Communications of the Association for Information Systems

Volume 43 Article 17

9-2018

Tool Support for Design Science Research—Towards a Software Ecosystem: A Report from a DESRIST 2017 Workshop

Stefan Morana
Karlsruhe Institute of Technology, stefan.morana@uni-saarland.de

Jan vom Brocke
University of Liechtenstein

Alexander Maedche
Karlsruhe Institute of Technology

Stefan Seidel
University of Liechtenstein

Marc T. P. Adam
The University of Newcastle

See next page for additional authors

Recommended Citation

Morana, S., vom Brocke, J., Maedche, A., Seidel, S., Adam, M. T., Bub, U., Fettke, P., Gau, M., Herwix, A., Mullarkey, M. T., Nguyen, H. D., Sjöström, J., Toreini, P., Wessel, L., & Winter, R. (2018). Tool Support for Design Science Research—Towards a Software Ecosystem: A Report from a DESRIST 2017 Workshop. Communications of the Association for Information Systems, 43, pp-pp. https://doi.org/10.17705/1CAIS.04317

Tool Support for Design Science Research—Towards a Software Ecosystem: A Report from a DESRIST 2017 Workshop

Authors

Stefan Morana, Jan vom Brocke, Alexander Maedche, Stefan Seidel, Marc T. P. Adam, Udo Bub, Peter Fettke, Michael Gau, Alexander Herwix, Matthew T. Mullarkey, Hoang D. Nguyen, Jonas Sjöström, Peyman Toreini, Lauri Wessel, and Robert Winter

Communications of the Association for Information Systems

Workshop Report DOI: 10.17705/1CAIS.04317 ISSN: 1529-3181

Tool Support for Design Science Research—Towards a Software Ecosystem: A Report from a DESRIST 2017 Workshop

Stefan Morana
Institute of Information Systems and Marketing (IISM)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
stefan.morana@kit.edu

Alexander Maedche
Institute of Information Systems and Marketing (IISM)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany

Marc T. P. Adam
School of Electrical Engineering and Computing
The University of Newcastle
Newcastle, Australia

Peter Fettke
German Research Center for Artificial Intelligence (DFKI)
and Saarland University
Saarbruecken, Germany

Alexander Herwix
Professorship of Integrated Information Systems
University of Cologne
Cologne, Germany

Hoang D. Nguyen
Department of Information Systems
National University of Singapore (NUS)
Singapore

Peyman Toreini
Institute of Information Systems and Marketing (IISM)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany

Robert Winter
Institute of Information Management
University of St. Gallen
St. Gallen, Switzerland

Jan vom Brocke
University of Liechtenstein
Institute of Information Systems
Vaduz, Liechtenstein

Stefan Seidel
University of Liechtenstein
Institute of Information Systems
Vaduz, Liechtenstein

Udo Bub
Eötvös Loránd University (ELTE) Budapest,
Faculty of Informatics and
EIT ICT Labs Germany GmbH
Berlin, Germany

Michael Gau
University of Liechtenstein
Institute of Information Systems
Vaduz, Liechtenstein

Matthew T. Mullarkey
MUMA College of Business
University of South Florida
Tampa, USA

Jonas Sjöström
Department of Informatics and Media
Uppsala University
Visby, Sweden

Lauri Wessel
Department of Information Systems
Freie Universität Berlin
Berlin, Germany

Communications of the Association for Information Systems

Workshop Report DOI: 10.17705/1CAIS.04317 ISSN: 1529-3181

Abstract:

The information systems (IS) field contains a rich body of knowledge on approaches, methods, and frameworks that supports researchers in conducting design science research (DSR). It also contains some consensus about the key elements of DSR projects—such as problem identification, design, implementation, evaluation, and abstraction of design knowledge. Still, we lack any commonly accepted tools that address the needs of DSR scholars who seek to structure, manage, and present their projects. Indeed, DSR endeavors, which are often complex and multi-faceted in nature and involve various stakeholders (e.g., researchers, developers, practitioners, and others), require the support that such tools provide. Thus, to investigate the tools that DSR scholars actually need to effectively and efficiently perform their work, we conducted an open workshop with DSR scholars at the 2017 DESRIST conference in Karlsruhe, Germany, to debate 1) the general requirement categories of DSR tool support and 2) the more specific requirements. This paper reports on the results from this workshop. Specifically, we identify nine categories of requirements that fall into the three broad phases (pre-design, design, and post design) and that contribute to a software ecosystem for supporting DSR endeavors.

Keywords: Report, Design Science Research, Open Discussion, Software Ecosystem, Tool Support, Requirements.

This manuscript underwent editorial review. It was received 12/06/2017 and was with the authors for 1 month for 1 revision. Christoph Peters served as Associate Editor.

1 Introduction

Design science research (DSR) focuses on developing knowledge about designing information systems (IS) artifacts (Gregor & Hevner, 2013). In recent years, we have seen important methodological contributions on how to conduct DSR emerge (e.g., Gregor, Müller, & Seidel, 2013; Hevner, March, Park, & Ram, 2004; Kuechler & Vaishnavi, 2012; March & Smith, 1995; Peffers, Tuunanen, Rothenberger, & Chatterjee, 2007; Sein, Henfridsson, Purao, Rossi, & Lindgren, 2011). We have also seen consensus about the outcomes of DSR at different levels of abstraction, completeness, and maturity—including instantiations, nascent design knowledge such as design principles, and full-blown design theories (Gregor & Hevner, 2013). Researchers can readily choose among several approaches to structure, document, and evaluate their research (Gregor & Hevner, 2013; Kuechler & Vaishnavi, 2008; Peffers et al., 2007; Sein et al., 2011; Sonnenberg & vom Brocke, 2012; Venable, Pries-Heje, & Baskerville, 2016), and these approaches all involve multiple, iterative cycles including phases of identifying problems, designing solutions, implementing solutions, evaluating solutions, learning from this process, and communicating the results.

Information systems[1] and information technology[2] artifacts (Lee, Thomas, & Baskerville, 2015) vary in their complexity: from simple concepts to complex systems that are often socio-technical in nature, that are built on functional interdependencies, and that need to be evaluated by different means of naturalistic and other methods (Venable et al., 2016). DSR projects, accordingly, are also complex and involve various stakeholders such as researchers, software developers, and industry practitioners (vom Brocke & Lippe, 2010). DSR scholars face challenges in effectively structuring, managing, and documenting these complex projects. As such, one might ask what support DSR scholars need to overcome these challenges. For instance, one can find many software programs that help someone manage projects, collect qualitative and quantitative data, or analyze qualitative and quantitative data, but we do not know the extent to which these tools meet the requirements of those who operate under a DSR paradigm. Recently, various academic papers have described what tool support DSR researchers need (e.g., Contell, Díaz, & Venable, 2017; Sjöström, 2017; vom Brocke et al., 2017), which highlights the topic's importance. We contend that providing a software ecosystem (Burkard, Widjaja, & Buxmann, 2012) of tools that can be flexibly combined to support the DSR process would benefit DSR scholars and help the field evolve further. At the 2017 DESRIST conference in Karlsruhe, we could fortunately ask DSR scholars from various fields of interests and in various career stages for their opinions and ideas on this issue. The essential question that guided our workshop was: "How can a software ecosystem help researchers to effectively and efficiently conduct design science research projects through appropriate tool support?"[3]. Answering this broad question may help the DSR community improve its productivity and mature further.

This paper proceeds as follows: in Section 2, we describe how we conducted the workshop and how we analyzed the results from the workshop. In Section 3, we then present the results in terms of 1) general requirement categories of DSR tool support and 2) more specific requirements. In Section 4, we discuss our findings, provide an outlook on what these results may mean for the discipline, and conclude the paper.

2 Method

The workshop (announced as an open session) occurred on 31 May, 2017 (see Figure 1). The second and third authors moderated the workshop. In all, 28 DSR scholars who represented different areas of interests and career stages (see Table A1 in Appendix A) participated in the workshop. The workshop lasted for 90 minutes.

[1] Information systems are socio-technical systems that store, process, and disseminate information (Piccoli, 2012).
[2] Information technologies describe the technology component of information systems, typically in terms of digital technologies.
[3] We have iteratively refined the research question has throughout the research and review process; however, it is still consistent with the original question, which simply focused on the tools needed to support DSR endeavors.

OPEN SESSION: ISSUES AND NEEDS FOR TOOL-SUPPORT IN DESIGN SCIENCE RESEARCH (2 P.M.)

DESRIST 3

Design Science research (DSR) is now an accepted research paradigm in the Information Systems field, aiming at developing purposeful IT artifacts and knowledge about the design of IT artifacts. A rich body of knowledge on approaches, methods, and frameworks supports researchers in conducting DSR projects. While methodological guidance is abundant, there is no advanced tool support that helps Design Science researchers structuring and managing their DSR projects and corresponding (intermediate) results. In this interactive session we would like to foster an open discussion on issues and needs of Design Science researchers with respect to providing advanced tool support. Thereby, we ask participants to contribute with their thoughts on potential tool support to structure, conduct, document, maintain, and present DSR projects, including the resulting design knowledge and artifacts. As a result of the session, we plan to publish a panel report including the findings and propose suggestions for research on advancing tool support for DSR.

Figure 1. Announcement of the Session in the Conference Program Brochure

In the workshop, the moderators first gave the participants index cards and asked them to think about what type of support in a tool they thought would help them in conducting their DSR activities. The moderators did not ask for specific tools or types of tools but instructed the participants to freely note all ideas that came to their minds. Second, the moderators asked each participant individually to briefly explain their thoughts to the other workshop participants, and they collected the corresponding index cards on a board that all participants could see. Moderators and workshop participants organized the identified ideas collaboratively into preliminary categories and sought consensus through discussion (see Figure 2). This process resulted in 82 cards organized into key categories (e.g., tools that support collaboration in DSR projects). Third, and after the actual workshop, we undertook a consensus-building process in which we revisited the results from the workshop, compared the categories of requirements for similarities and differences, engaged with existent substantive and methodological contributions in DSR, and constructed a framework of DSR tool support requirements until we reached intersubjective agreement. Note that we expressed requirements for tool support in the sense of both software and methodology (i.e., we did not constrain requirements to only software). Therefore, in this paper, we subsume software tools and methodological tools under the general term tool. In Section 3, we discuss the results from this process.

Figure 2. Impressions from Workshop

3 Results

Through our analysis, we identified three generic categories of requirements for DSR tool support: pre-design, design, and post design. Pre-design involves activities carried out before the actual project commences. The design phase involves all activities related to the design and knowledge extraction based on that design. The post-design phase comprises those activities that follow the actual DSR effort (i.e., mainly activities related to dissemination and impact). We identified nine requirements categories (RC) that fell into these three key stages. Figure 3 overviews the categories and their requirements. Table B1 in Appendix B shows the complete list of categories and requirements and the underlying workshop items.

In Sections 3.1 to 3.3, we describe each requirement category in detail: for each category, we highlight the more specific requirements. Requirements, broadly, are documented representations of a condition or capability that a user needs to solve a problem or achieve an objective (IEEE, 1998)—in our case, efficiently and effectively conducting DSR projects. Importantly, we do not argue that a single tool or platform will—or should—incorporate the identified requirements. Instead, we contend that these requirements will help DSR scholars develop and orchestrate infrastructures that build on digital and non-digital tools that implement these requirements to conduct their work. We also do not claim that the requirements identified are complete.

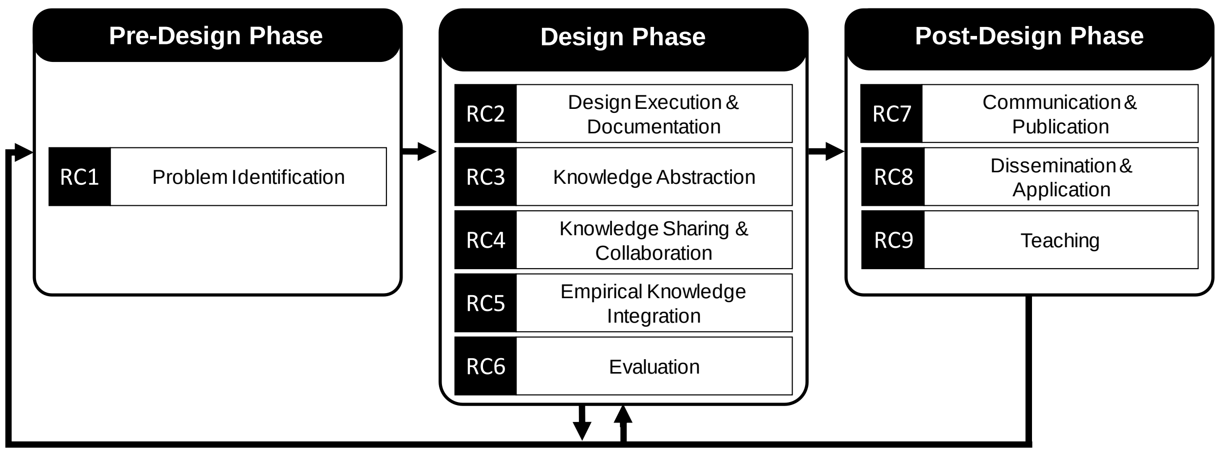

Figure 3. Three Design Phases and Associated Requirement Categories of Tool Support for DSR

3.1 Pre-design Phase

3.1.1 Requirements Category 1: Problem Identification

DSR methodologies, by and large, agree that every DSR project starts with identifying an important problem in a business or other setting (e.g., Gregor & Hevner, 2013; Hevner et al., 2004; Peffers et al., 2007). In congruence with this general understanding, participants highlighted that DSR scholars need support that provides guidance or best practices for describing a problem situation's relevance and impact (e.g., in terms of financial or societal impact) (R1: problem situation description). Different stakeholders hold different expectations, and one can evaluate the value of the outcome of DSR projects in terms of their aesthetic, scholarly, and practical utility (Rai, 2017). Notably, identifying a problem has particular relevance to DSR researchers, because they generally claim to solve real-world problems and, thus, need to be cautious to avoid type 3 errors—errors that occur when one uses the right research method (in our case rigorously conducted DSR) to provide answers to the wrong question (Rai, 2017).

In this line of thought, DSR scholars can benefit from problem libraries that highlight important research questions (R2: problem library) and perhaps even "grand challenges" (Becker, vom Brocke, Heddier, & Seidel, 2015) of high practical and societal relevance. A shared library might motivate the DSR community to embark on these—perhaps even wicked (Nunamaker, Twyman, Giboney, & Briggs, 2017)—problems and to undertake risky research (Gupta, 2017) that might have great impact. Such a problem library should also indicate how existing solutions already provide answers to these challenges and where most needs different levels of contributions (situated implementation of artifacts, nascent design theories, and well-developed design theories) (Gregor & Hevner, 2013). Monitoring the state-of-the-art knowledge in digital technology related to key research challenges can foster dialog among researchers and practitioners, help uncover solutions to existing and important problems, and maintain the library.

Because design science research projects typically involve collaboration (Lindgren, Henfridsson, & Schultze, 2004; vom Brocke & Lippe, 2010), collaboration features can assist research groups and the research community in analyzing problem instances (R3: collaborative problem analysis) and, thereby, support novice and experienced researchers alike (vom Brocke & Lippe, 2013). Table 1 summarizes the requirements related to problem identification.

Table 1. Problem Identification Requirements (RC1)

Requirement	Description
R1: problem situation description	DSR scholars need guidance in formulating research problems and in highlighting their relevance in terms of scholarly and practical utility.
R2: problem library	DSR scholars need access to 1) the state-of-the-art knowledge in DSR and 2) associated key research challenges.
R3: collaborative problem analysis	DSR scholars need to able to collaboratively analyze problem classes.

3.2 Design Phase

3.2.1 Requirements Category 2: Design Execution and Documentation

Once one has identified a problem situation, one needs to think about a feasible solution for it—that is, to define the more specific objectives of (see, e.g., Peffers et al., 2007). In other words, after identifying requirements, one needs to formulate and record them (R4: requirements recording). Such requirements can, for instance, be based on the results from expert interviews, literature reviews, or observations. Researchers widely distinguish between functional and non-functional requirements (e.g., Sommerville, 2016).

Design processes are often complex and messy, and their iterative nature means that one needs to continuously develop and refine models, concepts, and artifacts (e.g., Sein et al., 2011). Documenting this process (R5: design process documentation) ensures scientific rigor and research traceability and provides the foundation for subsequent reporting and publication (vom Brocke et al., 2017). Moreover, the review process may use the documentation (e.g., editors and reviewers might ask for it). Therefore, researchers need to meticulously collect, index, and interrelate notes and to keep record of all activities and relevant details (e.g., related work, references, etc.).

In this line of thought, the iterative nature and development of models, concepts, or artifacts pose challenges on researchers' ability to relate their design decisions and associated reasoning to specific changes (R6: design decision rationale log). From documenting their design decisions and the associated reasoning (e.g., theory-driven or pragmatic decisions), DSR scholars can report on their (potential) knowledge contribution and highlight how this knowledge contribution emerged from the design process. Common standards for documenting and reporting on design processes (i.e., approaches to design anamnesis) can increase the accessibility of one's research and help one abstract away from specific design solutions and move towards more general design knowledge (Gregor & Hevner, 2013).

Because of the iterative nature of DSR processes (e.g., Hevner et al., 2004; Peffers et al., 2007), researchers will, in most cases, work with various versions of their artifact and its functional decomposition, which means they need support to perform artifact versioning (R7: artifact versioning). Information about different versions becomes crucial as researchers seek to report on the temporal development of their research in general and of key design decisions in particular.

While these requirements seem to suggest that DSR should follow certain standards, respondents clearly expressed that such standards could straitjacket DSR scholars by forcing them to follow strict recipes in a process that, by and large, relies on the creativity of involved stakeholders. In other words, the help that such DSR tool support provides should not compromise flexibility in executing and documenting designs (R8: design flexibility and malleability). However, because creativity relies on both convergent and divergent processes (Weisberg, 1999), researchers may also need support in the form of established process frameworks (Kuechler & Vaishnavi, 2008; Peffers et al., 2007; Sein et al., 2011) (R9: supporting convergent and divergent activities). Both convergent and divergent processes co-occur as individuals and groups of individuals envision design outcomes (Müller-Wienbergen, Müller, Seidel, & Becker, 2011).

Table 2 summarizes the requirements related to design execution and documentation.

Table 2. Design Execution and Documentation Requirements (RC2)

Requirement	Description
R4: requirements recording	DSR scholars need assistance in recording requirements that originate from various sources.
R5: process documentation	DSR scholars toned to be able to record the entire process, such as notes, activities, and other relevant information, in detail.
R6: design decision rationale log	DSR scholars need to be able to keep track of the design decisions that they made along the process in (if possible) a standardized way.
R7: artifact versioning	DSR scholars need to keep various versions of the design artifact as it develops over time.
R8: design flexibility and malleability	DSR scholars need to be able to flexibly adapt the design process to their needs.
R9: supporting convergent and divergent activities	DSR scholars need to be able to combine convergent and divergent modes of thinking so they can explore and integrate important ideas.

3.2.2 Requirements Category 3: Knowledge Abstraction

Because DSR research projects should abstract away from specific problem instances and identify more general problem classes (Gregor et al., 2013; Gregor & Hevner, 2013; Sein et al., 2011), workshop participants expressed that tools should support the abstraction process (R10: design knowledge abstraction). This line of thought corresponds to the distinction between abstract and instance domains of design knowledge (Lee, Pries-Heje, Baskerville, 2011) and the argument that abstract design knowledge should be grounded in multiple sources including external theory and empirical data (Goldkuhl, 2004). In documenting strategies for creating, generalizing, and transferring design knowledge in a standardized manner, DSR researchers provide an important source from which they can extract design knowledge and other findings (e.g., Offermann, Blom, & Bub, 2011). The rigorous and detailed documentation of the design process discussed previously provides an important source for extracting design knowledge and other findings (e.g., to aggregate design decisions over multiple iterations). The informational basis through documentation might also provide opportunities for computationally identifying regularities or patterns. Templates for abstract design knowledge can also support DSR scholars. Such abstract knowledge could be either artefacts that are inherently abstract such as design principles (Chandra, Seidel, & Gregor, 2015; Sein et al., 2011) or even full-blown design theory (Gregor & Jones, 2007; Walls, Widmeyer, & El Sawy, 1992, 2004) or artefacts that can be constructed on all levels of generality such as methods or (reference) models. Recent research has started to explore appropriate formats to formulate design knowledge (Chandra et al., 2015; Gregor & Jones, 2007), and tools should support these approaches. For instance, a design theory comprises elements such as purpose and scope, principles of form and function, and justificatory knowledge (Gregor & Jones, 2007). Table 3 summarizes the requirement related to knowledge abstraction.

Table 3. Knowledge Abstraction Requirements (RC3)

Requirement	Description
R10: design knowledge abstraction	DSR scholars need support in abstracting design knowledge (e.g., through commonly accepted templates for formulating design principles or an entire design theory).

3.2.3 Requirements Category 4: Knowledge Sharing and Collaboration

DSR is a collaborative endeavor. Features for collaboration among stakeholders for storing, organizing, publishing, and discussing design knowledge in the design process can help support the inherently collaborative nature of DSR projects. Repositories help organize characteristics of problems and solutions (e.g., based on generality, domain, purpose, etc.); information about the application domain, method, type of project; and the actual artifacts. This informational basis allows researchers to explore and search for related projects with similar characteristics. Making available such information for the broader DSR community will help integrate and highlight the practical impact of DSR and might help scholars to reuse existing design knowledge to create new designs and foster learning. Indeed, digital technologies are characterized by their functional dependencies and unfold their potential through combinatorial effects (Yoo, Boland, Lyytinen, & Majchrzak, 2012). Thus, the DSR community and its future development require collaboration features for organizing, sharing, and discussing design knowledge in research projects and beyond the borders of a particular research project (R11: knowledge sharing & collaboration).

DSR researchers might also have an interest in managing their personal knowledge base (e.g., important literature, data, etc.) and making it accessible for other researchers to stimulate dialog. DSR researchers can, for instance, share important information such as literature reviews, frameworks, data, design process, and methods and might even help visualize emerging knowledge networks and pinpoint potential collaboration opportunities (R12: local DSR knowledge base).

DSR projects cover a broad variety of phenomena in terms of domains, problems, and technologies. Thus, identifying and categorizing keywords and features for topic modeling (Debortoli, Müller, Junglas, & vom Brocke, 2016) help DSR scholars to navigate and define various problem and solution spaces (R13: topic modeling & keyword extraction). Table 4 shows the requirements related to knowledge sharing and collaboration.

Table 4. Knowledge Sharing and Collaboration Requirements (RC4)

Requirement	Description
R11: knowledge sharing and collaboration	DSR scholars need access to a shared, informational basis containing information about problems and solutions, including application domain, method, type of product, and artifacts.
R12: local DSR knowledge base	DSR scholars need to be able to maintain local knowledge bases capturing literature as well as information about artifacts.
R13: topic modeling and keyword extraction	DSR scholars need to be able to extract keywords and do topic modeling.

3.2.4 Requirements Category 5: Empirical Knowledge Integration

Many DSR scholars would agree that theory should inform DSR (Gregor & Jones, 2007; Seidel, Chandra Kruse, Székely, Gau, & Stieger, 2017; Sein et al., 2011), and most DSR methodologies prominently feature the identification of kernel theory or justificatory knowledge. Therefore, DSR tools can help researchers navigate, explore, and select appropriate kernel theory (R14: integration of kernel theory library)—a requirement related to those about sharing knowledge bases. Furthermore, on a finer level of granularity, participants considered the ability to systematically explore existing nomological networks in the particular field of interest as valuable for design science researchers. Therefore, DSR tools should help researchers to search and navigate nomological networks, which includes searching for constructs and relationships between constructs and graph-based visualizations of specific nomological network excerpts (R15: access to nomological networks). Table 5 shows the requirements related to theory integration.

Table 5. Theory Integration Requirements (RC5)

Requirement	Description
R14: integration of kernel theory library	DSR scholars need access to a library with important kernel theory that can inform the design of artifacts.
R15: access to nomological networks	DSR scholars need access to existing nomological networks including constructs and relationships between them.

3.2.5 Requirements Category 6: Evaluation

Evaluation is a key step in every DSR project, and different types of evaluation exist (e.g., observational, analytical, experimental, etc.) (Hevner et al., 2004). One can use evaluation features in the form of, for example, software frameworks or APIs to evaluate artifacts in different stages. When collecting data, for example, one can collect trace data of users' interaction with a system and subject it to further manual and computational analysis (R16: artifact-evaluation frameworks).

Dedicated tools that support researchers in planning, selecting, and properly collecting and analyzing data can help reduce data's complexity through smart filtering, navigation, and visualization (R17: evaluation data collection and analysis), such as tools for conducting laboratory experiments (Hariharan, Adam, Dorner, Lux, Müller, Pfeiffer, & Weinhardt, 2017). DSR projects feature novelty and innovativeness, but one cannot easily assess one's own project's degree of novelty. As such, editors and reviews typically assess novelty with the obvious risks of rejection or, worse, results' being published without sufficient evidence.

Event simulation features can execute an artifact in different environments, and researchers can apply them in situations where they seek to identify and analyze emergent behaviors under varying boundary conditions (R18: event simulation). Hevner et al. (2004) suggest simulation as an important means to evaluate IT artifacts that originate from DSR processes.

DSR scholars need to measure the utility and performance of their artifacts, and measurement instruments with appropriate items for different types of artifacts such as concepts, methods, or design principles can help both authors and reviewers (R19: access to measurement instruments).

Because DSR scholars seek to identify prescriptive knowledge that is assessed in terms of utility rather than theory building or theory testing, they face a key challenge in benchmarking the findings from DSR projects (i.e., two feasible solutions might exist, but what solution is the better solution and under what boundary conditions?). Scheduling and monitoring the progress of the evaluation episodes and managing the associated evaluation data collected during these phases can support the actual evaluation process (R20: benchmarking). Table 6 shows the requirements related to evaluation.

Table 6. Evaluation Requirements (RC6)

Requirement	Description
R16: artifact-evaluation frameworks	DSR scholars need to be able to apply frameworks and APIs to evaluate artifacts in different stages of their lifecycle.
R17: evaluation data collection and analysis	DSR scholars need to be able to plan, select, and perform proper data collection and analysis and allow them reduce data complexity.
R18: event simulation	DSR scholars should be allowed to analyze processes related to the use of the artifact under varying boundary conditions.
R19: access to measurement instruments	DSR scholars need access to measurement instruments that can be applied to different types of artifacts such as instantiations, principles, or entire theories.
R20: benchmarking	DSR scholars need to be able to benchmark their solution with other solutions.

3.3 Post-design Phase Support

3.3.1 Requirements Category 7: Communication and Publication

DSR is an applied field and, as such, focuses on communicating with different stakeholders in both research and practice. Given the field's young age, communication in it continues to focus on cumulatively building a foundational body of design knowledge (related mostly to designing socio-technical artifacts). The community faces challenges in showing how its work contributes to emergent and ongoing debates, and DSR scholars need support in interlinking their substantive and methodological advancements in terms of design knowledge with related publications to strengthen the debate (R21: support publication standards). Some researchers, such as Gregor and Hevner (2013), have already suggested some first steps towards commonly accepted publication standards.

When communicating with stakeholders outside the field, DSR scholars should remember that practitioners use different lexicons and expect them to communicate in readily accessible ways (Te'eni, Seidel, & vom Brocke, 2017). Researchers can use storytelling to communicate their findings and improve the community's engagement with practitioners (R22: storytelling). DSR scholarship is a form of engaged scholarship and, thus, needs to maintain communication with practitioners (Mathiassen & Nielsen, 2008).

DSR scholars need to communicate and critically reflect on design knowledge—novice researchers in particular require guidance on how to identify, summarize, and communicate their findings, and they can benefit from lessons learned about how to conduct DSR in a variety of settings and to use different methods (R23: communicating lessons learned). Table 7 shows the requirements related to supporting the publication of results from DSR.

Table 7. Publication Requirements (RC7)

Requirement	Description
R21: support publication standards	DSR scholars need access to publication standards, much alike in other fields of research such as hypothesis testing research.
R22: storytelling	DSR scholars need support in their efforts of storytelling to engage with stakeholders who are not familiar with scientific jargon.
R23: communicating lessons learned	DSR scholars need to be able to easily communicate methodological lessons learned to the community.

3.3.2 Requirements Category 8: Dissemination and Application

There is an implicit assumption that DSR has high relevance for practitioners. The outcomes should be prescriptive and normative in nature and not propositional (i.e., they are not truth statements). But what is the practical relevance of design knowledge? DSR researchers might benefit from more specific guidelines and best practices on how to communicate the impact of their research—for instance, in terms of the artifact's expected business impact (who should use the artifact under what circumstances to accomplish what result, which practitioners typically measure in terms of business imperatives such as turnover, return on investment, etc.). Notably, DSR scholars should design business impact measurement frameworks in such way that both researchers and practitioners can use and find them useful (R24: business impact measurement framework).

Beyond business impact, DSR should also contribute to societal development, such as documented by the United Nations sustainable development goals[4] (Lee, 2015; Rai et al., 2017; vom Brocke, Watson, Dwyer, Elliot, & Melville, 2013; Watson, Boudreau, & Chen, 2010). DSR tools should provide societal impact measurement frameworks to complement business impact measurement frameworks (R25: societal impact measurement framework). Table 8 shows the requirements related to the communication of impact on business and society.

Table 8. Societal and Business Impact Requirements (RC8)

Requirement	Description
R24: business impact measurement framework	DSR scholars and practitioners need support to assess the business impact of artifacts.
R25: societal impact measurement framework	DSR scholars and practitioners need support to assess the societal impact of artifacts.

3.3.3 Requirements Category 9: Teaching

The DSR community has seen an increased public interest in knowledge on how to apply design-oriented research on digital technologies in a broad variety of domains. Standards, reference models, and templates on how to teach DSR are scarce (see, e.g., Hevner and Chatterjee, 2015)—and those who supervise PhD candidates would benefit from support for deriving DSR teaching cases and for publishing these cases using standard templates (R26: creation support for DSR case studies).

Contemporary forms of teaching such as podcasts or tutorials provide ample fodder for improving DSR teaching and reaching a broader community. DSR scholars can capture artifacts and other DSR deliverables in a seamless way from tools that offer such a thing (e.g., by embedded ScreenCam recordings) (R27: capture artifacts and other DSR deliverables for innovative teaching formats). Table 9 shows the requirements related to teaching DSR.

[4] http://www.un.org/sustainabledevelopment/sustainable-development-goals/

Table 9. DSR Teaching Requirements (RC9)

Requirement	Description
R26: creation support for DSR case studies	DSR scholars need access to standards, reference models, and templates that help them derive teaching cases from their DSR projects.
R27: capture artifacts and other DSR deliverables for innovative teaching formats	DSR scholars toned to be able to easily embed the outcomes of their DSR endeavors into tutorials.

4 Discussion and Conclusion

The workshop highlighted how tools can support DSR scholars in performing various activities related to preparing, executing, and communicating DSR. Together, this report paints a nearly comprehensive picture of requirements for tools that might be loosely coupled to build infrastructures and a software ecosystem that can support DSR projects, which are typically complex, multi-faceted, and multi-layered projects that involve different stakeholders who use different lexicons and have different incentive systems, motivations, and expectations. Specifically, through our analysis, we identified several requirements for DSR tool support (both software and methodological), and we grouped them into nine different categories. This paper describes parts of a DSR process in itself: after identifying a problem (i.e., the lack of sufficient tool support to conduct DSR projects), we formulate requirements for potential solutions, which are grounded in the workshop results and engagement with prior literature.

The requirements we identified in our workshop highlight that DSR scholars think in terms of common actions: identifying problems, designing solutions, implementing solutions, evaluating solutions, learning from this process, and communicating the results. While this might be interpreted as a self-fulfilling prophecy, one can also argue that this observation indicates the maturation of DSR as a research paradigm in the IS field. DSR scholars no longer have to defend their method per se—they have to conduct it rigorously, and we contend that an ecosystem of appropriate (digital) tools can support them in doing so. The requirements that we identified through our workshop might give some pointers.

In a next step, others might move our work forward by 1) identifying what tools are currently available that meet (parts of) the requirements we identify, 2) implementing appropriate solutions, and 3) abstracting away from those solutions and the gathered empirical evidence to provide more general design principles about tools that support DSR projects. These design principles might indeed fall into categories of problem identification, design execution and documentation, knowledge abstraction, theory integration, evaluation, publication support, societal impact, and teaching.

We hope that the results from our workshop stimulate a debate on what support DSR needs and that they contribute to how effectively and efficiently scholars conduct DSR projects; thus, we hope they contribute to maturing this research field at the intersection of IS, computer science, design, and many other fields. We, as a community of DSR researchers, have to move beyond paradigmatic and epistemological debates, show the value of our methods through meaningful applications, and communicate effectively with practitioners. As a community committed to improving our understanding of the design of digital artifacts, we should advance our own understanding of how digital artifacts can make our own work as productive as possible in order to realize DSR's full potential.

References

Becker, J., vom Brocke, J., Heddier, M., & Seidel, S. (2015). In search of information systems (grand) challenges: A community of inquirers perspective. *Business & Information Systems Engineering*, *57*(6), 377-390.

Burkard, C., Widjaja, T., & Buxmann, P. (2012). Software ecosystems. *Business & Information Systems Engineering*, *4*(1), 41-44.

Chandra, L., Seidel, S., & Gregor, S. (2015). Prescriptive knowledge in IS research: Conceptualizing design principles in terms of materiality, action, and boundary conditions. In *Proceedings of the 48th Hawaii International Conference on System Sciences* (pp. 4039-4048).

Contell, J. P., Díaz, O., & Venable, J. R. (2017). DScaffolding: A tool to support learning and conducting design science research. In A. Mädche, J. vom Brocke, & A. R. Hevner (Eds.), *Proceedings of the 12th International Conference on Designing the Digital Transformation* (LNCS vol. 10243, pp. 441-446). Cham: Springer.

Debortoli, S., Müller, O., Junglas, I., & vom Brocke, J. (2016). Text mining for information systems researchers: An annotated topic modeling tutorial. *Communications of the Association for Information Systems*, *39*, 110-135.

Goldkuhl, G. (2004). Design theories in information systems—a need for multi-grounding. *Journal of Information Technology Theory and Application*, *6*(2), 59-72.

Gregor, S., & Hevner, A. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly*, *37*(2), 337-355.

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

Gregor, S., Müller, O., & Seidel, S. (2013). Reflection, abstraction, and theorizing in design and development research. In *Proceedings of the 21st European Conference on Information Systems.*

Gupta, A. (2017). Editorial thoughts: What and how ISR publishes. *Information Systems* Research, *28*(1), 1-4.

Hariharan, A., Adam, M. T. P., Dorner, V., Lux, E., Müller, M. B., Pfeiffer, J., & Weinhardt, C. (2017). Brownie: A platform for conducting neuroIS experiments. *Journal of the Association for Information Systems*, *18*(4), 264-296.

Hevner, A., & Chatterjee, S. (2015). *Design science research in information systems*. Retrieved from http://eduglopedia.org/reference-syllabus/AIS_Reference_Syllabus_Design_Science_Research_in_IS.pdf

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

IEEE. (1998). Recommended practice for software requirements specifications (IEEE std. 830-1998). Retrieved from https://doi.org/10.1109/IEEESTD.1998.88286

Kuechler, B., & Vaishnavi, V. (2008). Theory development in design science research: Anatomy of a research project. *European Journal of Information Systems*, *17*(5), 489-504.

Kuechler, W. L., & Vaishnavi, V. (2012). A framework for theory development in design science research: Multiple perspectives. *Journal of the Association for Information Systems*, 13(6), 395-423.

Lee, A. S., Thomas, M., & Baskerville, R. L. (2015). Going back to basics in design science: From the information technology artifact to the information systems artifact. *Information Systems Journal*, *25*(1), 5-21.

Lee, J. K. (2015). Research framework for AIS grand vision of the bright ICT initiative. *MIS Quarterly*, *39*(2), iii-xii.

Lee, J. S., Pries-Heje, J., & Baskerville, R. (2011). Theorizing in design science research. In H. Jain, A. P. Sinha, & P. Vitharana (Eds.), *Proceedings of the 6th International Conference on Service-oriented Perspectives in Design Science Research* (LNCS vol. 6629, pp. 1-16). Berlin: Springer.

Lindgren, R., Henfridsson, O., & Schultze, U. (2004). Design principles for competence management systems: A synthesis of an action research study. *MIS Quarterly*, 28(3), 435-472.

March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, *15*(4), 251-266.

Mathiassen, L., & Nielsen, P. A. (2008). Engaged scholarship in IS research. *Scandinavian Journal of Information Systems*, *20*(2), 3-20.

Müller-Wienbergen, F., Müller, O., Seidel, S., & Becker, J. (2011). Leaving the beaten tracks in creative work—a design theory for systems that support convergent and divergent thinking. *Journal of the Association for Information Systems*, *12*(11), 714-740.

Nunamaker, J. F., Twyman, N. W., Giboney, J. S., & Briggs, R. O. (2017). Creating high-value real-world impact through systematic programs of research. *MIS Quarterly*, *41*(2), 335-351.

Offermann, P., Blom, S., & Bub, U. (2011). Strategies for creating, generalising and transferring design science knowledge—a methodological discussion and case analysis. In *Proceedings of the 10th International Conference on Wirtschaftsinformatik* (pp. 1187-1196).

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.

Piccoli, G. (2012). *Information systems for managers: Text & cases* (2nd ed.). Hoboken, NJ: Wiley.

Rai, A. (2017). Avoiding type III errors: Formulating IS research problems that matter. *MIS Quarterly*, 41(2), iii-vii.

Rai, A., Burton-Jones, A., Chen, H., Gupta, A., Hevner, A. R., Ketter, W., Sarker, S., & Yoo, Y. (2017). *Diversity of design science research*. *MIS Quarterly*, *41*(1), iii-xviiI.

Seidel, S., Chandra Kruse, L., Székely, N., Gau, M., & Stieger, D. (2017). Design principles for sensemaking support systems in environmental sustainability transformations. *European Journal of Information Systems*, *20*(2), 1-26.

Sein, M. K., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action design research. *MIS Quarterly*, 35(1), 37-56.

Sjöström, J. (2017). DeProX: A design process exploration tool. In A. Mädche, J. vom Brocke, & A. R. Hevner (Eds.), *Proceedings of the 12th International Conference on Designing the Digital Transformation* (LNCS, vol. 10243, pp. 447-451). Cham: Springer.

Sommerville, I. (2016). *Software engineering* (10th ed.). New York, NY: Pearson.

Sonnenberg, C., & vom Brocke, J. (2012). Evaluations in the science of the artificial—reconsidering the build-evaluate pattern in design science research. In In K. Peffers, M. Rothenberger, & B. Kuechler (Eds.), *Proceedings of the International Conference on Design Science Research in Information Systems* (LNCS vol. 7286, pp. 381-397). Berlin, Heidelberg: Springer.

Te'eni, D., Seidel, S., & vom Brocke, J. (2017). Stimulating dialog between information systems research and practice. *European Journal of Information Systems*, *26*(6),

Venable, J., Pries-Heje, J., & Baskerville, R. (2016). FEDS: A framework for evaluation in design science Research. *European Journal of Information Systems*, *25*, 77-89.

vom Brocke, J., Fettke, P., Gau, M., Houy, C., Maedche, A., Morana, S., & Seidel, S. (2017). Tool-support for design science research: Design principles and instantiation. *SSRN Electronic Journal*. Retrieved from https://doi.org/10.2139/ssrn.2972803

vom Brocke, J., & Lippe, S. (2010). Taking a project management perspective on design science research. In R. Winter, J. L. Zhao, & S. Aier (Eds.), *Proceedings of the 5th International Conference on Global Perspectives on Design Science Research* (LNCS vol. 6105, pp. 31-44). Berlin: Springer.

vom Brocke, J., & Lippe, S. (2013). Identifying and managing creative tasks in collaborative is research projects. *Project Management Journal*, *44*(6), 94-113.

vom Brocke, J., Watson, R. T., Dwyer, C., Elliot, S., & Melville, N. (2013). Green information systems: Directives for the IS discipline. *Communications of the Association for Information Systems*, *33*, 509-520.

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.

Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (2004). Assessing information system design theory in perspective: How useful was our 1992 initial rendition. *Journal of Information Technology Theory and Application*, *6*(2), 43-58.

Watson, R. T., Boudreau, M.-C., & Chen, A. J. (2010). Information systems and environmentally sustainable development: Energy Informatics and new directions for the IS community. *MIS Quarterly*, *34*(1), 23-38.

Weisberg, R. W. (1999). Creativity and knowledge: A challenge to theories. In R. J. Sternberg (Ed.), *Handbook of creativity* (pp. 226-250). Cambridge: Cambridge University Press.

Yoo, Y., Boland, R. J., Lyytinen, K., & Majchrzak, A. (2012). Organizing for innovation in the digitized world. *Organization Science*, *23*(5), 1398-1408.

Appendix A: List of Participants

In total, 28 scholars and two moderators (Jan vom Brocke and Alexander Maedche) participated in the open innovation session. The following Table A1 lists all scholars who agreed to publish their names.

Table A1. List of Open Session Participants

Scholar	Affiliation
Marc T. P. Adam	The University of Newcastle, Australia
Udo Bob	EIT ICT Labs Germany GmbH
Jan vom Brocke	University of Liechtenstein, Liechtenstein
Oscar Diaz	University of the Basque Country San Sebastián, Spain
Peter Fettke	German Research Center for Artificial Intelligence (DFKI) and Saarland University, Germany
Michael Gau	University of Liechtenstein, Liechtenstein
Amir Haj-Bolouri	University West, Sweden
Alexander Herwix	University of Cologne, Germany
Alan Hevner	University of South Florida, USA
Giovanni Maccani	Maynooth University / Intel, Ireland
Alexander Maedche	Karlsruhe Institute of Technology, Germany
Stefan Morana	Karlsruhe Institute of Technology, Germany
Matthew Mullarkey	University of South Florida, USA
Duy Hoang Nguyen	National University of Singapore, Singapore
Hannes Rothe	Freie Universität Berlin, Germany
Raphael Schilling	University of St. Gallen, Switzerland
Stefan Seidel	University of Liechtenstein, Liechtenstein
Dominik Siemon	Braunschweig University of Technology, Germany
Jonas Sjöström	Uppsala University, Sweden
Timo Strohmann	Braunschweig University of Technology, Germany
Peyman Toreini	Karlsruhe Institute of Technology, Germany
Monica Chiarini Tremblay	William and Mary, USA
Lauri Wessel	Freie Universität Berlin, Germany
Robert Winter	University of St. Gallen, Switzerland

Appendix B: List of Requirements

Table B1 contains all items we collected during the workshop and how we mapped them to the categories and requirements. Please note that, during the analysis and discussion after the workshop, three additional requirements (15, 24, and 25) emerged that no workshop item supported; hence, we do not list them in this table.

Table B1. List of Workshops Items, Category, Requirement, and Requirement ID

Category	Requirement	ID	Workshop Item
Problem identification requirements (RC1)	Problem situation description	R1	Detailing the relevance & impact of problems (possibility in financial terms)
	Problem library	R2	Convert a problem instance to class of problems
			Monitoring of new (open) technologies that can be adopted to existing problems/theories easily
	Collaborative problem analysis	R3	Cause-(problem) prioritization/problem library
			Open research questions
			Collaborative analyses of the problem
			Paper & expert recommendation based on collected papers
Design execution & documentation requirements (RC2)	Requirements recording	R4	Requirements analysis techniques
			Tool to capture quality requirements (security, performance, …) and make them computational
	Process documentation	R5	Collect & interrelate notes/ thoughts (maybe with keywords indexing)
			Activity log
			Process documentation
			Document research design/ activities with required details (e.g., related references, etc.)
			Tracing changes in the design idea
			Visualizing the DSR survey for publication/understanding of research
			Documentation of the iterative development/refinement of models/concepts/… (even within a specific design stage) and the reasoning that has led to changes/extensions)
			Quality and maturity management of DSR
			"Good" standardization of design and reporting guidelines & formats
			Documenting knowledge contributions throughout the design process
			Documentation of how knowledge contributions were developed
			Documenting DSR process for review/ publication
			Providing documentation for review processes (anonymized)
			Documenting and tracing back decisions about design
	Design decision rationale log	R6	Technique for design anamnesis (i.e., documentation of design choices for justification purposes)
			Technique to rationalize DSR from logs (e.g., to aggregate design decisions over multiple iterations into a standard documentation)
			Identifying regularities/ patterns
			Technique to identify (outcome/ process) patterns from DSR process logs
	Artifact versioning	R7	Artifact versioning
			Github for non-instantiations; sharing versions of my artifacts (models: process, procedures) with peers on a public repository joint model design (with commit notes on reasons for every change)
	Design flexibility and malleability	R8	Anti-use case: forcing others to follow strict recipes
	Supporting convergent and divergent activities	R9	Should support DSR processes (such as Peffers et al., 2007); should support both divergent (creativity) steps and convergent (decision) steps
			Design thinking toolbox
Knowledge abstraction requirements (RC3)	Design knowledge abstraction	R10	Pattern mining and recommendation
			Conceptualizing design knowledge
			Knowledge extraction/ coding DSR process documentation
			How to formulate design principles as DSR outcomes that are grounded in situational design and extract theory (i.e. linking together: design data, design principles, extant theory)

Table B1. List of Workshops Items, Category, Requirement, and Requirement ID

Knowledge sharing and collaboration requirements (RC4)	Knowledge sharing and collaboration	R11	Repository to organize problem/solution characteristics (e.g., based on generality, domain, purpose) that affords to identify related DSR work
			Making available design knowledge
			Community knowledge
			Database of earlier projects: application domain, method, type of project (more exploratory vs. established domain)
			Suggest related projects
			Combinator re-using previous designs (to come to new designs)
			Learning by/from example (i.e., take existing DSR process documentations as examples)
			Repository of artifacts from DSR community, goal: integrate and showcase DSR impact to practice and to specific domains (e.g., health)
			Reuse of design artifacts
			Managing the “data lake” of iterative DSR—documenting various data sources, types that are integrated into a joint database
			Creativity support
			Brainstorming (no mediation)
			Assuming and changing roles (lead, leads)
			Collaboration
			Being able to assign papers, repost, post etc. from different resources to project and shared platform with features (like pocket application)
			Manage stakeholder: use CRM to track contacts, communicate with contacts, define goals, organize workshops, gather feedback from photos, share presentation and results, collect feedback
			Conduct design activities with researchers/ practitioners online (e.g., refinement of a model/ concepts/…) + documentation
			Open-source development -> platform for artefacts; share experience and artefacts themselves
	Local DSR knowledge base	R12	Personal knowledge base
			Knowledge base
			Expertise matching
			Knowledge networks with collaborative features & visualization
			Share data/process
			Information sharing: literature reviews, frameworks, methods
	Topic modeling and keyword extraction	R13	Topic modeling and keyword extraction
Empirical knowledge integration requirements (RC5)	Integration of kernel theory library	R14	How to aid the design rational process to be theory informed; the integration of design rational thinking (QDC) with the idea of theory-ingrained artifact (ADR)
			Kernel theory identification
			relating problem/needs to relevant theory
Evaluation requirements (RC 6)	Artifact-evaluation frameworks	R16	Evaluation components/API that can be integrated into software to support artifact evaluation/ data collection
			A/B experimenting with non-instantiations (i.e., models) on the Web
	Evaluation data collection & analysis	R17	Analysis of evaluation data, evaluation techniques
			Evaluation facilities: shake ideas and let them be discussed by peers
			Support data analysis (large amounts of data—reduce complexity through smart filtering, navigation, and visualization)
	Event simulation	R18	Have an event simulation tool that exercises an artifact under different environment; funds emergent behaviors across environment
	Access to measurement instruments	R19	Behavior computation (software) tool—identifies all behaviors in software artifacts
			Evaluation of the innovativeness of the artifact
	Benchmarking	R20	Benchmarking (platform)
			Scheduling & monitoring progress
Publication support (RC7)	Support publication standards	R21	Agreed upon publication recommendations for design science research
			Personas; types of DSR
	Storytelling	R22	Storytelling if the knowledge can't be formalized
	Communicating lessons learned	R23	Assisting reflection to formulate lessons learned
			Putting a financial value on the artifact
			Should be useful for researchers & practitioners
DSR teaching requirements (RC9)	Creation support for DSR case studies	R26	It would be great to publish just D&R cases in a DSR case; have a standard template; detail on DSR projects
			Teaching cases for DSR
	Capture artifacts and other DSR deliverables for innovative teaching formats	R27	Modern teaching forms: podcasts, tutorials

About the Authors

Stefan Morana is a Postdoctoral Researcher in Information Systems at the Karlsruhe Institute of Technology (KIT). He received a Bachelor and Master degree in Computer Science from the University of Applied Sciences in Darmstadt and a PhD in Economics from the University of Mannheim. His research focuses on the design of interactive systems and methodological aspects of design science research. More specifically, he investigates the design of assistant systems as well as conversational interfaces supporting the individual usage of information systems. His research is published in the *Journal of the Association for Information Systems*, *Decision Support Systems*, and *Business & Information Systems Engineering*, and conferences such as ICIS, ECIS, and DESRIST. He was proceedings chair and member of the organization committee at DESRIST 2017.

Jan vom Brocke is Professor of Information Systems, the Hilti Chair of Business Process Management, Director of the Institute of Information Systems, and Vice President Research and Innovation at the University of Liechtenstein. His research focuses on business process management and related aspects of digital innovation and transformation. He has published, among others, in *MIS Quarterly*, *Journal of Management Information Systems*, *Journal of Information Technology*, *European Journal of Information Systems*, *Information Systems Journal*, *Communications of the ACM*, and *MIT Sloan Management Review*. He has held various editorial roles and leadership positions in Information Systems research and education.

Alexander Maedche is a Full Professor at the Karlsruhe Institute of Technology (KIT) and head of the research group "Information Systems & Service Design" at the Institute of Information Systems and Marketing (IISM) and the Karlsruhe Service Research Institute (KSRI). His research work focuses on designing interactive intelligent systems. His work has been published in leading international journals such as the *Journal of the Association of Information Systems*, *Business Process Management Journal*, *Information & Software Technology*, *IEEE Intelligent Systems*, *SIGMOD Record*, and *AI Magazine*.

Stefan Seidel is Professor and Chair of Information Systems and Innovation at the Institute of Information Systems at the University of Liechtenstein. His research explores the role of digital technologies in creating organizational, social, and environmental innovation and change. Moreover, he is interested in philosophical and methodological aspects of building theory and conducting impactful research. His work has been published or is forthcoming in leading academic journals, including *MIS Quarterly*, *Information Systems Research*, *Journal of Management Information Systems*, and several others. He is an Associate Editor to *Information Systems Journal* and Past Chair of the Special Interest Group on Green Information Systems within the Association for Information Systems.

Marc T. P. Adam is a Senior Lecturer in Computing and Information Technology at the University of Newcastle, Australia. In his research, he investigates the interplay of cognitive and affective processes of human users in electronic commerce and mobile health. He received an undergraduate degree in Computer Science from the University of Applied Sciences Würzburg, Germany, and a PhD in Economics of Information Systems from the Karlsruhe Institute of Technology, Germany. His research has been published in top international outlets such as *International Journal of Electronic Commerce*, *Journal of Management Information Systems*, *Journal of the Association for Information Systems*, *Journal of Retailing*, *Journal of Interactive Marketing*, and *Economics Letters*.

Udo Bub is Associate Professor of Innovation and Information Systems Engineering at the Faculty of Informatics at Eötvös Loránd University (ELTE) in Budapest, Hungary. Before, he had held senior management positions (CEO, Vice President or Director) at research and innovation institutions like EIT ICT Labs Germany, Deutsche Telekom Innovation Laboratories, and EIT Digital. His research interests are Design Science Research in the context of Innovation Engineering and Software Engineering. Furthermore, he is active in the field of Artificial Intelligence and Machine Learning where he is also author/co-author of international scientific publications and patents with a focus on Speech Recognition. Udo has received both his Master's and PhD degrees in Electrical and Computer Engineering from TU Munich, after which he had spent a long-term research appointment at Carnegie Mellon University's School of Computer Science. He had received a habilitation degree in Computer Science from ELTE.

Peter Fettke is Professor of Business Informatics at Saarland University and Principal Researcher, Research Fellow and Research Group Leader at the German Research Center for Artificial Intelligence (DFKI) in Saarbrücken, both Germany. In his research, Peter and his 30-member research group focus on the intersection between process management and artificial intelligence (AI). In addition to investigating the possible applications of AI technologies such as deep learning and process mining in an operational context, his work focuses on researching the effects of digitization on business model innovations and modelling business processes. Peter is the author of more than 100 publications. Some of his publications are the most cited papers in leading international journals on business informatics and he is one of the top 10 most cited scientists at the DFKI. He is also a well-recognized reviewer for renowned conferences, journals, and research organizations worldwide.

Michael Gau received his degree in information and communication engineering from the Vorarlberg University of Applied Sciences, Austria. He works as a research software engineer at the Institute of Information Systems at the University of Liechtenstein.

Alexander Herwix is a research assistant and PhD student at the Professorship of Integrated Information Systems, University of Cologne. His dissertation project focuses on the development of foundations for tool-support of design science-related research areas (e.g., information system, education, etc.). His research interests are design science research and its foundations, business process management, knowledge management, and educational technology. Alexander has published in proceedings of international conferences (e.g., DESRIST).

Matthew T. Mullarkey is the Director of the Doctor of Business Administration (DBA) Program for the USF Muma College of Business and an Instructor in the Information Systems and Decision Sciences (ISDS) Department. Mullarkey received a PhD in Business Administration with a concentration in Information Systems from the University of South Florida. His areas of research interest include design science research and action research in the evaluation and creation of information systems, healthcare systems, social networking systems, and data science. He teaches using case based discussions that engage Doctoral, MBA and MIS undergraduates in complex business decisions that typically involve a technology component. He is the Editor-in-Chief of the USF *Muma Case Review* and an editor and reviewer for a number of academic journals.

Hoang D. Nguyen is an associated researcher at the Department of Information Systems and Analytics at the National University of Singapore. He holds a BSc in Information Systems and a PhD in Information Systems and Analytics from the National University of Singapore. His research interests include health optimisation, big data analytics, and mobile interventions. He focused on creating design artefacts for patients and healthcare professionals to obtain best possible outcomes informed by the current best evidence and patients' values. His work has been published in *Journal of Decision Systems*, *Communications of the Association for Information Systems*, and *Health and Quality of Life Outcomes*.

Jonas Sjöström is a senior lecturer in information systems in the Informatics and Media Department at Uppsala University—Campus Gotland. His research interests include software process improvement, research-based information systems education, information systems evaluation methodology, and privacy management in organizations. He has 15 years of design science research experience and engaged scholarship, primarily in the context of e-government and e-health. His work also includes the design of DeProx—a software tool to facilitate retrospective analysis of code evolution and social interaction in design processes. He holds a PhD in computer science in intersection with social sciences from Uppsala University.

Peyman Toreini is a research assistant and PhD student at the Institute of Information Systems and Marketing (IISM) at the Karlsruhe Institute of Technology (KIT). His research focuses on the intersection between human-computer interaction and business intelligence and analytics (BI&A) systems applying design science research methodology. Particularly, he is designing attention-aware BI&A dashboards by integrating eye-tracking technologies.

Lauri Wessel is an assistant professor of Information Systems and Organization at the Freie Universität Berlin, Germany. His research is focused on understanding and designing digital health management. Recently he has worked on assistive technologies for caring for persons with dementia at home and the role of ICT for managing novel organizational forms in health care. Lauri is an Associate Editor for the *Communications of the Association for Information Systems* and sits on the editorial board of *Information & Organization*.

Robert Winter is full Professor of Business & Information Systems Engineering at the University of St. Gallen (HSG) and Director of HSG's Institute of Information Management. He was founding Academic Director of HSG's Executive Master of Business Engineering program, Academic Director of HSG's PhD in Management program. He served as vice Editor-in-chief of the Business & Information Systems

Engineering journal and Senior Associate Editor of *European Journal of Information Systems*. He is currently serving on the editorial board of *MIS Quarterly Executive*. His research interests include design science research methodology, enterprise architecture management and the governance of very large IT projects/programs.