8-6-2011

Understanding Artifact Knowledge in Design Science: Prototypes and Products as Knowledge Repositories

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

Katja Thoring
Anhalt University of Applied Sciences, k.thoring@design.hs-anhalt.de

Understanding Artifact Knowledge in Design Science: Prototypes and Products as Knowledge Repositories

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

Katja Thoring
Anhalt University of Applied Sciences, Dessau, Germany
k.thoring@design.hs-anhalt.de

ABSTRACT

This article analyzes the role of artifacts in Information Systems regarding their capability to store and transfer design knowledge and to support the researcher-practitioner collaboration. A definition for design artifacts and their characteristics is presented that distinguishes between prototypes and products. Based on three examples for artifacts from the field of Information Systems, deficits concerning the accessibility of artifacts are identified, as well as problems of unintentional embedding of knowledge into artifacts—both results in a 'design science dilemma'. A strategy of 'open artifacts' is suggested, since this might support the collaboration between authors of artifacts (practitioners) and researchers who may want to extract and analyze the embedded knowledge. The work in this paper contributes to a better understanding of knowledge transfer mechanisms in design science and can be used as a foundation for further research in the development of artifact-based knowledge transfer and re-engineering.

Keywords

Design Science, Knowledge Management, Knowledge Transfer, IS Artifacts, Embodied Knowledge.

INTRODUCTION

Even though artifacts are the central outcome of design science research and practice, the capability of storing and transferring knowledge through these artifacts is not well understood so far. Artifacts play an important role in scientific and educational contexts, since it is possible to extract the 'frozen' knowledge from the artifact. Therefore, it is possible to support collaborative knowledge exchange between researchers and practitioners. This article analyzes the role of artifacts in design science. The first section discusses related work about artifacts in design science research and presents a typology of design knowledge (Müller et al. 2010), which suggests four different types of design knowledge. According to this typology, design knowledge can be represented in artifacts, as tacit gut feeling, as codified knowledge, or as scientific theories. We focus on the first level, *design artifacts*, which describes artifacts as 3-dimensional forms and signals of an instantiation of a design. The main section of this article discusses the concept of artifacts in general and suggests differentiating between two types of artifacts—prototypes and products. To illustrate further, three examples of different artifacts from the field of Information Systems are presented, and their capabilities to store and to transfer design knowledge in design science is analyzed. The last section continues by discussing the resulting consequences and suggesting a strategy of 'open artifacts' to allow for better knowledge exchange between practitioners and researchers. The work presented in this article can be considered as a theory for analyzing (Type I theory) according to (Gregor 2006).

RELATED WORK

This section discusses different classifications of design knowledge in general, and artifact knowledge in particular, before we continue by applying these to the field of Information Systems in the next section.

Typology of Design Knowledge According to Müller & Thoring

According to a typology of design knowledge from Müller and Thoring (2010), which is adapted from Radermacher (1996), there exist four different types of design knowledge (see figure 1): Design Artifacts—level A—are composed of 3-dimensional forms and signals, Design Intuition—level B—is also known as neuronal or tacit knowledge, Design Rational—level C—is based on explicit or symbolic knowledge, and Design Theories—level D—are represented as testable models.

Levels of Design Knowledge	Type	Representation	Design Example
D **Model Level** (Models and Theories)	**Design Theories**	Testable Design Theories	Golden Ratio, Design Patterns, Ergonomic Norms
Transition C↔D (Theory Formation / New Concepts)			
C **Symbolic Level** (Explicit Knowledge)	**Design Rational**	Design Terminology, Drawings, Modelmaking, Design Rules, Design Rational	Technical Drawings, Instruction Manual for Machines, Material and Production Knowledge
Transition B↔C (Externalise / Internalise)			
B **Neuronal Level** (Tacit Knowledge)	**Design Intuition**	Design Intuition, Design Skills	Trial-and-Error, Master-Apprentice-Relation
Transition A↔B (Filtering / Adjust Filters)			
A **Physical Level** (3D Form & Instantiations)	**Design Artifacts**	Form, „Gestalt“, Embodied Knowledge	Bottle Opener, Bionics

Figure 1. A Typology of Design Knowledge (Müller et al. 2010)

This framework has similarities to Karl Popper's (1972) "Three Worlds of Knowledge." World 1 (the world of physical objects and events) can be mapped to level A, World 2 (the world of mental objects and events) to level B, and World 3 (the world of the products of the human mind) to level C and D.

This article focuses on the first level: Design Artifacts. As suggested in the above-mentioned typology, artifacts can store and represent knowledge either in their 3-dimensional form (like a bottle opener stores information in its shape about how a bottle can be opened), or as perceivable signals that are captured by an instantiation of a design or a system. The designer or researcher gets feedback from the artifact (signals), but not all signals are perceived, because of a filtering mechanism. This happens in the transition between level A (artifact knowledge) and level B (neuronal knowledge). Part of the design education is to adjust those filters, so that relevant signals can be distinguished from the irrelevant. The question arises, what kind of signals are to be observed, and what is relevant for evaluating the artifact.

For the remainder of this article we refer to this typology of design knowledge and use it as a foundation to define and understand IS artifacts.

Four Types of IT Artifacts According to Hevner et al.

Hevner, March, Park, and Ram (2004) suggest four types of IT artifacts: constructs, models, methods, and instantiations. They describe constructs as a set of vocabulary and symbols. Models are described as abstractions and representations. Methods are defined as algorithms or practices. And instantiations are characterized as implemented and prototyped systems.

According to the previously presented typology of design knowledge, constructs can be placed on level C (*design rational*), since vocabulary and symbols store knowledge in a codified form. Models can be considered either level C or level D knowledge, since abstracted models qualify as *design theories*, while codified representations that are not testable better apply to *design rational*. Methods can be considered either level C or B knowledge, since algorithms are usually codified, while practices might also be tacit knowledge (*design intuition*). And, finally, instantiations can be considered as *design artifacts*.

We agree that constructs, such as particular symbols or vocabulary, are essential for describing artifacts. Models can also represent an artifact in an abstracted form (such as a diagram), and also the suggested methods in the form of algorithms or practices may be considered as a specific representation of an artifact. However, in this paper we do not consider these as design artifacts in the sense of instantiations of a design. Hence, we consider only the last type—instantiations—as design artifacts in terms of physical prototyped objects or implemented systems (see Figure 2 for a comparison).

IS/IT Artifacts According to Gregor and Jones

Gregor and Jones (2007) suggest three types of IS or IT artifacts: instantiations of products and methods (also referred to as material artifacts), a human subjective understanding of artifacts (perception and acceptance from users), and artifacts as theories (including constructs, models, principles, and methods). According to the above-mentioned knowledge typology, instantiations can be considered level A knowledge (*design artifacts)*, human subjective understanding would be level B knowledge (*design intuition*), and theories would be either level C or level D knowledge (*design rational or design theories*). As a consequence, we focus on the first type (instantiations of material artifacts) as design artifacts (see Figure 2 for a comparison).

ARTIFACT KNOWLEDGE IN INFORMATION SYSTEMS

Following the above-mentioned classifications, we limit our research to artifacts as instantiations of a system or design. This concurs with the definition of instantiations by Hevner et al. (2004) and Gregor et al. (2007), and is in-line with the first level of design knowledge (level A) of the typology presented by Müller and Thoring (2010). See Figure 2 for a comparison of different classifications of artifacts.

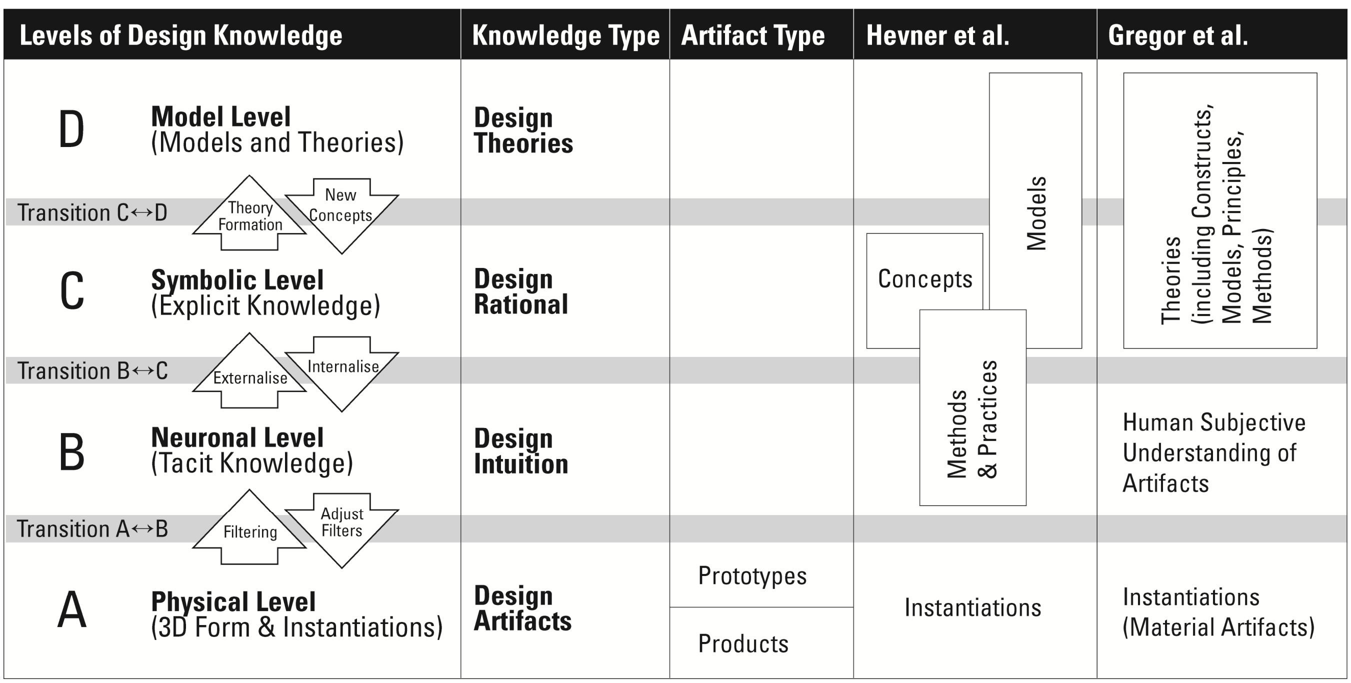

Figure 2: Different Classifications of IT&IS Artifacts

To illustrate further, we suggest a differentiation of artifacts as it is commonly applied in the areas of industrial design or industrial engineering. Here we can distinguish between two types of artifacts with different functions and characteristics: prototypes and products.

Prototypes

Artifacts as 'prototypes' are intended to visualize an idea in order to communicate it to a potential user or to a client. These types of artifacts are also known as 'models' or 'mock-ups'. They are usually not finished, but only a more or less detailed representation of a concept. There exist several stages of detail, which are labeled differently according to different disciplines: In design thinking the word 'prototype' refers to a rough 'mock-up'—a 3-dimensional sketch model that is quickly built from found materials (Brown 2008), while in engineering, it refers to the first instance of an almost finished product (before it goes into serial production). Of course there are many stages in between these two extremes. Rudd, Stern, and Isensee (1996) differentiate these two poles as low-fidelity and high-fidelity prototypes. The purpose of a prototype artifact is mainly to get feedback from potential users, or to test its functionality. While a low-fidelity prototype (a rough paper mock-up, for instance) usually produces user feedback concerning the overall concept, high-fidelity prototypes (that look almost finished) are more likely to elicit feedback about details, such as the choice of colors. You can also distinguish between different working mechanisms of such prototypes: a) a prototype that looks like the real thing, b) a prototype that works like the real thing, and c) a prototype that behaves like the real thing (Buchenau et al. 2000). 'Looks like…' usually refers to a visual resemblance with a prospective design product. The similarity of the model and the real thing can be so close that one couldn't tell the difference by just looking at it. However, the functionality is not implemented in this kind of prototype—just the shell is being imitated. A 'works like…' prototype is quite the opposite. Here, the working mechanism of the real product is being reproduced, while the appearance might be totally different. 'Behaves like…' prototypes usually are used for intangible concepts, such as services or business models. This kind of prototype can be described as a simulation of a desired interaction. Which type of prototype suits best for a specific purpose, depends on the focus of the idea: If it is an aesthetic (re-)design, the 'looks like…' prototype (a) is probably appropriate (for example a clay or foam model or a photomontage, etc.). If it is a technical innovation, you should choose a 'works like…' prototype (b), for example an electronic or mechanic installation. And if your idea is intangible, you should use a 'behaves like…' prototype (c), which could be an interactive animation, a role play, a video, or the so-called Wizard of Oz method (Maulsby et al. 1993), in which the simulation is performed by real people hiding behind e.g. a curtain. Of course, a prototype can also combine more than just one of the above-mentioned aspects. E.g. in the automotive industry, the so-called 'test mule' is the last prototype before the production starts, and it usually looks, works, and behaves exactly as the real car. In the following, we refer to these types of artifacts as 'prototypes'.

Products

The second type of artifact is an already in-use product, which has been carefully designed by a designer or engineer. Such artifacts are established on the market and usually do not need further testing. Therefore the purpose of such artifacts is different. In research contexts, the product can be used for experiments and analyses, even if the product is not your own. The existing artifacts can be used to extract the embedded knowledge. Therefore, products are also relevant in educational contexts. In industrial design and industrial engineering, products can be used to learn from them. A design student can analyze a well-designed chair and try to understand why it is comfortable, or why it looks beautiful. An engineering student can analyze an existing bridge and extract the 'frozen' knowledge about its construction, statics and adequate use of materials. The original designers might have already passed away, but their knowledge is still available in their artifacts. In the further discussion of this paper, we refer to these types of artifacts as 'products'.

Comparison of Prototypes and Products

Both types of artifacts—prototypes and products—have advantages but also some drawbacks, and they can be used for different purposes. Prototypes can be accessed faster, since they are only representations of a certain step within the development process of the artifact. Therefore they can be used to gather feedback from potential users in order to iterate and improve the artifact, before it goes into production. The purpose of prototypes is mainly to test a concept. The problem with prototypes is, that they are usually not accessible for any researcher who wants to run experiments and tests on it, but only for the practitioner(s) who created the artifact. Prototypes sometimes contain only some elements of the artifacts (the ones that are supposed to be tested with the prototype), which means they might be incomplete, which applies to the contained artifact knowledge as well.

Products, on the other hand, need more time until they are available on the market. Then, however, they are usually available or accessible for other researchers. You can just buy the product (if that's feasible), or try to access companies that are working with the product (for instance, it is possible to conduct research on SAP in any company that works with SAP). This, however, limits the availability of such artifacts somehow to well-established (mainstream) products, while cutting-edge products (which are usually more interesting to the researcher) are not so prevalent among companies, which still makes it difficult to access them. The purpose of research on products is usually to extract the existing knowledge in order to learn from it or to conduct research. In contrast to a prototype, a completed product contains all relevant artifact knowledge, and moreover, the knowledge drawn from user feedback is from a real environment—not from a test setting.

The above-mentioned principles apply to every science that deals with artificially created artifacts, such as Architecture, Graphic Design, Computer Science, or Information Systems. We would like to emphasize, that any kind of artifact (prototype or product) is not necessarily a physical object—it could be any instantiation of a design concept, such as a website, a software, a business model, or a service system. In the following we focus on artifacts in the field of Information Systems.

Examples of IS Artifacts

To illustrate the different characteristics of IS artifacts, we present three examples from different areas: 1) a software example, 2) a hardware example, and 3) a systemic design. We then analyze if and how these artifacts can be used to store and transfer design knowledge. We would like to emphasize that the three chosen examples are just meant to compare the advantages and drawbacks of artifact knowledge. This is by no means a direct critique of the presented cases. On the contrary, we would like to express our appreciation for the presented works.

1) Software: Lynne Markus' TOP Modeler

In their well-known article "A Design Theory for Systems That Support Emergent Knowledge Processes" Markus, Majchrzak, and Gasser (2002) refer to a case in which an artifact called TOP Modeler was developed (TOP stands for Technology, Organization, and People integration). The product offers a decision support system for complex decision-making. The described project was supported by a consortium of several Fortune 500 companies and resulted in the creation of a commercial product (TOP Modeler) and a new company to sell the product— according to the paper both was described as 'successful' by all involved parties. However, by now the developed TOP Modeler cannot be found anywhere in the Internet, and the founded company seems not to exist anymore—the artifact is neither traceable nor reproducible. Although a finished product, the artifact of TOP Modeler disappeared. Hence, as a researcher, one has to rely solely on the descriptions provided by the authors, which is mainly level C knowledge plus an additional design theory (level D knowledge), according to the above-mentioned knowledge typology. The question arises, what is the value of a design theory of which the 'proof' is an artifact—if that specific artifact failed?

2) Hardware: Tele-Board–A Digital Whiteboard for Remote Collaboration

In their article "Tele-Board: Enabling Efficient Collaboration In Digital Design Spaces Across Time and Distance" Gumienny, Meinel, Gericke, Quasthoff, LoBue, and Willems (2011) present a digital whiteboard that is intended to be used for remote collaborative brainstorming. The authors claim, that the developed artifact is able to support the design thinking process by enabling creative collaboration between remote teams. The paper, however, is only a verbal description with diagrams and pictures of the artifact—the knowledge is stored in level C—and certain effects of the artifact are explained with theories—knowledge stored in level D of the above-mentioned knowledge typology. The artifact itself, however, is only available to the authors and creators. The Tele-Board can be considered as a prototype, since it is not available on the market, yet. If somebody else would want to work with the artifact or conduct experiments with it, they would need to reproduce it. It remains unclear, if this paper described all the important characteristics, and if the artifact has indeed the described characteristics. Even plausible effects of described characteristics are still hypotheses, which could also be wrong; or implausible effects could be right. Only actual tests of artifacts are able to prove the claimed effectiveness. The question arises, in which detail we have to describe the characteristics of an artifact to guarantee scientific reproducibility.

3) System: The Intel Fab

Intel—the world-leading manufacturer of semiconductors—operates about a dozen factories around the world (so called fabs). The interesting thing about these fabs is that they are all the same—which means *exactly* the same, from the colors of the walls to the bending radius of a particular pipe. The reason is that the system of such a fab is just too complex for anybody to understand it as a whole. Nobody can know, if the bending radius of the pipe in one fab is a coincidence, or if the wall colors maybe influence the room temperature somehow. 'Copy *exactly'* has become Intel's credo (McDonald 1998; McDonald 2002), since the effect of changing the system is somehow unpredictable. All previously made design decisions are stored in the physical representation of the fab (level A design knowledge). If Intel now wants to build a new fab, they totally rely on the stored knowledge of the artifact (the factory) and rebuild it in exactly the same way. If, however, they would want to improve their production process, they would take one fab and change only one parameter (e.g. the color of the walls), and then compare this modified fab with an existing one. The effect is measured in terms of the output loss rate of the semiconductors. The reason for this 'copy *exactly* method' is that there is nobody who understands the whole fab completely. Maybe someone might have a theory about why something is the way it is (level D knowledge), or some aspects have been written down somewhere (level C knowledge), but the artifact alone—the fab—contains all the knowledge about how the entire fab has to be constructed (level A knowledge). What is interesting here is the fact that knowledge stored in the artifact can be somehow hidden or non-intentional—it is not clear why the artifact is build in that specific way, but it proved to work. The fab can be considered a product, since it is running and available. One could go there and examine it (at least in theory).

CONSEQUENCES AND SUGGESTED STRATEGIES

While in most behavioral and natural sciences the emphasis lies on the two top levels, where knowledge is either represented in codified form (written text, equations, or diagrams) or even as scientific theories, the role of intuitive knowledge and artifact knowledge is usually less regarded. In design science, however, which is mainly concerned with research on artificially created objects, the significance of artifacts is already established (Simon 1996).

The artifact is the central outcome of design science (Hevner et al. 2004), however, this outcome is usually not shared in the scientific process. As the three examples presented in the previous section show, artifacts are crucial, since it is possible to store knowledge in a shape or an instantiation of a design. If the artifact is accessible, the knowledge can be extracted by researchers, but if it isn't, the knowledge is difficult to achieve, and claimed characteristics of artifacts cannot be easily reproduced. Moreover, knowledge may sometimes be embedded unintentionally into artifacts, without proper reflection by the creator. This makes it difficult to extract and transfer the artifact knowledge, since it is not condensed into an insight or a theory. Both facts result in what we call the 'design science dilemma'.

According to Star and Griesemer (1989), the communication between different communities can be supported by so-called *boundary objects*. Boundary objects are "both plastic enough to adapt to local needs and constraints of the several parties employing them, yet robust enough to maintain a common identity across sites" (Star et al. 1989). Artifacts can be used as boundary objects, because on the one hand they allow relevant stakeholders to experience the design and to give feedback (Carlile 2002; Henderson 1991; Subrahmanian et al. 2003). On the other hand they hide unnecessary design complexity. Therefore artifacts can be 'plastic enough' and also 'maintain a common identity' between researchers and practitioners. We think that the exchange of artifacts as boundary objects is a promising strategy in researcher-practitioner collaboration (Österle et al. 2010).

The access to artifacts is also important since they can be the source for discovering design patterns (Alexander et al. 1977; Gamma et al. 1994). This is done by filtering and structuring signals from design artifacts (level A) in such a way, that the rational of the design (level C) or even a testable design theory (level D) is created.

But what are the consequences of this design science dilemma? There seems to be a need for more openness concerning research artifacts. From a scientific point of view, those artifacts that are not accessible for other researchers are of less value. Let's take a look at an initiative called "open data", which says that data used in scientific articles must be archived somehow in order to keep it available for later access (Arzberger et al. 2004; Hanson et al. 2011; Miller et al. 2008; Reichman et al. 2011; Rowen et al. 2000). Unlike scientific texts, which are usually available with very limited restrictions, like citation guidelines, specific research data, such as maps, mathematical and scientific formulae, or medical data undergo more restrictive terms of use. They are usually controlled by the involved organizations and are often restricted through licenses, copyright, patents, and access charges. Advocates of the open data initiative claim for a better accessibility of such data without those restrictions. We would go even further and argue for something we call 'open artifacts'. We see the need to guarantee access to artifacts for scientific research purposes. We would recommend establishing a situation in which not only the creators of the artifact (the practitioners) have access to it, but also other researchers, who might want to work with those artifacts.

However, it might not be feasible to share all available artifacts. The question is which artifacts are important and should be selected for open access. In open data, at least all data that form the basis of a scientific claim are interesting for other scientists and should be made public. Therefore we suggest that at least all artifacts that are the source of a design-scientific claim should be open.

In IS-related fields (such as Computer Science) first indications of such a strategy are already becoming evident: Open source and open design initiatives provide products and prototypes for public use and modification. Examples are the Arduino electronics platform (Mellis et al. 2007), the 'One Laptop per Child' initiative (Kraemer et al. 2009), or the Insight Toolkit for medical image analysis (Yoo et al. 2005). There are also some scientific conferences like TREC (Voorhees et al. 2010) or RoboCup (Kitano 1997; Kitano et al. 2002) that demand that authors submit their artifacts additionally to the papers (either as source code or as executable programs).

For Information Systems, we suggest four strategies to establish such a culture of open artifacts: 1) The most radical strategy would include an open design process. This strategy would not only allow the open access to the final prototype or product but the continuous opening of the whole design process for example in a Living Lab process (Almirall et al. 2008; Bergvall-Kåreborn et al. 2009) or through a collaborative design environment. The open design process would be the design science equivalent of the open notebook science (Bradley 2007; Sanderson et al. 2008). This would allow faster knowledge transfer through even preliminary prototypes and also the feedback and contributions from different stakeholders. However, this strategy might not be desirable for a design scientist because the results are not yet published. In this case, we suggest that 2) every practitioner should try to preserve a copy of the artifact as some kind of a time capsule (in case that the company goes bankrupt or the product is taken off the market some day). If for commercial reasons it is not possible to grant free access to that artifact, we suggest 3) to involve some kind of trusted third party (who acts as an escrow holder (Franklin et al. 1997)). That third party could archive the artifact and provide some—maybe limited—access for scientific comparison or testing purposes. Of course, there might be situations where neither 1), 2), nor 3) are possible, and in such cases a verbal description must suffice. Therefore, we suggest 4) to define some criteria for descriptive characteristics of artifacts. The goal would be to provide such a level of detail that other researchers would be able to reproduce the artifact (e.g. through a blueprint). Unfortunately there are two problems with this strategy: On the one hand, such a blueprint would be usually too large for scientific publications, which would result in an arbitrary selection by the creator of only some elements to be published. On the other hand, practitioners usually have no interest in sharing a blueprint in such a detail that others could rebuild the artifact, due to commercial reasons. This even worsens the design science dilemma—artifacts, along with their embedded artifact knowledge, are not accessible and researchers have to rely on limited verbal descriptions provided by the practitioners.

The described four strategies reflect the requirements for artifact accessibility form a scientific point of view. Of course creating open access to artifacts might not be commercially feasible for a lot of companies. Companies might or might not want to contribute to the scientific knowledge base. However, our paper is focusing on design science and not just design. We don’t imply that all companies should make their artifacts public. We only describe the consequences of closed artifacts for design science and suggest that ‘open artifacts’ could solve the design science dilemma.

CONCLUSION

Collaboration and knowledge exchange among researchers or between researchers and practitioners usually occurs through explicit and somehow codified knowledge, such as text, speech, formulae, diagrams, or even theories. Other forms of knowledge representations are rarely considered. We argue in this paper, that artifact knowledge is not only supporting this communication, but it might even contain additional knowledge that is not present in the codified form. To illustrate this phenomenon, we present an example of Intel factories, which store knowledge about a very complex system in their physical construction. Sometimes the creators of such artifacts are not even aware of the specific knowledge embedded in the artifact. While codified knowledge might get lost, especially in complex systems, the artifact remains and freezes the knowledge as long as it exists. As a consequence, we advise researchers to pay thorough attention to artifacts, since this might be a good way to extract the embedded knowledge to use for their research. To illustrate further, we present two more examples of software and hardware artifacts that both are not accessible for researchers. The characteristics of these artifacts are just described in a verbalized form, which makes it difficult if not impossible for researchers to reproduce these artifacts. Hence, we suggest that practitioners consider publishing their artifacts or archiving them as long as possible, in order to ensure accessibility for other researchers who might want to analyze them. Such a culture of ‘open artifacts’ would improve the researcher–practitioner collaboration to a high degree. The implementation of a trusted third party that archives artifacts and provides access to it on a limited basis might be an option to consider.

Furthermore, we highlight the difference between prototype and product artifacts. While the former represents an incomplete instantiation of an artifact, which does usually not contain all the knowledge about all the aspects of the artifact, the latter represents a complete artifact that embodies knowledge about all relevant aspects. However, this should not imply that products are better artifacts than prototypes. Both types of artifacts can provide insights about the embedded knowledge—however, the researcher must be aware of limitations concerning the completeness of the embedded information and possible differences in feedback the two types might produce.

Finally, we highlight the problems that occur when artifacts are not accessible for researchers. We debate the need to define descriptive characteristics for artifacts that could be used to reproduce an artifact that is not accessible. Such an analysis of descriptive characteristics should be conducted in further research.

The work presented in this article provides a foundation to implement or to improve artifact-based knowledge transfer and re-engineering. It contributes to a better understanding of the role of artifacts in design science and therefore might serve as a first step to challenge the design science dilemma.

REFERENCES

1. Alexander, C., Ishikawa, S., and Silverstein, M. *A pattern language: towns, buildings, construction* Oxford University Press, USA, New York, 1977.
2. Almirall, E., and Wareham, J. "Living Labs and open innovation: roles and applicability," *The Electronic Journal for Virtual Organizations and Networks* (10:3) 2008, pp 21-46.

3. Arzberger, P., Schroeder, P., Beaulieu, A., Bowker, G., Casey, K., Laaksonen, L., Moorman, D., Uhlir, P., and Wouters, P. "Promoting access to public research data for scientific, economic, and social development," *Data Science Journal* (3:0) 2004, pp 135-152.
4. Bergvall-Kåreborn, B., Holst, M., and Ståhlbröst, A. "Concept design with a Living Lab approach," in: *Proceedings of the 42nd Hawaii International Conference on System Sciences (HICSS)*, IEEE, 2009.
5. Bradley, J.C. "Open notebook science using blogs and wikis," in: *American Chemical Society Symposium on Communicating Chemistry. Nature Precedings.*, Nature, 2007.
6. Brown, T. "Design thinking," *Harvard Business Review* (86:6) 2008, pp 84-92.
7. Buchenau, M., and Suri, J.F. "Experience prototyping," Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques, 2000, pp. 424-433.
8. Carlile, P.R. "A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development," *Organization Science* (13:4) 2002, pp 442-455.
9. Franklin, M.K., and Reiter, M.K. "Fair exchange with a semi-trusted third party," in: *Proceedings of the 4th ACM Conference on Computer and Communications Security*, ACM, 1997, pp. 1-5.
10. Gamma, E., Helm, R., Johnson, R., and Vlissides, J.M. *Design Patterns: Elements of Reusable Object-Oriented Software* Addison-Wesley Professional, Reading, MA, 1994.
11. Gregor, S. "The nature of theory in information systems," *Management Information Systems Quarterly* (30:3) 2006, pp 611-611.
12. Gregor, S., and Jones, D. "The anatomy of a design theory," *Journal of the Association for Information Systems* (8:5) 2007, pp 312-335.
13. Gumienny, R., Meinel, C., Gericke, L., Quasthoff, M., LoBue, P., and Willems, C. "Tele-Board: Enabling Efficient Collaboration In Digital Design Spaces Across Time and Distance," in: *Design Thinking,* H. Plattner, C. Meinel and L. Leifer (eds.), Springer, Berlin, 2011, pp. 147-164.
14. Hanson, B., Sugden, A., and Alberts, B. "Making Data Maximally Available," *Science* (331:6018) 2011, pp 649-649.
15. Henderson, K. "Flexible sketches and inflexible data bases: Visual communication, conscription devices, and boundary objects in design engineering," *Science, technology & human values* (16:4) 1991, pp 448-448.
16. Hevner, A.R., March, S.T., Park, J., and Ram, S. "Design science in information systems research," *MIS Quarterly* (28:1) 2004, pp 75-105.
17. Kitano, H. "RoboCup as a research program," Intelligent Robots and Systems, 1997. IROS'97., Proceedings of the 1997 IEEE/RSJ International Conference on, 1997.
18. Kitano, H., Asada, M., Noda, I., and Matsubara, H. "RoboCup: Robot world cup," *Robotics & Automation Magazine, IEEE* (5:3) 2002, pp 30-36.
19. Kraemer, K.L., Dedrick, J., and Sharma, P. "One laptop per child: vision vs. reality," *Communications of the ACM* (52) 2009, pp 66-73.
20. Markus, L., Majchrzak, A., and Gasser, L. "A Design Theory for Systems That Support Emergent Knowledge Processes," *MIS Quarterly* (26:3) 2002, pp 179-212.
21. Maulsby, D., Greenberg, S., and Mander, R. "Prototyping an intelligent agent through Wizard of Oz," Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems, ACM, New York, NY, USA, 1993, pp. 277-284.
22. McDonald, C.J. "The Evolution of Intel's Copy EXACTLY! Technology Transfer Method'," *Intel Technology Journal* (4:1) 1998.
23. McDonald, C.J. "Copy EXACTLY! A paradigm shift in technology transfer method," Advanced Semiconductor Manufacturing Conference and Workshop, 1997. IEEE/SEMI, 2002, pp. 414-417.
24. Mellis, D., Banzi, M., Cuartielles, D., and Igoe, T. "Arduino: An open electronic prototyping platform," in: *Proc. CHI*, ACM, San Jose, USA, 2007.
25. Miller, P., Styles, R., and Heath, T. "Open data commons, a license for open data," Proceedings of the Workshop on Linked Data on the Web (WWW2008), 2008.
26. Müller, R., and Thoring, K. "A Typology of Design Knowledge: A Theoretical Framework," in: *AMCIS 2010 Proceedings*, 2010.
27. Österle, H., and Otto, B. "Consortium Research: A Method for Researcher-Practitioner Collaboration in Design-Oriented IS Research," *Business & Information Systems Engineering* (2:5) 2010, pp 283-293.
28. Popper, K. *Objective knowledge: an evolutionary approach* Clarendon Press, Oxford, 1972.
29. Radermacher, F.J. "Cognition in systems," *Cybernetics and Systems* (27:1) 1996, pp 1-41.
30. Reichman, O.J., Jones, M.B., and Schildhauer, M.P. "Challenges and Opportunities of Open Data in Ecology," *Science* (331:6018) 2011, pp 703-705.
31. Rowen, L., Wong, G.K.S., Lane, R.P., and Hood, L. "Intellectual property. Publication Rights in the Era of Open Data

Release Policies," *Science* (289:5486) 2000, p 1881.
32. Rudd, J., Stern, K., and Isensee, S. "Low vs. high-fidelity prototyping debate," *interactions* (3:1) 1996, pp 76-85.
33. Sanderson, K., and Neylon, C. "Data on display," *Nature* (455:7211) 2008, p 273.
34. Simon, H.A. *The sciences of the artificial* MIT Press, Cambridge, MA, 1996.
35. Star, S.L., and Griesemer, J.R. "Institutional Ecology, `Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39," *Social Studies of Science* (19:3) 1989, pp 387-420.
36. Subrahmanian, E., Monarch, I., Konda, S., Granger, H., Milliken, R., and Westerberg, A. "Boundary objects and prototypes at the interfaces of engineering design," *Computer Supported Cooperative Work (CSCW)* (12:2) 2003, pp 185-203.
37. Voorhees, E.M., and Buckland, L.P. *The Nineteenth Text REtrieval Conference (TREC 2010) Proceedings*, Gaithersburg, Maryland, 2010.
38. Yoo, T.S., and Metaxas, D.N. "Open science - combining open data and open source software: Medical image analysis with the Insight Toolkit," *Medical Image Analysis* (9:6) 2005, pp 503-506.