A Framework of Design Method Corroboration

Thoring, Katja; Mueller, Roland M.; Lecuna, Alejandro; Badke-Schaub, Petra

Publication date
2023

Document Version
Final published version

Published in
Proceedings of the 56th Annual Hawaii International Conference on System Sciences, HICSS 2023

Citation (APA)
Thoring, K., Mueller, R. M., Lecuna, A., & Badke-Schaub, P. (2023). A Framework of Design Method Corroboration. In T. X. Bui (Ed.), *Proceedings of the 56th Annual Hawaii International Conference on System Sciences, HICSS 2023* (pp. 5156-5165). (Proceedings of the Annual Hawaii International Conference on System Sciences; Vol. 2023-January). IEEE. https://hdl.handle.net/10125/103265

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

A Framework of Design Method Corroboration

Katja Thoring
Technical University of Munich
katja.thoring@tum.de

Roland M. Mueller
Berlin School of Economics and Law
roland.mueller@hwr-berlin.de

Alejandro Lecuna
Anhalt University
alejandrolecuna@gmail.com

Petra Badke-Schaub
TU Delft
p.g.badke-schaub@tudelft.nl

Abstract

Practitioners design artifacts of different kinds. Researchers and practitioners both create methods for designing such artifacts. The question arises whether those methods are actually valid and useful. In this conceptual paper, we argue that there is a need for "method corroboration"—the deliberate and reflected use and possible validation of a design method. We present a literature review of method corroboration in the IS and more specifically in the DSR literature. The findings are summarized as a conceptual model outlining eight strategies of method use, which are then condensed into a 2-by-2 framework of method corroboration. The results of this paper present insight into the current state of method corroboration in the DSR field and provide guidance for working with design methods in research and practice.

Keywords: design science research, design methods, method evaluation, method validation, research rigor**.**

Introduction

A design method is defined as a descriptive or prescriptive procedure on how to create new artifacts (Cross, 2021). It provides "procedures, techniques, aids, or tools for designing" (J. C. Jones, 1992, p. 290). In that sense, the term "design method" encompasses both, abstract principles or step-by-step instructions and concrete tools for how to design something.

Design research unites a wealth of disciplines, which requires a suitable approach to bring structure to the questions and applications, and the possibility of reducing the complexity of the field to such an extent that events, constructs, and applications become meaningfully manageable. This is the task of design methodology. A design method is thus a structured procedure of complexity reduction of findings, events, constructs, and applications from a variety of perspectives and research fields (Badke-Schaub & Voute, 2018).

However, there are several problems related to design methods: (1) According to Cross (1993), design methodology suffers from a lack of confidence by design practitioners, and it has little (acknowledged) practical application. At the same time, (2) the actual impact and effectiveness of a method is often not well understood. If we look, for example, at brainstorming (Osborn, 1953), a very popular design method for developing ideas in groups, research has recently shown that it actually produces "fewer and poorer quality ideas than the same number of individuals working alone" (Furnham, 2000). This example illustrates the need for a thorough understanding or even validation of a method. We argue that it is crucial to know if and when a method works and when it does not. Still, many design methods are used in blind trust. As stressed by Badke-Schaub and Voute (2018) the actual scientific rigor of design methods is often doubtful, since there is usually little to none evidence about their actual impact and usefulness. Finally, (3) many new design methods are being developed from scratch. Often, these appear to be rather arbitrary. For example, Roberts (2016) presents a list of more than 100 innovation canvases. However, most of them did not describe their development process, nor were they empirically evaluated or theoretically understood (Thoring et al., 2019).

We address the three problems outlined above within the Information Systems (IS) discipline for the following reasons: The IS discipline, and specifically the design science research (DSR) field is concerned with the design of artifacts, where artifacts are mostly considered as socio-technical IT artifacts (Hevner et al., 2004). Hence, here the need for and use of design methods is well-established. Furthermore, Hevner et al. (2004, p. 77; emphasis added) define IT artifacts "as constructs (vocabulary and symbols), models (abstractions and representations), *methods* (algorithms and practices), and instantiations (implemented and prototype systems)".

Walls et al. (1992) differentiate between prescriptive and descriptive theories. Prescriptive

URI: https://hdl.handle.net/10125/103265
978-0-9981331-6-4

theories resemble methods in so far that both are concerned with the questions of an intervention that changes the initial state into a preferred state. Davison et al. (2021) mention instrumental theories that try to solve a problem or develop an organization.

We argue that design methods could offer a process for designing (1) *technological* resources or (2) any kinds of resources. Hevner et. al (2004, p. 100) limit design science to activities "with respect to technology, engaging in the creation of technological artifacts [...]". Technological resources behave more predictable than people. Procedural and structural constraints might place limits to their choices but they "always have the possibility to do otherwise" (M. R. Jones & Karsten, 2008, p. 137) and are therefore challenging as a part of a prescriptive method.

IS is not only concerned with the design of new artifacts (and the related involvement of design methods), but also with the development of new methods. Consequently, we identify two levels of interest within this context: What are the methods for designing IT artifacts? And how can we design a new IS method itself? We argue that for both, we need a thorough understanding of a method's effectiveness and working mechanisms. We argue that there is a need for what we call "method corroboration" – that is, the deliberate and reflected use, development, and possible validation of design methods, possibly increasing their theoretical understanding and/or empirical validation.

While the DSR discipline provides several rigorous guidelines, frameworks, and step-by-step instructions for artifact development on an abstract scientific level, e.g., (Gregor & Jones, 2007; Peffers et al., 2007; Venable et al., 2016), there is less research on concrete, practical *design methods and tools* that would help researchers and practitioners to actually better design an IT artifact. Moreover, we can find several sources dealing with the construction of DSR *research methods* (Möller et al., 2020; Venable & Baskerville, 2012) but not many sources presenting research on how to validate or develop DSR *design methods*. Consequently, the goal of this conceptual paper is guided by the following research questions:

RQ1: What is the state of design method corroboration according to the IS literature?

RQ2: How could the corroboration of design methods be increased?

The term "corroboration" is used throughout this paper as an indicator for a robust and valid design method.

The remainder of this conceptual paper is structured as follows: First, we provide a systematic literature review on method corroboration within the Basket of Eight of top IS journals. Based on the findings, we derive (1) a conceptual model outlining the possible approach of design method usage, and (2) a structured 2-by-2 framework that defines different degrees of method corroboration. We conclude with a critical discussion and provide an outlook to future work.

2. Literature review of design methods in Information Systems

In order to understand if there are different levels of method corroboration and how these are applied in the Information Systems discipline, we take a look into the Information Systems DSR literature.

We created a sample of Information Systems papers based on the Senior's Scholars Basket of Eight IS journal list. We limited the papers to the last 10 years from 2012 to 2022. We applied the IS Ontology (Mueller et al., 2022) and only looked at DSR papers that mentioned design methods or one of its synonyms (as defined in the IS Ontology). The following synonyms for "design method" were used in the full text search (including the plural version): design methodology, design process, DSR process, design science process, DSR process model, design science research process, design science research process model, and framework for evaluation in design science.

This procedure created a list of 53 candidate DSR papers. We looked at each candidate paper manually and excluded (1) editorial papers, (2) papers that only discussed research methods, and (3) papers that only presented design artifacts but not the methods for designing these artifacts. This reduced the list of papers to 13. Each paper was analysed, and types of method usages were noted in an Excel sheet. These codes were collected, clustered, and combined by two authors until no new concepts emerged. Table 1 shows an excerpt of our coding database, in which selected phrases from the analysed papers are presented to illustrate the allocation to the respective categories.

Table 1: Selected key phrases used to allocate the method usage category

Paper	Selected relevant key phrases	Allocated category
(Thummadi & Lyytinen, 2020)	We collected process data from six software projects with the goal of understanding the extent to which development routines differed in terms of the two design methods.	Observe
(Payton, 2016)	[…] followed by design lessons as offered by the MyHealthImpactNetwork.org user experience.	Reflect

Paper	Selected relevant key phrases	Allocated category
(Zhang et al., 2021)	[…] two coders […] conducted comparative analysis to identify key characteristics of the design practices […] by detailing how each process unfolded […].	Describe
(Pee et al., 2021)	The system was […] used to estimate and simulate […] with high accuracy.	Evaluate
(Barrett & Holeman, 2017)	[…] we draw on the notions of material 'back-talk' […] and concomitant practice breakdown […]	Understand
(Pee et al., 2021)	The guidelines are illustrated with a design science research project	Copy/Apply
(Thummadi & Lyytinen, 2020)	Three software development projects were carried out with the object-oriented method and three followed agile methods.	Adapt
(Pee et al., 2021)	This article […] develop guidelines for […]	Develop

Table 2 shows the result of the literature analysis. The resulting eight categories identified through the literature review will be described in the following section. More specifically, in Section 3.1 we develop a conceptual model of possible method usage based on the insights from the literature, before we delve into developing a framework of method corroboration in Section 3.2.

3. Toward a framework of method corroboration

3.1. Conceptual model of method usage

Based on the insights from the literature review, we suggest a conceptual model of method usage, outlining three strategic layers: (1) *orientation* (including observation, reflection, and description), (2) *corroboration* (including empirical evaluation and theoretical understanding), and (3) *action* (including application and adaptation of existing methods, and developing new methods).

The underlying eight strategies of how to achieve method corroboration are as follows: The methods need to be (1) carefully observed and (2) reflected; (3) they need to be presented as a formal description; (4) one needs to develop a theoretical understanding of the method's working mechanisms; (5) the methods can be empirically tested and evaluated; (6) methods can be copied and applied; (7) existing methods can be adapted to new contexts, and (8) new methods can be developed from scratch.

This is partly in-line with existing categorizations of evolutionary-supported cognitive modes that distinguish, for example, between generating, selecting, comparing, and evaluating methods (Stempfle & Badke-Schaub, 2002). The relationship of the eight strategies is illustrated in Figure 1. We will

Table 2. Classification of design method usage in selected DSR papers

Paper	Designed Artifact	Observe	Reflect	Describe	Evaluate	Understand	Copy/Apply	Adapt	Develop
Thummadi & Lyytinen, 2020	Software	X			X			X	
Janiesch et al., 2020	Service Platform				X	X	X	X	X
Zhang et al., 2021	Microchips	X	X	X	X	X			
Miah et al., 2019	DSS				X	X	X		X
Pee et al., 2021	Future artifacts				X		X		X
Zaitsev & Mankinen, 2022	Financial education applications	X					X		
Brandt et al., 2018	Cyberphysical systems				X				X
Barrett & Holeman, 2017	ICT4D	X	X			X			
Kolkowska et al., 2017	Information security			X	X				X
Payton, 2016	Health Portal	X	X						
Spagnoletti et al., 2015	Elderly assistance	X	X				X		
Silsand & Ellingsen, 2014	Electronic patient record	X	X						
Yang et al., 2012	Emergency response	X			X		X		X

discuss each of the eight strategies in more detail in the following subsections.

(1) Observe. The first step toward method corroboration is to carefully observe the methods and processes. This can be either the observation and reflection of one's own approach, or of an existing method that is applied by others. A typical approach in the design field are protocol analysis (Ericsson & Simon, 1984; Hay et al., 2017)—a systematized approach for observing and analysing team behaviour.

(2) Reflect. A critical reflection of the observed methods and processes is required to describe a method in a formalized and reproducible way. Jobst et al. (2020) introduced a tool to facilitate reflection in the design process.

(3) Describe. Formalized descriptions of a method can be derived, for example, based on method engineering, a systematic approach that is concerned with the description, design, adaption, and evaluation of methods, using engineering principles (Welke & Kumar, 1992). Thoring and Mueller (2011) introduced a formalized process model of the design thinking process, based on method engineering. The description of a method constitutes the first step and the required foundation for copying and applying a method and for adapting and/or designing it.

(4) Evaluate. Empirical testing and evaluating of a method can provide additional insights. When testing and evaluating methods, we can distinguish two major goals: testing the usefulness of the method (does it actually work?) and the usability of the method (is it easy to use, understand, and apply?). The question of how to evaluate a method can also take two different approaches: (1) Experiments result in measurable results, but are often performed with students, due to their easy availability. Hence, these kinds of evaluations provide a high internal validity, but typically a low external validity. By contrast, (2) case studies, action research, and workshops in a practitioner's context provide a high external validity but lower internal validity, because of the higher practical relevance of such qualitative studies, whereas access to practitioners and real cases is usually limited.

Thoring et al. (2020) present a set of guidelines for evaluating artifacts (such as methods) through workshops.

(5) Understand. By contrast, other approaches focus on the theoretical understanding of a method's working mechanism, first, rather than or before conducting empirical validation studies. One example is presented by Kannengiesser and Gero (2019). The authors refer to the dual systems theory and the book "Thinking fast and slow" by Daniel Kahneman (2013) and map this to the design thinking process. This approach might lead to a better understanding of the design thinking process steps and how and when these should be applied. Another example is the "theory of creative workspace design" and its impact on the creative process (Thoring et al., 2021). The authors suggest a set of propositions of how the physical workspace design might influence creativity and innovation in the design process, based on related theories from other disciplines. The propositions are, thus, grounded in theory, but not (yet) empirically tested. This type of theory building is suggested specifically when there is no extant theory available (Eisenhardt, 1989; Yin, 2018). This approach is often used for developing design theories. Propositions are usually not tested in the same qualitative context where they are developed, but should be evaluated in subsequent quantitative work (Lee & Baskerville, 2003). Another strategy to develop theory without empirical testing is simulation (Dörner & Wearing, 1995). We argue that theoretical understanding of the method will increase its corroboration.

(6) Copy/Apply. Numerous design methods exist and several books present collections of manifold design methods (Curedale, 2013; J. C. Jones, 1992; Kumar, 2012). The vast availability of design methods can be overwhelming for the designer and present them with the challenge to pick the right method for their respective goals. As a consequence, the way how these methods are applied by designers and design researchers, varies. We can observe intuitive and skilful method application, mainly applied by expert designers on the one hand, and unreflected copying of step-by-step instructions in a "painting-by-numbers" manner on the other, as well as several steps in between. Several authors have studied the impact of expertise level on method usage and application and suggest that typically novice designers rely more on methods than experts, who tend to follow a skilful, intuitive process (Dreyfus & Dreyfus, 1980).

An example for a successful method application, based on empirical evaluation is the story of the Intel semiconductor factories. To maintain the high productivity of specific factories, the company started to copy those successful production sites "piece by piece". Since the reasons for some factories being more efficient than others was not fully understood (the complexity of the system was simply too high), the Copy EXACTLY! Method was developed in the late 1980s. The idea behind this method is to control literally *every* manufacturing variable, including all four levels (physical input, process and equipment, modules, and products), and copy them into a new production site—in exactly the same way (McDonald, 2002). Although the working mechanisms of the original factory's design elements were not theoretically understood, the positive effects could be

replicated. This example presents a *successful* method application, based on copying and underpinned by empirical evaluation.

(7) Adapt. Sometimes an existing method might not be suitable for one's own requirements and context. Then the method needs to be adapted or modified. This can have various reasons. Many design methods originated from the post-World War II era, for example, TRIZ (Altshuller et al., 1997) or the Delphi method (Linstone & Turoff, 1975). Today, the context and requirements have changed. Creativity and innovation are much more team-based, rather than an individual endeavour, as it was considered in the 1950s. Furthermore, new technologies and digital applications allow for new methods and approaches (Thoring et al., 2015). For example, Netnography (Kozinets, 2010) allows to conduct ethnographic research through the Internet and Social Media, rather than observing people in person. Furthermore, individual contexts need to be taken into account. Existing tools might need to be modified in order to address a new target group or to be presented through a different channel. Individual constraints like resources and available time can lead to the necessity to adjust or skip process steps. These circumstances need to be considered carefully in order not to jeopardize the method's impact.

(8) Develop. Designers love to create new things. This is not limited to products and services but also applies to methods. Many designers develop their own methods, and only a few use each other's existing methods. This is sometimes necessary for the following reasons: (a) The existing methods might not fit to the context, (b) the existing methods do not fit to the users, or (c) the context is new so that no methods exist yet. If there is no adequate method available, the researcher might need to develop their own, new method. One example for this scenario is Mosaic (SAP AppHaus, n.d.). Mosaic proposes a method aimed at helping teams prototype their future workspace. This method addresses the gap that there was no systemic approach to developing creative workspaces at that time. Another well-known example is the Business Model Canvas (Osterwalder & Pigneur, 2010), developed to simplify the existing process of writing a business plan and make it accessible to a larger audience. The Idea Arc (Lecuna et al., 2019), a canvas and method for facilitating team-based refinement of ideas, was systematically developed after discovering that no method existed for fleshing-out ideas within interdisciplinary teams. By contrast, we can also observe many methods being developed where actually other solutions already exist. The "canvas collection" (Roberts, 2016) lists more than a hundred canvases, many of them addressing the same problem. Moreover, many of these canvases lack a formal description and do not present the process they followed when developing the tool. Hanington (2003) suggests the need for more "creative" design methods in contrast to "traditional" or "adapted" methods, in order to achieve creative and innovative results. The question of how to develop new design methods is rarely picked up in the literature. Thoring and colleagues present a set of guidelines on how to design canvases (Thoring et al., 2019), and workshops as a research method (Thoring et al., 2020).

We suggest that the previously introduced concepts do not follow a linear sequence but are rather intertwined and influencing each other. Figure 1 illustrates the concepts' interrelationships. We argue

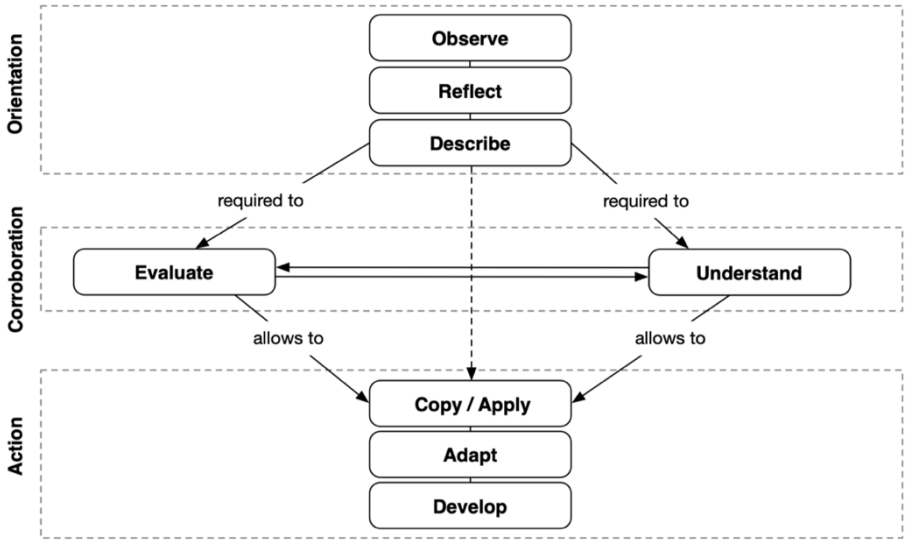

Figure 1. Conceptual model of design method usage, based on literature review

that a precise observation and description of a design method is required to either test and evaluate the method, or to develop a theoretical understanding. Both, evaluating and understanding are influencing each other as well: theoretical understanding allows to properly conduct an evaluation study, whereas the evaluation results can provide insights that lead to theoretical understanding. Both concepts individually allow to copy and apply, and to adapt or design a method. A direct connection between the "orientation" steps and the "action" steps could also skip the corroboration level. This approach might or might not work, but would result in a lack of method corroboration.

Figure 1 outlines the eight identified concepts and their interrelations.

3.2. A framework of method corroboration

While Figure 1 outlines the different concepts involved in design method usage, as well as their interrelationships, the next section goes a step further and looks at the two central concepts of the *corroboration* step—"evaluate" and "understand"—in more detail. We consider both, empirical evaluation and theoretical understanding as the cornerstones of methodology research. However, both dimensions lead to different types of method corroboration. When we combine the two dimensions with their two possible states and juxtapose them with each other, four quadrants emerge naturally, which results in a 2-by-2 matrix of method corroboration (Figure 2).

If an empirical test is missing or unsuccessful and there is no or only limited theoretical understanding, the result would be (1) "method trust"—the application of a method without understanding or testing it. If we do have empirical validation, but still no theoretical understanding, we call the result (2) "method trial-and-error"—the testing of a method without understanding how and why it works. If we have a theoretical understanding but no or unsuccessful validation, this would lead to a preliminary (3) "method grounding"—we understand how and why the method works, but we have not tested it. And only when we have both, theoretical understanding and empirical validation, the result would represent the sweet spot of (4) "full method corroboration". While method corroboration would be the preferred result, we acknowledge that in practice sometimes it might not be feasible to achieve. Hence, we argue that method trial-and-error, as well as method grounding are also justifiable solutions.

To summarize, the presented framework suggests four degrees of design method corroboration, based on combinations of validation and theorizing. We argue that there is a need for a new terminology to be able to refer to these four different granularities of evidence. Hence, we suggest the four terms "method trust", "method grounding", "method trial-and-error", and "full method corroboration" as four degrees of "method corroboration".

Figure 2 outlines the two dimensions of empirical testing and theoretical understanding of a method, along with the four quadrants as outlined above.

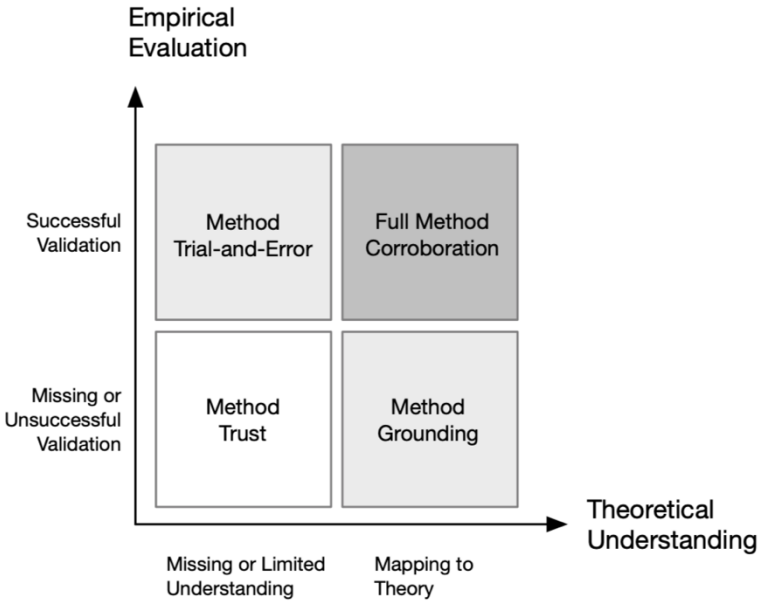

Figure 2. Framework outlining four degrees of method corroboration between empirical evaluation and theoretical understanding

4. Discussion

4.1. Critical Reflection

When looking at the results from the literature review of how design methods are applied or developed in the DSR literature (Table 1), several insights emerge:

(1) Apparently, not many papers "adapt" existing methods (only 2 out of 13), but rather develop an entirely new method (6 out of 13). Moreover, a formal description of developed methods is rarely provided (only 2 out of 13), although this would allow other researchers and practitioners to use and replicate the suggested method. We argue that when adapting an existing method, one should provide a clear description of what we call the *method lineage*, that is, the method's origin or family tree. Explaining precisely what elements from an existing method were used to develop the new/modified one can help to add clarity to the discussion in the design community. Relabelling of methods (that is, giving a new name to an old method), can lead to several problems. In social sciences, two related phenomena are known: "jingle fallacy" (Thorndike, 1904), which refers to two constructs with identical names referring to two different real-world phenomena, and "jangle fallacy" (Kelley, 1927), which refers to two different construct names referring to the same real-world phenomenon. What we can sometimes observe in the design field would qualify as "method identity fallacy", which can be considered a combined "jingle-jangle fallacy" for methods. The reasons for people relabelling methods can be manifold. One possible explanation could be that people are not aware of the same or similar method (a more formalized description would help to avoid or at least to better identify method overlaps and redundancies). Another possible explanation is some sort of "method fatigue". People might feel more intrigued by a new method rather than an old one. Especially in the design thinking area we can observe this sort of saturation and boredom with the term itself. This gives rise to the formation of new concepts and terms. But there could also be a third reason for reinventing and relabelling methods: The academic world, where publication count and h-index are used as key performance indicators, might trigger a behaviour of publishing "old methods in news skins". Coming up with a new method (although it is actually an old one) may lead to "sham productivity".

(2) The relatively frequent occurrences of empirical method validation in the analysed literature (8 out of 13) is astonishing. We assume that this fact can be attributed to the very high-quality standards of the Basket of Eight IS journals. In future work we will also look into IS papers from relevant conferences to investigate whether we will find fewer method evaluations there. Authors who consider submitting their papers to one of the Basket of Eight journals should be aware of this possibly high hurdle. However, we argue that the power of empirical evaluation needs a more critical reflection. For example, the Copy EXACTLY! method by Intel is relatively easy to evaluate. Does the factory produce more defect-free chips or less? In design, measuring the impact of a design method is more complicated: Different design projects are not comparable; results may occur with a time delay (maybe the innovation is not successful at the time of the measurement, but it will be in five years), and case studies usually have a small sample size. When considering these factors, it becomes clear that the theoretical understanding may in fact be much more relevant for design methods than the empirical testing. However, in our analysed sample the theoretical understanding was much lesser present (4 out of 13). This might be attributed to the common academic understanding that empirical testing is usually the silver bullet to achieve robustness and a high design validity.

Finally, we suggest that testing and evaluating methods could go beyond experiments and case studies as outlined in Section 3. New technologies, crowdsourcing, or citizen science could be implemented to bridge the validity gap (Thoring et al., 2015).

(3) Reflection of the used methods was not often explicitly applied in our analysed sample (5 out of 13). When reflecting on the method, one should be patient if the success takes some time. One should be aware that even if the results are not immediately visible, the method might have helped the designer to unconsciously develop ideas or to change their mindset. Again, providing precise and formal descriptions of your experiences and observations and presenting them, for example, as actionable guidelines will help other researchers to replicate them.

(4) Finally, only 3 out of 13 papers (Janiesch et al., 2020, Zhang et al., 2021, and Miah et al., 2019) presented both, empirical evaluation and theoretical understanding of the involved methods, and hence, provide what we call "full method corroboration". By contrast, 4 out of 13 papers did engage neither with empirical evaluation nor with theoretical understanding of the involved methods and, hence, would fall into the category of "method trust". Nevertheless, we argue that not every method needs to be empirically tested or theoretically understood. There is nothing wrong with a little bit of "method trust", if the results are promising and helpful. Trusting a method and following a process can help

with achieving design goals, specifically for novice designers in design education (Dorst & Reymen, 2004), even if the methods are not fully understood.

4.2. Implications and Practical Relevance

The proposed models of method usage and method corroboration provide manifold insights for DSR research and practice.

As outlined in the Copy EXACTLY! example, methods might work also without full corroboration. Sometimes you do not need to understand how something works if you test the successful result empirically. Consequently, we propose different possible paths when aiming for method corroboration. We do not suggest that all eight strategies outlined in Figure 1 must be always followed. Instead, we rather suggest that different paths may lead to different (but still acceptable) degrees of method corroboration. For example, one might focus either on the empirical evaluation of a method, or on the theoretical understanding. Moreover, one could also skip the corroboration phase completely, and go directly to applying, adapting, or developing methods. However, in that case a detailed description of the method to follow is crucial to ensure replicability, as is a critical reflection of the process and outcome.

We acknowledge that in design and IS practice it might not always be feasible to test any given method before applying it, and to check whether it actually works. The proposed framework sheds light on other possible approaches to method corroboration and outlines their advantages and limitations. The different granularities of method corroboration are labelled with specific terms, to allow researchers to refer to them and hence, to add transparency to their method use.

To summarize, the suggested model of design method usage (Figure 1) and the framework of design method corroboration (Figure 2) provide a conceptual guideline for researchers and practitioners who want to use, analyse, understand, evaluate, or develop DSR methods. Both models outline a space of possible strategies towards method corroboration. On the one hand, people can take them as a blueprint when designing methods, and on the other hand, they could use them as a reference for evaluating methods. Finally, both models can be used by reviewers and editors to evaluate the rigor of DSR papers that work with existing or newly developed design methods

5. Conclusions

In this paper, we present a conceptual model of design method usage, based on a literature review within the Basket of Eight IS journals, and a resulting framework of design method corroboration. The conceptual model (Figure 1) suggests eight concepts relevant for design method usage and implementation. The mutual influences and relationships of these concepts are discussed and, thus, answer our first research question (*What is the state of design method corroboration according to the IS literature?)*. It should be noted that we do not suggest that all eight steps are always required to arrive at a corroborated design method. As outlined in the previous sections, some strategies appear to be more crucial than others, and some may even be skipped completely, depending on one's individual focus and expertise.

The resulting framework of method corroboration (Figure 2) addresses our second research question (*How could the corroboration of design methods be increased?*). This framework outlines the influence of empirical evaluation and theoretical understanding to achieve different degrees of method corroboration. The framework suggests four degrees of evidence, based on combinations of validation and theorizing. We argue that there is a need for a new terminology ("method trust", "method grounding", "method trial-and-error", and "full method corroboration") to be able to refer to these four different granularities of method evaluation. As a consequence, with this paper we focus on the detailed investigation of the corroboration layer of our conceptual model, presented in Figure 1. However, we acknowledge that also the other two layers (the "action" and "orientation" layer) warrant further research that will be addressed in future work.

Moreover, in this paper, we focus on the IS discipline, but the framework could also be applied to other (design) disciplines. Thus, future work will investigate the use of method corroboration within literature from the design field.

One limitation of this paper is the small sample size of analysed literature, focusing only on high-quality papers in the basket of eight from the past 10 years. Future work will include the analysis of more papers—including conference papers, but also journal papers from earlier years (to potentially analyse a change of method corroboration over time).

Another limitation of this paper is the question of how the eight concepts for method corroboration could actually be implemented. What we need here is some sort of "meta-methodology" that provides the methods for developing corroborated design methods. This question will be explored in future work.

6. References

Altshuller, G., Shulyak, L., & Rodman, S. (1997). *40 principles: TRIZ keys to innovation* (3593). Technical Innovation Center.

Badke-Schaub, P., & Voute, E. (2018). DESIGN METHODOLOGY: WHERE DO YOU GO? *DS 92: Proceedings of the DESIGN 2018 15th International Design Conference*. https://doi.org/10.21278/idc.2018.0550

Barrett, M., & Holeman, I. (2017). Insights from an ICT4D Initiative in Kenya's Immunization Program: Designing for the Emergence of Sociomaterial Practices. *Journal of the Association for Information Systems*, *18*(12), 900–930. https://doi.org/10.17705/1jais.00476

Brandt, T., Feuerriegel, S., & Neumann, D. (2018). Modeling interferences in information systems design for cyberphysical systems: Insights from a smart grid application. *European Journal of Information Systems*, *27*(2), 207–220. https://doi.org/10.1057/s41303-016-0030-1

Cross, N. (1993). Science and design methodology: A review. *Research in Engineering Design*, *5*(2), 63–69. https://doi.org/10.1007/BF02032575

Cross, N. (2021). *Engineering design methods: Strategies for product design*. John Wiley & Sons.

Curedale, R. (2013). *Design Research Methods: 150 ways to inform design*. Design Community College Inc.

Davison, R., Martinsons, M., & Malaurent, J. (2021). Research Perspectives: Improving Action Research by Integrating Methods. *Journal of the Association for Information Systems*, *22*(3). https://doi.org/10.17705/1jais.00682

Dörner, D., & Wearing, A. J. (1995). Complex problem solving: Toward a (computersimulated) theory. *Complex Problem Solving: The European Perspective*, 65–99.

Dorst, K., & Reymen, I. (2004). Levels of Expertise in Design Education. *Proceedings of E&PDE 2004*. The 7th International Conference on Engineering and Product Design Education, Delft, the Netherlands.

Dreyfus, S. E., & Dreyfus, H. L. (1980). *A five-stage model of the mental activities involved in directed skill acquisition* (Operations Research Center). University of California.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review*, *14*(4), 532–550.

Ericsson, K. A., & Simon, H. A. (1984). *Protocol analysis: Verbal reports as data*. The MIT Press.

Furnham, A. (2000). The Brainstorming Myth. *Business Strategy Review*, *11*(4), 21–28. https://doi.org/10.1111/1467-8616.00154

Gregor, S., & Jones, D. (2007). The anatomy of a design theory. *Journal of the Association for Information Systems*, *8*(5), 312–335.

Hanington, B. (2003). Methods in the making: A perspective on the state of human research in design. *Design Issues*, *19*(4), 9–18.

Hay, L., Duffy, A. H. B., McTeague, C., Pidgeon, L. M., Vuletic, T., & Grealy, M. (2017). A systematic review of protocol studies on conceptual design cognition: Design as search and exploration. *Design Science*, *3*. https://doi.org/10.1017/dsj.2017.11

Hevner, A., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75–105.

Janiesch, C., Rosenkranz, C., & Scholten, U. (2020). An Information Systems Design Theory for Service Network Effects. *Journal of the Association for Information Systems*, *21*(6), 1402–1460. https://doi.org/10.17705/1jais.00642

Jobst, B., Thoring, K., & Badke-Schaub, P. (2020). *Introducing a Tool to Support Reflection Through Sketching and Prototyping During the Design Process*. 207–214. https://doi.org/10.1017/dsd.2020.263

Jones, J. C. (1992). *Design Methods* (2. edition). John Wiley & Sons.

Jones, M. R., & Karsten, H. (2008). Giddens's Structuration Theory and Information Systems Research. *MIS Quarterly*, *32*(1), 127–157. https://doi.org/10.2307/25148831

Kahneman, D. (2013). *Thinking, Fast and Slow* (1st edition). Farrar, Straus and Giroux.

Kannengiesser, U., & Gero, J. S. (2019). Design thinking, fast and slow: A framework for Kahneman's dual-system theory in design. *Design Science*, *5*. https://doi.org/10.1017/dsj.2019.9

Kelley, T. L. (1927). *Interpretation of educational measurements*. World Book Company.

Kolkowska, E., Karlsson, F., & Hedström, K. (2017). Towards analysing the rationale of information security non-compliance: Devising a Value-Based Compliance analysis method. *The Journal of Strategic Information Systems*, *26*(1), 39–57. https://doi.org/10.1016/j.jsis.2016.08.005

Kozinets, R. V. (2010). *Netnography: Doing ethnographic research online*. SAGE.

Kumar, V. (2012). *101 Design Methods: A Structured Approach for Driving Innovation in Your Organization*. Wiley.

Lecuna, A., Thoring, K., & Mueller, R. (2019). The Idea Arc: Designing a Visual Canvas for Fuzzy Ideas. *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 551–560. http://scholarspace.manoa.hawaii.edu/handle/10125/59495

Lee, A. S., & Baskerville, R. L. (2003). Generalizing generalizability in information systems research. *Information Systems Research*, *14*(3), 221–243.

Linstone, H. A., & Turoff, M. (1975). *The delphi method*. Addison-Wesley Reading, MA.

McDonald, C. J. (2002). Copy EXACTLY! A paradigm shift in technology transfer method. *Advanced Semiconductor Manufacturing Conference and Workshop, 1997. IEEE/SEMI*, 414–417.

Miah, S. J., Gammack, J. G., & McKay, J. (2019). A Metadesign Theory for Tailorable Decision Support. *Journal of the Association for Information Systems*, *20*(5), 570–603. https://doi.org/10.17705/1jais.00544

Möller, F., Guggenberger, T. M., & Otto, B. (2020). Towards a method for design principle development in information systems. *International Conference on*

Design Science Research in Information Systems and Technology, 208–220.

Mueller, R. M., Huettemann, S., Larsen, K. R., Yan, S., & Handler, A. (2022). Toward an Information Systems Ontology. *17th International Conference on Design Science Research in Information Systems and Technology (DESRIST)*, 55–67.

Osborn, A. F. (1953). *Applied Imagination: Principles and Procedures of Creative Problemsolving*. Scribeners and Sons.

Osterwalder, A., & Pigneur, Y. (2010). *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*. Wiley.

Payton, F. C. (2016). Cultures of participation—For students, by students. *Information Systems Journal*, *26*(4), 319–338. https://doi.org/10.1111/isj.12086

Pee, L. G., Pan, S. L., Wang, J., & Wu, J. (2021). Designing for the future in the age of pandemics: A future-ready design research (FRDR) process. *European Journal of Information Systems*, *30*(2), 157–175. https://doi.org/10.1080/0960085X.2020.1863751

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, *24*(3), 45–77.

Roberts, A. (2016, December 10). Canvas collection I - A list of visual templates. *Andi Roberts*. http://masterfacilitator.com/canvas-collection/

SAP AppHaus. (n.d.). *SAP Mosaic*. SAP User Experience Design Services. Retrieved May 11, 2018, from https://experience.sap.com/designservices/approach/mosaic

Silsand, L., & Ellingsen, G. (2014). Generification by Translation: Designing Generic Systems in Context of the Local. *Journal of the Association for Information Systems*, *15*(4), 177–196.

Spagnoletti, P., Resca, A., & Sæbø, Ø. (2015). Design for social media engagement: Insights from elderly care assistance. *The Journal of Strategic Information Systems*, *24*(2), 128–145. https://doi.org/10.1016/j.jsis.2015.04.002

Stempfle, J., & Badke-Schaub, P. (2002). Thinking in design teams-an analysis of team communication. *Design Studies*, *23*(5), 473–496.

Thoring, K., Gonçalves, M., Mueller, R. M., Desmet, P., & Badke-Schaub, P. (2021). The Architecture of Creativity: Toward a Causal Theory of Creative Workspace Design. *International Journal of Design*, *15*(2), 17–36.

Thoring, K., Mueller, R. M., & Badke-Schaub, P. (2015). Technology-Supported Design Research. *Proceedings of the 20th International Conference on Engineering Design (ICED 15)*, *11*, 31–40.

Thoring, K., Mueller, R. M., & Badke-Schaub, P. (2019). Exploring the Design Space of Innovation Canvases. *Academy of Design Innovation Management Conference (ADIM)*. Academy of Design Innovation Management Conference (ADIM), London, UK.

Thoring, K., Mueller, R. M., & Badke-Schaub, P. (2020). Workshops as a Research Method: Guidelines for Designing and Evaluating Artifacts Through Workshops. *Proceedings of the 53rd Hawaii International Conference on System Sciences*. HICSS, Maui, HI.

Thoring, K., & Müller, R. M. (2011). Understanding Design Thinking: A Process Model based on Method Engineering. *Proceedings of the 13th International Conference on Engineering and Product Design Education (E&PDE), London, UK*.

Thorndike, E. L. (1904). *An introduction to the theory of mental and social measurements*. Science Press.

Thummadi, B. V., & Lyytinen, K. (2020). How Much Method-in-Use Matters? A Case Study of Agile and Waterfall Software Projects. *Journal of the Association for Information Systems*, *21*(4), 864–900. https://doi.org/10.17705/1jais.00623

Venable, J., & Baskerville, R. (2012). Eating our own cooking: Toward a more rigorous design science of research methods. *Electronic Journal of Business Research Methods*, *10*(2), pp141-153.

Venable, J., Pries-Heje, J., & Baskerville, R. (2016). FEDS: A Framework for Evaluation in Design Science Research. *European Journal of Information Systems*, *25*(1), 77–89.

Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an Information System Design Theory for Vigilant EIS. *Information Systems Research*, *3*(1), 36–59. https://doi.org/10.1287/isre.3.1.36

Welke, R. J., & Kumar, K. (1992). Method Engineering: A proposal for situation-specific methodology construction. In S. Cotterman (Ed.), *Systems Analysis and Design: A Research Agenda* (pp. 257–268). Wiley.

Yang, L., Su, G., & Yuan, H. (2012). Design Principles of Integrated Information Platform for Emergency Responses: The Case of 2008 Beijing Olympic Games. *Information Systems Research*, *23*(3-part-1), 761–786. https://doi.org/10.1287/isre.1110.0387

Yin, R. K. (2018). *Case study research and applications: Design and methods* (6th ed.). Sage Publications Inc.

Zaitsev, A., & Mankinen, S. (2022). Designing financial education applications for development: Applying action design research in Cambodian countryside. *European Journal of Information Systems*, *31*(1), 91–111. https://doi.org/10.1080/0960085X.2021.1978341

Zhang, Z., Yoo, Y., Lyytinen, K., & Lindberg, A. (2021). The Unknowability of Autonomous Tools and the Liminal Experience of Their Use. *Information Systems Research*, *32*(4), 1192–1213. https://doi.org/10.1287/isre.2021.1022