innovation capabilities—resources, processes, and mindsets—so stakeholders could opine on DT's role in developing more innovative configurators compared to their previous experience.

We selected four interviewees: two from the IT team and two stakeholders. The four interviewees were the only personnel members present in all four configuration projects, which enabled them to answer every question about the four projects. They all had years of experience in configurators development and had all experienced

the case company's transition from traditional development to using DT for development. Thus, they could critically compare their experiences. As mentioned above, the configurators were internal IT teams for internal use at this case company, and the stakeholders were the engineers and managers at the case company (Table 6).

Table 6. Characteristics of interviewees

	IT team	IT team	Stakeholders	Stakeholders
Years in current position in the case company	7 years	10 years	5 years	8 years
Years of relevant experience in company	7 years	8 years	5 years	8 years
Current role	Configuration team manager, DT manager & facilitator	Senior IT developer, DT manager & facilitator	Sales manager, chemical engineer, business project manager	Technical leader, chemical engineer, business project manager

To collect the richest data possible, we asked several questions about the value of using DT, including the direct questions "What do you perceive to be the value of DT?" and "What has changed in the company since DT was introduced/what are the effects of using DT?"

4.4. Data analysis

Both qualitative and quantitative data were structured into tables to identify regularities (as recommended by Runeson and Höst, 2008) and similarities and differences among cases. An initial analysis was done on the archival documents collected. Two researchers independently pored over the documents and identified all the data relevant for understanding each case, including project context (organizational and team culture), scope, duration, applied theories, planning approaches, meetings, etc. This analysis produced first descriptions of the four projects, helped design a focused protocol and questionnaire, and identified specific aspects to investigate openly in the interview phase. A second set of analyses was performed in parallel with the interviews, further refining the cross-case comparison obtained with the first analysis. The first round of interviews was based on identified issues related to configurator challenges; and the second investigated innovation capabilities: resources, processes, and mindsets. The answers to the questions and the discussions about the answers were coded, cross-tabulated, and analyzed to find patterns and shared explanations. After concluding the interviews, the results on the influence of DT on configurator challenges were consolidated.

Finally, we measured the motivation of the study in the first stage (C1 and C2) and measured the overall information content of C'2. The results demonstrate that the epistemic challenges had both higher and lower information content than the compelling reasons. We discuss the results in the discussion and conclusions section.

5. Case application and findings

We analyzed DT along the social dimension of the stakeholders involved in the configuration projects. Focusing on internal stakeholders, we suggested how DT could contribute to addressing the challenges they encounter. DT emphasizes the iterative identification of stakeholders and promotes rich, frequent interactions with them, involving artifacts such as stimulators to develop empathy (Mahmoud-Jouini et al., 2014).

The steps in Table 2 were utilized to apply DT phases to the KM of configuration projects. The following sections describe the two DT-integrated configuration projects.

5.1. Exploring the problem space

Step 1: Determining the knowledge scope of the project

In this step, we explored the problem space, so we started by observing the situation and interviewing the stakeholders. Understanding the current working situation, thoughts, and expectations gave us journey mapping and AS-IS and TO-BE analysis. During the observations and conversations, we found a common understanding of the problem that all stakeholders agreed on, which was a great step toward a solution.

Brainstorming sessions supported agreements on challenges to determine the goals of the project. We used sketching and graphs to illustrate opinions. We did not use any standard visualization methods while discussing, but later, all the discussions were easily converted to class diagrams, use-case diagrams, Product Variant Masters (PVMs), and user stories. This step also gave us a clear understanding of different roles, their perspectives on the problem, their different expectations, and how they collaborate and influence each other. The main takeaway from

this step was the team collaboration across different disciplines to develop a general, mutual understanding of the problem.

Figure 5 shows an example of a user profile from the project.

Unit engineer

Description

Jimmy is a unit engineer working at a plant. His job is to keep his units run smoothly and to meet the quality level that is set by the production planner. He is responsible for operation and maintenance of the whole unit. He manages a number of work crews that operates the unit. Is depending on their skills to meet product quality.

He wants to spend time on: optimization

Work environment: He daily moves around the plant to check on things, but mostly he works in his office, next to the control room, where the unit operator is located. He is not allowed to bring e.g. a tablet around the plant, so he write notes on paper.

Cooperates with the unit operator who maintains the unit, the production planner who continuously calculates (and implements) optimizations and set quality standard, client tech service, tech service for trouble shooting, reports and support.

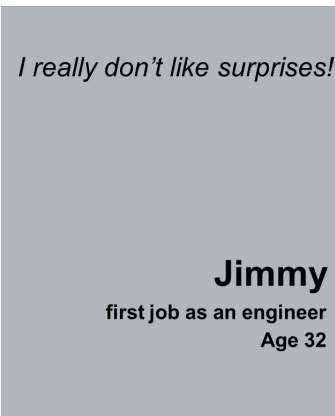


Figure 5. An example of a user profile

Step 2: Acquiring knowledge

In this step, we conducted interviews with individual stakeholders to understand their specific worries about the current situation and expectations for a future solution. We also gave them a general presentation about the solution, and we showed them examples of similar cases and the influence of configurators in eliminating challenges so they could perceive the solution and connect it to their current problems. They also asked many questions about the possibilities, which enabled the team to customize the solution based on their requirements. We then defined the solution, and we illustrated our first perceptions of different integrations and user interfaces. This work included “sketching”. However, due to the confidentiality of the projects, we cannot provide figures to demonstrate the ideation process at the case company. The main benefit of DT in this phase was to support the conceptualization of knowledge and determine the level of the details required for the system.

5.2. Exploring the solution space

Step 3: Developing and validating knowledge

In this step, we considered all the possible assumptions based on our data from previous steps and problem space. We searched within the existing solution space and looked at the stakeholders with different perspectives and ideas. In other words, we promoted outside-the-box thinking with stakeholders, which produced unusual ideas, as they were allowed to discuss anything. We then explored all the ideas using initial-benefit analysis, risk analysis, and their business cases regarding value creation and scalability.

Prototypes were created to reify abstract ideas and convert them into innovative solutions. Specifically, in this phase, we developed IT solutions with minimum time and effort to illustrate the concept for the users. The main benefit of DT was a clear understanding of the user interface and the inputs and outputs of the system.

After the prototyping was accepted and approved, the team started IT development. DT accelerated and streamlined this phase, as the team clearly understood the expectations and the system’s intended appearance.

Step 4: Documenting and maintaining knowledge

In this phase of the project, DT provided good documentation, not only of product knowledge, but also of brainstorming sessions, interviews, sketching, ideas, problem and solution spaces, workshops, and thinking-room outcomes. The documentation generated would enable any IT expert to maintain the system easily by reading the documents and examining prototypes when the proposed solution seemed inadequate or satisfactory or the project required updates and maintenance. In other words, DT produced documentation accessible to an audience beyond technical experts.

5.3. Iterative alignment of both spaces

As a configurator becomes more successful and popular among users, their expectations and requirements for the system tend to rise (Barker et al., 1989). DT must thus include the possibility of iterations in KM, necessitating

constantly observing the system application and exploring all the potential assumptions. As some users and stakeholders might change over time, the team must be present to have informal conversations to form future hypotheses. If the system provokes adverse reactions or dissatisfaction, it faces a great risk of system abundance. Hence, the team should start DT and KM iteration from Step 1.

5.4. Case comparison

Table 7 summarizes the two cases that used DT. Column 1 lists various DT techniques based on the literature, and columns 2 and 3 present the details of cases 1 and 2, respectively, using DT techniques. For techniques such as brainstorming and prototyping, the experiences were similar, while for sketching and interviewing, a few small differences emerged.

Table 7. DT techniques and experiences (cases 1 and 2)

Techniques	Experiences (contribution of DT techniques)	
	Case 1	Case 2
Sketching and journey mapping	<ul style="list-style-type: none"> • Understanding and agreement on their own requirements • Agreement on the AS-IS process by engineers and sales • Common understanding of the sales and technical structure of the product/process 	<ul style="list-style-type: none"> • Agreement on challenges and common understanding of the AS-IS by engineers and sales • User interface design • Project goals and stakeholder requirements • Product structure and goals of the project
Informal/formal conversation/interviews	<ul style="list-style-type: none"> • Full understanding of our proposal and discussions for domain experts • Full understanding of expectations and new insights 	<ul style="list-style-type: none"> • Workshop on the stakeholders' requirements and informal discussions • A thinking room for unusual observations and suggestions
Brainstorming	<ul style="list-style-type: none"> • Project goals and proposals (TO_BE) 	<ul style="list-style-type: none"> • Project goals and proposals (TO_BE)
Prototyping	<ul style="list-style-type: none"> • Prototyping the accepted proposals 	<ul style="list-style-type: none"> • Prototyping the accepted proposals

The two projects not using DT were carried out likewise. Tables 8 (IT team interviews) and 9 (stakeholder interviews) summarize the effects of DT application based on the semi-structured qualitative interviews, where participants compared the experiences of a normal and a DT-integrated project.

Table 8 presents the IT team interview questions and answers to compare cases 1 and 2 (using DT and the KM framework) to cases 3 and 4 (using the KM framework without applying DT). The first column contains the exact interview questions based on the challenges, and the second explains the importance of each challenge, so in the last column, the IT team outlined the justifications and the additional explanations for their preference to use DT in the future. Based on the interviews, DT improved organization and knowledge acquisition challenges in configurator development by involving more diverse stakeholders and enabling a full understanding of AS-IS and TO_BE processes. The knowledge representation was improved by mapping and visualizing knowledge, and DT met technical IT challenges by clarifying inputs, outputs, and user-interface characteristics.

Table 8. IT team evaluation of the application of DT methods (categories based on Kristjansdottir et al. (2018))

Configurator challenge	Importance of the configurator challenge	Key justifications for cases 1 and 2 compared to cases 3 and 4
Did DT meet the organizational challenges?	Very high	DT improved communication, involving more stakeholders and considering their requirements. DT also facilitated change management during configurator implementation, and allocating stakeholders was easier, as they were directly involved and interested.
Did DT meet the knowledge-acquisition challenges?	High	Due to involving more stakeholders from different disciplines, considering and highlighting their ideas and requirements, acquiring knowledge was easier and faster. DT also enabled a full understanding of AS-IS and TO_BE processes.
Did DT meet the product modeling (knowledge representation) challenges?	Medium	DT empowered team with strong communication, meetings, and stakeholder involvement from the very beginning of the project. DT helped map and visualize all the required knowledge to be modeled inside the system.
Did DT meet the IT (technical) challenges?	Medium	DT facilitated IT-development clarifications, such personalization, inputs, outputs, and user interface design, due to the direct intervention of stakeholders.

Did DT meet the resource constraints challenges?	Low	NA
Did DT meet the product-related challenges?	Low	NA

Table 9 presents the results from the interviews with stakeholders. The interview questions were based on Carlgren's (2014) categories of the benefits of DT within the framework of innovation capabilities: resources, processes, and mindsets. In this interview round, we asked the stakeholders to reflect on their justifications for cases 1 and 2 (DT-based projects) compared to cases 2 and 3. The results show that DT benefited resource management, development processes and mindsets, and culture changes at the case company.

Table 9. Stakeholder evaluation of DT benefits (categories based on Carlgren et al. (2014))

Strategy	Overall questions	Key justifications for cases 1 and 2 compared to cases 3 and 4
Resources	How does DT improve the management of resources for configurator projects, including knowledge and competence base, technology, networks, and relations	DT gave employees a more holistic understanding of the developed system, other opportunities and applications for configurators, motivation, empowerment, innovation, diversity, and room for different personalities. One interviewee mentioned, however, that the development was slower.
Processes	How does DT improve the process of development for configuration projects, including organizational structures, managerial systems, generative processes, and ways of working?	Learning from the end users, more innovative outputs, better understanding the problem, maximizing configurators, accelerating feedback cycles with stakeholders, reducing time and development waste, collaborating and innovating across multidisciplinary teams, and communicating with managers.
Mindset	How does DT facilitate the gradual changing of values, cultures, and norms in the company?	Understanding the reasons for changes, early disagreement and argument as key to innovation, innovative awareness in the team, and competition in the team. One interviewee mentioned the difficulty of managing DT at times, however, as some employees perceived it as not fun.

Besides the experiences from the different phases, from a wider perspective, more general conclusions about the application of DT to configuration projects can be drawn based on the interviews:

1. Supporting configuration projects with DT requires additional time and resources.
2. Supporting configuration projects with DT decreases development time.
3. DT clearly defines problems for the IT team and the stakeholders for a better understanding of what their needs and expectations should be. Many good solutions failed due to lacking a meaningful problem space.
4. DT can help describe and validate the problem domain. Once you have developed a few prototypes and successfully tested them, everyone can profoundly understand what the perfect solution should look like.
5. Normally, the solution based on the IT team's understanding is different from what users need. DT can play the role of a meta-disciplinary rationale, which allows a team to develop a general, mutual understanding of the problem and solution across disciplines.
6. Determining the level of details in configuration projects is difficult, and DT enables the IT team to scope the project more easily.
7. The IT team manager aligned the stakeholders and implemented the system more easily and managed the changes at the company by using DT and involving users while designing the system.
8. Previously, the IT team had to re-do projects once or even twice because they lacked a clear understanding of the users' needs or were just receiving feedback from a few stakeholders, not all of them. Comparing the previous state with cases 1 and 2 using DT, the team spent more time in the beginning of the project, but since the scoping and decision processes were based on communication and involvement, the development and implementation were fast and easy.

There were also some challenges reported, such as the time needed to learn techniques and additional DT discussions. Some employees also found it difficult to learn and work with DT. In one case project, an interviewee mentioned the difficulty of maintaining the motivation of project participants in large-scale DT projects, and DT activities were considered an additional burden.

5.5. Measuring motivation after case project implementation and interviews

According to the case study and interview-based validation, the calculation results change as the truth values changed. In fact, based on Figure 4, K4 has been changed to K'4 due to the acquired knowledge, and we changed C2 to C'2 (Table 10). Here, we investigate the new propositions quantitatively. The values of CE and RE for {P31, P32} are 0.2 and 1, respectively, whereas the values for {P33, P34} are 0.2 and 0, respectively.

Table 10. Settings for determining the information content of C'2

	Propositions	Truth values	Requirement
P31	C'2 is suitable for stakeholders' requirements	T5 = 0.9	Stakeholders are satisfied
P32	C'2 is not suitable for stakeholders' requirements	T1 = 0.1	
P33	Performance of C'2 is satisfactory for KM and developing configuration projects	T5 = 0.9	KM and development are satisfactory
P34	Performance of C'2 is not satisfactory for KM or developing configuration projects	T1 = 0.1	

The overall information content of C'2 is equal to 1.4 (in terms of coherency measure), which is lower than C1 (2.08). Hence, the results confirm that when new knowledge is available, the information content should significantly decline, and while pursuing a conceived (creative) concept in the presence of new knowledge, the information content should be minimized (Sharif Ullah et al., 2012).

6. Discussion and conclusions

In this research, we acquired results from a case company applying DT, extended with C-K theory, in configuration projects. The results show that DT can support the KM of configuration-system development. A design process in C-K theory is motivation-driven; this motivation can be quantified by information content (entropy) measured under epistemic uncertainty. To measure the information content of C1 and C2 with respect to the C-K map in terms of CE and RE, we defined propositions and calculated the overall information content accordingly. The results indicate that the epistemic challenge has higher information content (C2 is equal to 3) than the compelling reason (C1 is equal to 2.08). Conceiving a creative concept involves maximizing the information content; however, pursuing a conceived (creative) concept in the presence of new knowledge involves minimizing information content. Our multiple-case-study approach compared two projects involving the conceptual framework of DT and another two similar projects not involving DT. After the case-study research and interviews, we gained new knowledge and recalculated the information content again, indeed witnessing the minimization of information content (C'2 is 1.4).

First, we defined the potential users of the system and the states in which the user applies DT for a common understanding of the expectations of the configuration system. In this stage and during several workshops, we defined the problem from different perspectives. Second, we started prototyping the configuration system, a quick, effective way to initiate designing the solution. Based on the stakeholders' comments, this prototype helped them realize their expectations:

1. Providing the visual guide for a website, which helped stakeholders look at layout without thinking about aesthetics
2. Creating a consistent level of features or pages to test with users, management team, other colleagues, etc.
3. Addressing doubts as the IT team designed features in detail
4. Facilitating the developers'/testers' work with a prototype that specified interactions, feedback, and notifications, as these are parts of the user journey

The proposed framework and case study make two contributions. First, we contribute to the product-configuration literature by providing a framework that integrates DT and C-K theory into the development processes of such systems. The case study demonstrates the usefulness of the framework. Thus, the proposed DT framework and empirical study extend research on executing configuration projects (Forza & Salvador, 2006; Haug, 2008; Hvam et al., 2008; Shafiee et al., 2018). Second, we contribute to the design literature by extending the applications of DT and C-K theory to the area of product configuration, specifically expanding the configuration-project manager's toolbox of techniques and methods. Specifically, it supports increased user involvement in such projects, which may address user resistance (Haug et al., 2019b). The major limitation of the research is that the framework was tested at only a single company, which limits the generalizability of the results. Future research should therefore further investigate the potential and limitations of applying DT and C-K theory in configuration projects by studying them at other companies. More specific studies applying individual DT techniques are also needed.

Reference

- Barker, V.E., O'Connor, D.E., Bachant, J. and Soloway, E. (1989), "Expert systems for configuration at Digital: XCON and beyond", *Communications of the ACM*, Vol. 32 No. 3, pp. 298–318.
- Basili, V.R. and Weiss, D.M. (1984), "A methodology for collecting valid software engineering data", *IEEE Transactions on Software Engineering*, Vol. SE-10 No. 6, pp. 728–738.
- Beckman, S.L. and Barry, M. (2007), "Innovation as a learning process: Embedding design thinking", *California Management Review*, Vol. 50 No. 1, pp. 25–56.
- Bell, E. and Bryman, A. (2007), "The ethics of management research: an exploratory content analysis", *British Journal of Management*, Vol. 18 No. 1, pp. 63–77.
- Blazek, P. and Pilsle, K. (2017), "Learnings From Setting Up Product Configurator Projects", *Annals of the Faculty of Engineering Hunedoara - International Journal of Engineering*, Vol. 15 No. 1, pp. 25–28.
- Braha, D. and Reich, Y. (2003), "Topological structures for modeling engineering design processes", *Research in Engineering Design*, Vol. 14 No. 4, pp. 185–199.
- Brown, T. (2008), "Design 84", *Harvard Business Review*, No. June, pp. 85–92.
- Carlgren, L., Elmquist, M. and Rauth, I. (2014), "Design thinking: Exploring values and effects from an innovation capability perspective", *Design Journal*, Vol. 17 No. 3, pp. 403–424.
- Dorst, C.H. (2006), "Design problems and design paradoxes", *Design Issues*, Vol. 22 No. 3, pp. 4–17.
- Dunne, D. and Martin, R. (2006), "Design thinking and how it will change management education: An interview and discussion", *Academy of Management Learning and Education*, Vol. 5 No. 4, pp. 512–523.
- Felfernig, A., Hotz, L., Bagley, C. and Tiihonen, J. (2014), *Knowledge-Based Configuration from Research to Business Cases*, Morgan Kaufmann, Newnes, NWS, Australia, available at: <https://doi.org/10.1016/B978-0-12-415817-7.00029-3>.
- Finifter, A.W. (1993), *Political Science: The State of the Discipline II*, American Political Science Association, Washington, DC.
- Forza, C. and Salvador, F. (2002), "Managing for variety in the order acquisition and fulfilment process: the contribution of product configuration systems", *International Journal of Production Economics*, Vol. 76 No. 1, pp. 87–98.
- Forza, C. and Salvador, F. (2006), *Product Information Management for Mass Customization: Connecting Customer, Front-Office and Back-Office for Fast and Efficient Customization*, Palgrave Macmillan, New York, NY.
- Forza, C., Trentin, A. and Salvador, F. (2006), "Supporting product configuration and form postponement by grouping components into kits: the case of MarelliMotori", *International Journal of Mass Customisation*, Vol. 1 No. 4, pp. 427–444.
- Gummesson, E. (2000), *Qualitative Methods in Management Research*, Sage publications, Inc., California, USA.
- Hatchuel, A., Masson, P.L.E. and Weil, B. (2008), "Teaching innovative design reasoning: how could C-K theory help?", *Proceedings of E and PDE 2008, the 10th International Conference on Engineering and Product Design Education*, No. September.
- Hatchuel, A. and Weil, B. (2009), "C-K design theory: an advanced formulation", *Research in Engineering Design*, Vol. 19 No. 4, pp. 181–192.
- Haug, A. (2008), *Representation of Industrial Knowledge as a Basis for Developing and Maintaining Product Configurators*, Technical University of Denmark, Denmark, available at: <http://forskningbasen.deff.dk/Share.external?sp=S7c4c9d41-faa9-4544-bbca-fdad9b8d9970&sp=Sdtu>.
- Haug, A. (2013), "Improving the design phase through interorganisational product knowledge models", *International Journal of Production Research*, Vol. 51 No. 2, pp. 626–639.
- Haug, A., Shafiee, S. and Hvam, L. (2019a), "The costs and benefits of product configuration projects in engineer-to-order companies", *Computers in Industry*, Vol. 105, pp. 133–142.
- Haug, A., Shafiee, S. and Hvam, L. (2019b), "The causes of product configuration project failure", *Computers in Industry*, Vol. 108, pp. 121–131.
- Heiskala, M., Paloheimo, K. and Tiihonen, J. (2007), "Mass customization with configurable products and configurators: a review of benefits and challenges", *Mass Customization Information Systems in Business*, IGI Global, Finland, pp. 75–106.
- Hvam, L., Mortensen, N.H. and Riis, J. (2008), *Product Customization*, Springer-Verlag, Berlin Heidelberg, Germany, available at: <https://doi.org/10.1007/978-3-540-71449-1>.
- Kolko, J. (2011), *Exposing the Magic of Design: A Practitioner's Guide to the Methods and Theory of Synthesis*, Oxford University Press.
- Kristjansdottir, K., Shafiee, S., Hvam, H., Forza, C. and Mortensen, N.H. (2018), "The main challenges for manufacturing companies in implementing and utilizing configurators", *Computers in Industry*, Vol. 100, pp. 196–211.
- Lindberg, T., Köppen, E., Rauth, I. and Meinel, C. (2012), "On the perception, adoption and implementation of design thinking in the IT industry", *Design Thinking Research*, Springer, pp. 229–240.

- Lindberg, T., Meinel, C. and Wagner, R. (2011), "Design thinking: a fruitful concept for IT development", in Meinel, C., Leifer, L. and Plattner, H. (Eds.), *Design Thinking*, Springer, Berlin Heidelberg, pp. 3–18.
- Lockwood, T. (2010), *Design Thinking: Integrating Innovation, Customer Experience, and Brand Value*, Skyhorse Publishing, Inc.
- Mahmoud-Jouini, S. Ben, Midler, C., Cruz, V. and Gaudron, N. (2014), "How physical artefacts contribute to design processes", *Academy of Management Proceedings*, Vol. 2014, Academy of Management, p. 14113.
- Le Masson, P., Weil, B. and Hatchuel, A. (2017), *Design Theory: Methods and Organization for Innovation*, Springer International Publishing, Gewerbestrasse, Cham, available at: <https://doi.org/10.1007/978-3-319-50277-9>.
- McCutcheon, D.M. and Meredith, J.R. (1993), "Conducting case study research in operations management", *Journal of Operations Management*, Vol. 11 No. 3, pp. 239–256.
- Meinel, C. and Leifer, L. (2015), *Design Thinking Research: Building Innovators, Design Thinking Research: Understanding Innovation*, Springer, available at: <https://doi.org/10.1016/B978-0-12-387667-6.00013-0>.
- Nonaka, I. (1994), "A dynamic theory of organizational knowledge creation", *Organization Science*, Vol. 5 No. 1, pp. 14–37.
- Petersen, T.D. (2007), "Product configuration in ETO companies", in Blecker, T. (Ed.), *Mass Customization Information Systems in Business*, IGI Global, pp. 59–76.
- Polanyi, M. (1969), *Personal Knowledge*, Isis, Chicago.
- Polanyi, M. and Nye, M.J. (2015), *Personal Knowledge: Towards a Post-Critical Philosophy*, The University of Chicago Press, Chicago, IL.
- Ravasi, D. and Stigliani, I. (2012), "Product Design: A Review and Research Agenda for Management Studies", *International Journal of Management Reviews*, Vol. 14 No. 4, pp. 464–488.
- Razzouk, R. and Shute, V. (2012), "What is design thinking and why is it important?", *Review of Educational Research*, Vol. 82 No. 3, pp. 330–348.
- Runeson, P. and Höst, M. (2009), "Guidelines for conducting and reporting case study research in software engineering", *Empirical Software Engineering*, Vol. 14 No. 2, pp. 131–164.
- Salvador, F. and Forza, C. (2004), "Configuring products to address the customization-responsiveness squeeze: A survey of management issues and opportunities", *International Journal of Production Economics*, Vol. 91 No. 3, pp. 273–291.
- Seidel, V.P. and Fixson, S.K. (2013), "Adopting design thinking in novice multidisciplinary teams: the application and limits of design methods and reflexive practices", *Journal of Product Innovation Management*, Vol. 30 No. Suppl. 1, pp. 19–33.
- Shafiee, S. (2017), *Conceptual Modelling for Product Configuration Systems*, Technical University of Denmark, Denmark.
- Shafiee, S., Haug, A., Shafiee, S. and Hvam, L. (2019), "Complementing the Scoping Process of Configuration Projects by Design Thinking", available at: <https://doi.org/10.3233/ATDE190105>.
- Shafiee, S., Hvam, L. and Bonev, M. (2014), "Scoping a product configuration project for engineer-to-order companies", *International Journal of Industrial Engineering and Management*, Vol. 5 No. 4, pp. 207–220.
- Shafiee, S., Hvam, L., Haug, A., Dam, M. and Kristjansdottir, K. (2017), "The documentation of product configuration systems: A framework and an IT solution", *Advanced Engineering Informatics*, Vol. 32, pp. 163–175.
- Shafiee, S., Kristjansdottir, K., Hvam, L. and Forza, C. (2018), "How to scope configuration projects and manage the knowledge they require", *Journal of Knowledge Management*, Vol. 22 No. 5, pp. 982–1014.
- Shafiee, S., Wautelet, Y., Hvam, L., Sandrin, E. and Forza, C. (2020), "Scrum versus Rational Unified Process in facing the main challenges of product configuration systems development", *The Journal of Systems & Software*, Elsevier Inc., Vol. 170, p. 110732.
- Sharif Ullah, A.M.M., Rashid, M.M. and Tamaki, J. (2012), "On some unique features of C-K theory of design", *CIRP Journal of Manufacturing Science and Technology*, CIRP, Vol. 5 No. 1, pp. 55–66.
- Song, Y.W., Kim, J.D., Yu, L., Lee, H.K. and Lee, H.S. (2012), "A comparative study of the telematics industry in Korea and China", *Journal of Internet Banking and Commerce*, Vol. 17 No. 1, pp. 144–156.
- Speel, P.H., Schreiber, A.T., Van Joolingen, W., Van Heijst, G. and Beijer, G.J. (2001), "Conceptual modelling for knowledge-based systems", *Encyclopedia of Computer Science and Technology*, Vol. 44, pp. 107–132.
- Studer, R., Benjamins, V.R. and Fensela, D. (1998), "Knowledge engineering: Principles and methods", *Data & Knowledge Engineering*, Vol. 25 No. 1–2, pp. 161–197.
- Tsang, E.W.K. (2013), "Case study methodology: Causal explanation, contextualization, and theorizing", *Journal of International Management*, Elsevier, Vol. 19 No. 2, pp. 195–202.
- Tsoukas, H. (2003), "Do we really understand tacit knowledge?", in Easterby-Smith, M. and Lyles, M.A. (Eds.), *The Blackwell Handbook of Organizational Learning and Knowledge Management*, Blackwell Publishing Ltd, Oxford, UK, pp. 410–427.
- Ullah, A.M.M.S. (2005a), "Handling design perceptions: An axiomatic design perspective", *Research in*

- Engineering Design*, Vol. 16 No. 3, pp. 109–117.
- Ullah, A.M.M.S. (2005b), “A fuzzy decision model for conceptual design”, *Systems Engineering*, Vol. 8 No. 4, pp. 296–308.
- Ullah, A.M.M.S. and Harib, K.H. (2008), “An intelligent method for selecting optimal materials and its application”, *Advanced Engineering Informatics*, Vol. 22 No. 4, pp. 473–483.
- Van de Ven, A.H. (1989), “Nothing is quite so practical as a good theory”, *Academy of Management Review*, Vol. 14 No. 4, pp. 486–489.
- Vetterli, C., Falk, Ü., Walter, B., Franziska, H., Thomas, K., Jens, K., Jürgen, M., et al. (2013), *Jumpstarting Scrum with Design Thinking*, available at: <https://www.alexandria.unisg.ch/224403>.
- Walcher, D. and Werger, L. (2012), “Why MC organizations fail—the classification of failure reasons”, *2011 World Conference on Mass Customization, Personalization, and Co-Creation (MCPC)*, Lulu, Raleigh.
- Wang, Y., Wu, J., Zhang, R., Shafiee, S. and Li, C. (2020), “A ‘user-knowledge-product’ Co-creation cyberspace model for product innovation”, *Complexity*, Vol. 2020 No. 1, available at: <https://doi.org/10.1155/2020/7190169>.
- Yin, R.K. (2009), *Case Study Research: Design and Methods (Applied Social Research Methods)*, Sage, Thousand Oaks, CA.
- Yoshikawa, H. (1981), “General design theory and a CAD system”, in Sata, E. and Warman, E. (Eds.), *Man-Machine Communication in CAD/CAM*, Amsterdam, Netherlands, pp. 35–58.
- Zhang, L.L. (2014), “Product configuration: a review of the state-of-the-art and future research”, *International Journal of Production Research*, Vol. 52 No. 21, pp. 6381–6398.
- Zhang, L.L., Lee, C.K.M. and Akhtar, P. (2020), “Towards customization: Evaluation of integrated sales, product, and production configuration”, *International Journal of Production Economics*, Vol. 229, p. 107775.