The Anatomy of a Design Principle

Article *in* Journal of the Association for Information Systems · November 2020

DOI: 10.17705/1jais.00649

3 authors:

Shirley Gregor
Australian National University

Leona Chandra Kruse
University of Liechtenstein

Stefan Seidel
University of Liechtenstein

The Anatomy of a Design Principle*

Shirley Gregor
Australian National University
shirley.gregor@anu.edu.au

Leona Chandra Kruse
University of Liechtentsein
leona.chandra@uni.li

Stefan Seidel
University of Liechtenstein
stefan.seidel@uni.li

**Paper accepted for publication in*
Journal of the Association for Information Systems.

Abstract

This essay derives a schema for specifying design principles for information technology-based artifacts in sociotechnical systems. Design principles are used to specify design knowledge in an accessible form, but there is wide variation and lack of precision across views on their formulation. This variation is a sign of important issues that should be addressed, including a lack of attention to human actors and levels of complexity as well as differing views on causality, on the nature of the mechanisms used to achieve goals, and on the need for justificatory knowledge. The new schema includes the well-recognized elements of design principles, including goals in a specific context and the mechanisms to achieve the goal. In addition, the schema allows: (i) consideration of the varying roles of the human actors involved and the utility of design principles; (ii) attending to the complexity of IT-based artifacts through decomposition;(iii) distinction of the types of causation (i.e., deterministic versus probabilistic); (iv) a variety of mechanisms in achieving aims; and (v) the optional definition of justificatory knowledge underlying the design principles. We illustrate the utility of the proposed schema by applying it to examples of published research.

Keywords: design principle, design science research, design theory, prescriptive knowledge

1. Introduction

Design science research (DSR) aims to provide knowledge that has scientific legitimacy and also provides utility in achieving goals. The characteristic that distinguishes design science knowledge from other forms of knowledge is that it includes design principles: prescriptive statements that show how to do something to achieve a goal. This "know how" knowledge has played an important role in human history since ancient times, and understanding how to formulate design principles continues to play an important role with today's increasingly complex artifacts so they can be used as a means of accumulating knowledge and acted on in real-world situations.

Research has examined design knowledge in terms of design theory (Gregor & Jones, 2007; Walls, Widermeyer, & El Sawy, 1992; 2004) and technological rules (e.g., Bunge, 2009; van Aken, 2001) and in recognition of a continuum from the knowledge represented in an instantiation of an artifact, through nascent design theory in the forms of design principles, schemas, and methods, to full design theory (Gregor & Hevner, 2013). The examination of design theory in Gregor and Jones' *Anatomy of a Design Theory* (2007) was detailed and in-depth. Now, more than ten years after that work was published, it is time to look again at the most distinctive part of a design theory's anatomy: the prescriptive knowledge represented in the design principles. Further analysis shows inconsistencies in how design principles have been treated in the literature and a comparative lack of attention to the "people aspects" of design principles. More attention should also be paid to issues such as the possibility of decomposition as well as the artifact propensities and affordances that allow for IT-based artifacts' non-deterministic potential. A fresh look is needed to synthesize what is known about design principles and to point to some new directions. Therefore, we ask:

> *How should design principles for technology-based artifacts in socio-technical systems be presented so that they are understandable and useful in real-world design contexts?*

We contend that attending to this question will help us devise a formulation of design principles that accounts for their nature as prescriptive knowledge that can readily be applied in design situations where implementers require guidance. Therefore, the objective of the current work is to derive a schema for specifying design principles that is appropriate for research practice and that supports the application of design knowledge to professional practice. The context is that of IT-based systems that include human and non-human actors (i.e., socio-technical systems), and the approach includes critical analysis of prior work, an analysis of research practice, and some novel insights.

This work makes a number of contributions. First, we synthesize the various conceptualizations of design principles to provide a precise yet integrative perspective. Second, we attend to the various roles of human actors and their use of design principles. Third, we show how decomposable principles help to reduce the complexity involved in formulating design principles for IT-based artifacts and allow the principles to be presented to an audience of designers at multiple levels of abstraction. Fourth, we attend to several types of causal reasoning that can be considered in the formulation of design principles, thereby accounting for the deterministic effects of information system artifacts and the propensities and affordances that enable ends. This differentiation is important, as socio-technical systems involve both elements that deterministically achieve an outcome (e.g., algorithms) and elements in which human and machine elements interact to bring about a result. This interaction means indeterminacy, so designers need to consider that artifacts can be used in unexpected ways. Fifth, we provide an encompassing view of how the means or mechanisms for achieving goals can be specified. Finally, we attend to the optional role of justificatory knowledge or kernel theory that underlies design principles. In some cases, design principles are developed based on observation or experimentation, while in others they are based on prior theoretical knowledge in a field. The multi-level schema is formulated in a way that is generally applicable to design principles for IT-based artifacts, including socio-technical artifacts that involve both human and machine actors. Our analysis is based primarily on the use of design principles in information systems, but the schema has broader applicability, which presents opportunities for further work.

We proceed as follows. The next section describes the design knowledge and design principles that serve as a background to the study, compares prior conceptualizations of design principles, and highlights some issues that require further examination. This overview is followed by the description of our research approach, a report on a grounded study of design principles in research practice, and an exploration of the previously identified issues. Next, we describe an integrative schema for specifying design principles at multiple levels, apply it to published work that includes design principles explicitly and implicitly, discuss the implications for both research and practice, highlight the limitations of our study, and draw conclusions.

2. Conceptual Background

The purpose of this section is to introduce a number of basic ideas and terms related to design knowledge and DSR, to compare prior conceptualizations of design principles with each other, and to identify issues with previous formulations that require further examination.

2.1 Introduction to Design Principles

Design activities and technologies have always been important in applied science disciplines such as engineering, economics, medicine, computer science, applied mathematics, and information systems. The type of knowledge that is produced in these applied disciplines is "know how"—imperative or prescriptive knowledge—as opposed to the descriptive "know what" knowledge found in other areas of science (Niiniluoto, 1993)[1]. Exhibit 1 provides a simple example of design knowledge expressed in a design principle that is attributed to Aristotle (Kenny, 1996).

Exhibit 1: Design Principle Example

Statement Number	Statement	Comment
(1)	To provide pain relief to individuals with contusion injuries, …	Establishes the aim and the recipient
(2)	in general …	Establishes the context
(3)	apply cold (e.g., an ice pack)…	Prescriptive statement to show someone (implicitly) how to obtain the aim
(4)	because application of cold to a contusion injury has a pain-killing effect and helps stop internal bleeding	Descriptive statement providing rationale

As Exhibit 1 demonstrates, the design principle, to provide pain relief to individuals with contusion injuries, is an abstraction, as it does not refer to a concrete instance. In addition, it assumes that

[1] Here, as throughout this essay and in much of the literature on design knowledge, the terms "imperative" and "prescriptive" apply to feasible means for achieving an end. These terms are not used in a normative sense to imply that one course of action is better in some way than another in terms of some value system, or that the course of action should or ought to be done. Neither do they imply a command, although they can be discussed in terms of imperative logic (Simon, 1996, p. 115) and in relation to "the logic of action" (Segerberg, Meyer, & Kracht, 2016).

someone can understand the principle sufficiently well to be able to apply it and achieve a desired or at least acceptable outcome. The design principle contains a lower-level abstraction that refers to a designed artifact (an ice pack) and is at an appropriate level; that is, the audience for the design principle is able to understand what this abstraction means. The prescriptive statement to apply an ice pack has process (action) steps ("apply" something) and refers to a "thing" that can be applied (an ice pack). The rationale for the prescriptive statement is probabilistic rather than deterministic, as the treatment of cooling may not always work. The prescriptive statement can also be inferred from descriptive statements (in this case, that application of cold to a contusion injury has a pain-killing effect and helps stop internal bleeding), but such inferences are not always possible. The efficacy of applying cold may have been discovered through experimentation and justified through repeated experience, without the underlying anatomical knowledge shown in the descriptive statement. Finally, the treatment (apply an ice pack) does not follow necessarily in terms of deductive logic from the descriptive statement, as there may be other feasible and possibly better ways of treating the contusion and obtaining the goal of providing pain relief.

2.2 Design Knowledge and Conceptualization of Design Principles

Concern with design knowledge as a special type of knowledge has grown across a number of disciplines. Seminal thinking was presented in the first edition of the influential monograph *The Sciences of the Artificial* (Simon, 1996, p. xii), which addresses:

> *in the problem of artificiality an explanation of the difficulty that has been perceived in filling engineering and other professions with theoretical and empirical substance distinct from the substance of their supporting sciences. Engineering, medicine, business and architecture, and painting are concerned not with the necessary but with the contingent—not with how things are but with how they might be—in short, with design.*

Simon's (1996) work had a number of things to say about knowledge of artificial things, even though he did not describe precisely the structure of this knowledge. He did show how complexity could be dealt with through decomposition, and that it was not always necessary in specifying the design of a complex system to know all the inner workings (the interior view) of a component (a module), but only that the component could achieve a certain goal in a particular environment (the exterior view). This “decomposable” aspect of design knowledge, although generally applied in practice in computer science and software engineering, has not been so explicitly taken up in discussions of design knowledge.

The research approach that develops design knowledge is now commonly called “design science,” a term introduced by Buckminster Fuller in the 1960s (Fuller, 1983) to refer to a combination of science, technology, and rationalism. The forms that design knowledge can take have been referred to as design theory (Gregor & Jones, 2007; Walls et al., 1992), theory for design and action (Gregor, 2006), design patterns (Alexander, Ishikawa, & Silverstein, 1977), technological rules (Bunge, 2009; van Aken, 2001), technical norms (Niiniluoto, 1993; von Wright, 1963), design rules (e.g., Plsek, Bibby, & Whitby, 2007), analysis patterns (Fowler, 2007), computing principles (Denning & Martell, 2015), design propositions (van Aken, Chandrasekaran, & Halman, 2016), and design principles (Gregor & Hevner, 2013; Sein, Henfridsson, Purao, Rossi, & Lindgren, 2011).

Design science as a research approach has gained legitimacy in a number of fields, including information systems, and with specialized workshops and conferences, tracks in major conferences, and editors’ calls for more design science work (e.g., Goes, 2014). The *Journal of Operations Management* has even introduced a design science department (see van Aken et al., 2016). DSR is seen as one way of responding to calls for academics to engage in work that has greater impact outside academia. Histories of design research in information systems can be found in Iivari (2007) and March and Storey (2008). Research methodologies for DSR (e.g., Bider, Johannesson, & Perjons, 2012; Peffers, Tuunanen, Rothenberger, & Chatterjee, 2007) and action design research (Sein et al., 2011) are well accepted, and textbooks have appeared (e.g., Johannesson & Perjons, 2014). In charting a course for information systems research in the next twenty-five years, Lee (2010, p. 345) contended that “the predominant form of theory in IS research ultimately needs to become theory for design and action.”

The last decade has seen an increase in IS research that has addressed business and societal challenges by systematically designing, developing, and evaluating innovative artifacts, and that has contributed to knowledge through this process (Rai, 2017). These innovative artifacts are expected to fulfill particular goals through their material properties and, more importantly, through their functional affordances (Markus & Silver, 2008). This line of thinking indicates the need to understand users in their routines and activities and their interaction with the envisioned artifacts.

When practicing DSR, IS researchers follow various genres and methodological approaches. Along with ten other contributors, Rai (2017) proposed six genres: (i) *design thinking*, which deepens our understanding of the relationship between human experience and digital artifacts as articulated in the vision of an "archaeology of the future" (Dahlbom, 2002, p. 33), "experiential computing" (Yoo, 2010), and "performative research" (Law & Urry, 2004); (ii) *a complexity perspective*, which encourages DSR researchers to deal effectively with the messy complexity of system problems, learn and adapt through the design process, and reflect on the results to identify and extend knowledge (Gregor & Hevner, 2013); (iii) *the computational genre* of DSR, which embraces an interdisciplinary approach to developing novel representations of data, computational algorithms, business intelligence and analytics methods, and innovations in human-computer interactions (HCI) (e.g., Chen, Chiang, & Storey, 2012; Lin, et al., 2017); (iv) *the optimization genre* of DSR, which designs and implements IS solutions to support process innovation and value creation (e.g., Menon & Sarkar, 2016); (v) the representation genre of DSR, which designs and validates schemas, grammars, scripts, and methods that facilitate the faithful representation of phenomena in the domain of interest (e.g., Burton-Jones & Volkoff, 2017; Lukyanenko et al., 2014); and (v) the IS economics genre of DSR, which seeks to explain the roles of IT functionalities in economic activities and goal attainments and to design such IT artifacts (e.g., Ketter et al., 2016).

Peffers, et al. (2018) undertook a similar pursuit, resulting in five DSR genres: (i) *IS design theory* (Gregor & Jones, 2007), which emphasizes the development and presentation of IS design theories and validates them conceptually or through an artifact instantiation; (ii) *DSR methodology* (Peffers et al., 2007) and (iii) *design-oriented IS research* (Österle, et al., 2011; Winter, 2008), which focus more on developing and evaluating useful artifacts than on building theory; (iv) *explanatory design theory* (Baskerville & Pries-Heje, 2010; Niehaves & Ortbach, 2016), which emphasizes design features and their effect on the environment; and (v) *action design research* (Sein, et al., 2011), which combines action research and design research and views design as a situated process that occurs in an organizational context and as a reflective process that generates prescriptive design knowledge about a class of artifacts to address a class of problems.

Gregor and Jones (2007) gave a full account of design theory with the aim of showing how this form of theory was comparable to views of theory in other areas of science. Their "anatomy of a design theory" showed design theory as being composed of eight components: purpose and scope, constructs, principles of form and function, artifact mutability, testable propositions, justificatory knowledge (kernel theory), principles of implementation, and an expository instantiation.

Design principles, as shown in Exhibit 1 and as represented by other authors, are an important part of design theory, as they contain the distinctive element that distinguishes design knowledge: the prescriptive statements. Design principles are comparable to Gregor and Jones' (2007) component (3), principles of form and function, in their formulation of a design theory. In this paper, we focus on the detailed structure of these prescriptive design principles. Table 1 shows the range of views and nomenclature for design principles.

Table 1: Views on Design Principles

Terminology	Field	Definition and Reference
Technical norms	General	Niiniluoto's technical norms are of the form "If you want A, and you believe that you are in a situation B, then you ought to do X" (Niiniluoto, 1993, p. 12), citing Von Wright (1963).
Technological rule	General	To achieve A, do (act_1, act_2, …, act_n) (Bunge, 1967). "Instructions to perform a finite set of actions, including manipulations of one or more artifacts, in a given order and with a given aim" (Bunge, 1967, p. 132).
Design pattern	Software design	"a method of mapping human actions to software functions in a way that is intelligible to clients, designers, and engineers simultaneously" (Denning & Dargan, 1996, p. 6).
	Object-oriented design	"Descriptions of communicating objects and classes that are customized to solve a general design problem in a particular context" (Gamma, 1995, p. 13).
Analysis pattern	Business schema	"Groups of concepts that represent a common construction in business schemes. It may be relevant to only one domain, or it may span many domains" (Fowler, 1997, p. 8).
Design principle	Education	"If you want to design intervention X (for the purpose/function Y in context Z), then you are best advised to give that intervention the characteristics A, B, and C (substantive emphasis), and to do that via procedures K, L, and M (procedural emphasis), because of arguments P, Q, and R)" (Van den Akker, 1999, p. 9).

Technological rule	Management	"if you want to achieve Y in situation Z, then something like action X will help" (van Aken 2004 p. 227).
Technological knowledge	General	"Goal-directed series of considered actions, including manipulations of one or more artefacts" (Houkes & Vermaas, 2004, p. 57).
Principles of form and function	Information systems	"The abstract 'blueprint' or architecture that describes an IS artifact, either product or method/intervention" (Gregor & Jones, 2007, p. 322).
Computing principle	Computing	Computing principles for conduct have the purpose of enabling "good design by increasing understanding and reducing complexity" (Denning & Martell, 2015, p. xiv).
Design proposition	Management	"if you want to achieve Y in situation Z, then use the generic design X (or perform the action type X): Y = X(Z)" (van Aken et al., 2016 p. 4).

2.3 Synthesis and Issues

The analysis in Table 2 shows some common components in prior conceptualizations of design principles and some divergent thinking. While views on the form design principles should take vary, all conceptualizations agree on the requirement for a statement of the aim (goal, purpose) and means for achieving the goal. However, there is little or no recognition of the actors concerned with the design principle and its use, apart from indiscriminate use of the term "you." An exception is Denning and Martell (2015), who refer to the design principle's being used by human designers/implementers to aid understanding (Table 1).

Table 2: Analysis of Existing Formulations of Design Principles

Component	Reference	Comment
Aim	All formulations refer to a "goal" (Bunge, 2009), "aim" (von Wright, 1963), "purpose" (Gregor & Jones, 2007), or similar concept.	Some formulations refer to the aim's being tied to an individual user (e.g., if **you** want the aim), while others do not.
Context/boundary condition	Not included by all: "in situation B" (von Wright, 1963), "scope" (Gregor & Jones, 2007).	Aim and context are often closely linked. Heidegger (1993) gave an example of a silver chalice: we cannot fully understand the nature of the aim/requirement unless we understand that the chalice is to be used in a religious ceremony, where an object of beauty is important (Heidegger, 1993).
Means of achieving aim	All formulations include some component of this type, but there are variations: "finite number of acts in a given order" (Bunge, 2009), "intervention or artifact" (van Aken, 2004), "principles of form and function" (Gregor & Jones 2007), "manipulation of one or more artifacts, in a given order" (Houkes, 2009), and "something like action X	There is variation between humans doing something (acting/intervening) and/or using an artifact, and variation in whether there is one or more in a series of actions/use of artifacts. Human activity is not distinguished from an artifact's activity. van Aken (2004) indicated some indeterminacy in that the means may be "something like" what is specified.

	will help" (van Aken, 2004, p. 227).	
Justificatory knowledge	Not included by all: "grounded on scientific knowledge" (Bunge, 1967), "justificatory knowledge" (Gregor & Jones, 2007), "kernel theory" (Walls et al., 1992).	Gregor and Jones (2007) defined "justificatory knowledge" as "the underlying knowledge or theory from the natural or social or design sciences that gives a basis and explanation for the design (kernel theories)."

The definitions in Table 1 and the analysis in Table 2 highlight a number of issues that require further examination:

- There is inconsistency in views of how design principles should be formulated.
- There is little attention to the actors involved in applying the design principle, whether as the designer implementing it, the recipient user tied to the aim to be achieved, or those who play a part in achieving the aim. Several conceptualizations use an undifferentiated "you" for these actors.
- Existing formulations of design principles do not provide means to deal with complexity such as decomposition (Arthur, 2009; Simon, 1996). This issue is tied to the second issue, as an appropriate level of generality could assist implementers to understand the design principles. Decomposition may be required if understanding is missing about how some element of the "means"—either an action or the use of an artifact—is to be achieved.
- Existing formulations do not distinguish types of causation, particularly artifacts' functioning, as in Houkes (2009) (where the IT-based artifact deterministically performs certain actions); human action, as in Bunge (2009) (where the IT-based artifact provides a potential for human action that may or may not occur); and the possibility of interactions between these components (i.e., interactions between human usage activity and an artifact's functioning).
- The formulations differ with regard to the composition of the means for achieving the aim (whether the means is a single act or multiple acts) and the nature of the means (whether it includes people and actions, as well as artifacts), and they do not all attend to the temporal logic that underlies processes.
- The formulations differ in their emphases on the need to provide supporting knowledge for the design principle.

Next, we describe our research method and then address the issues outlined above.

3. Research Approach

The method employed in developing this article is in itself a DSR approach. (For a smiliar approach, see Gregor and Hevner (2013)). We draw inspiration from Peffers et al.'s (2007) work, whose guidance on conducting DSR consists of six steps: identify and motivate the problem, define the objectives of a solution, design and develop the solution, demonstrate the solution, evaluate the solution, and communicate the solution. We explain how these guidelines are applied in the context of our research by describing four key clusters of activities: recognizing the problem situation and setting goals; examining the use of design principles in information systems research to identify issues with current formulations of design principles and so define the objectives of the envisioned solution; developing the anatomy of a design principle; and demonstrating the anatomy of a design principle as a first step toward evaluation. This paper is a key part of a final communication step of this project.

Activities Cluster 1: Recognizing the problem situation and setting goals

Recognizing the problem situation occurred by means of reflective and empirical approaches. The empirical approach included personal experience with formulating design principles in real-world projects (Gregor et al., 2014), and the reflective approach involved looking back through the professional and philosophical journey of one of the authors and collectively examining the author's extant conceptual work (e.g., Gregor, 2006; Gregor & Jones, 2007; Gregor & Hevner, 2013). Motivation to continue the project arose because we recognized the need for an exploration of the most distinctive part of Gregor and Jones' (2007) design theory anatomy, design principles, but observed issues in their specification and use.

Activities Cluster 2: Examining the use of design principles in information systems research

We performed the groundwork for investigating an area that had been given scant attention in prior conceptualizations of design principles: the roles of the human actors involved. We addressed the question concerning how information systems studies that developed design principles had addressed human activity. The extant formulations of design principles, which differed in how they treat human activity, were investigated in a literature review. We conducted the first literature review in 2014, the results of which suggested that formulations of design principles either focused on user activity, or on artifacts, or indeed considered both (see Chandra, Seidel, & Gregor, 2015). An update of this literature review that we completed March 2020 confirmed our analysis. In this last review, we obtained a sample of 69 articles from the eight journals in the AIS Senior Scholar's basket of journals, each of which provided a set of explicit design principles. We examined these articles for the manner in which they treated user activity: 11 articles presented sets of design principles that addressed user activity, 27 presented sets about artifacts, and 31 presented sets attending to both. Appendix A provides further detail of this analysis.

Activities Cluster 3: Developing the anatomy of a design principle

After examining the identified issues, we developed a new schema for a design principle based on the results from previous steps and derived a means for representing the schema graphically. As is common in design endeavors, the development of our conceptual schema was abductive in nature. Our reading of the existing literature and examples of published design principles led to an attempt at providing the best possible explanation of how design principles were formulated and, where feasible, why. The explanation is captured in our schema. Consequently, the production did not follow pre-determined sequences, as we often revisited our schema and added new ideas when they emerged.

Activities Cluster 4: Demonstrating the applicability of the schema

We provided a proof-of-concept demonstration of the applicability of the proposed schema with illustrative cases that we analyzed to explore the use of the schema in portraying design principles as represented either explicitly or implicitly in published work.

The following sections provide further details of the activities.

4. Design Principles in Research Practice in Information Systems

An analysis of published articles that have presented information systems design principles (Appendix A) identified three categories of design principle formulation: design principles that encapsulate users' use of artifacts, design principles that encapsulate artifact features, and design principles that describe both (i.e., that are focused on both artifact features and user activity).

Design principles that fall into the first category primarily state what (human) users should be able to do with an artifact, so we call this category *design principles about user activity*. These design principles generally say "the system should support users in doing this or that" or "the intervention should support/improve goal A, B, C of the employee/team/organization." These design principles emphasize the role of human activity in the design principle, which distinguishes this category from another category that focuses on the features or the functionalities that are embedded in the artifact. An example of *design principles about human activity* concerns the development of knowledge management systems:

> Social actors (spectators) who experience "breakdowns: in understanding should be able to use the technology to access the interpretations of others who faced similar situations in the past, to learn from the experiences of these social actors, and apply this learning in repairing their own "breakdowns", build more informed "horizons of understanding", thereby informing subsequent action. (Butler & Murphy, 2007, p. 159)

This particular design principle is formulated from the users' point of view, as it indicates user activities, from experiencing "breakdowns" to learning from others' experience and applying the learning to one's own context.

The second category is that of *design principles about an artifact*, which focus on the features that should be built into an artifact, including shape/architecture and function. Design principles that fall into this category usually say "the system should do this or that," "the system should have features F, G, H," or in the case of interventions, "the intervention should have the procedure P, Q, R or take the form of this or that." For instance, a set of design principles for artificial immune systems that can detect credit card fraud (Wong, Ray, Stephens, & Lewis, 2012, p. 70) suggests that it should be "multilayered: The immune system is composed of many layers from physical barriers such as skin through to the lymphocyte detectors. These layers in combination offer a complete defense system against foreign antigens."

The third category includes principles that combine the properties of the principles that belong to the first two categories, which spell out what users should be able to do with an artifact as well as the features the artifact should have to allow that particular user activity, *design principles about user activity and an artifact*. Such design principles prescribe that "the system should have features F, G, H and do I, J, K, in order to allow users to do X, Y, Z" or "the intervention follows procedure P, Q, R and has features F, G, H in order to support people in activity A, B, C." An example of this category prescribes the design of creativity support systems (Müller-Wienbergen, F., Müller, O., Seidel, S., & Becker, J., 2011, p. 724): "Principle C3: Enable dynamic filtering of the knowledge base - Different types of graphical filters can be combined to interactively restrict the set of displayed knowledge items." This design principle states which user activity is to be supported (users can filter the knowledge base in a dynamic manner) and the features a creativity support system should have to support them (integrated graphical filters). Table 3 summarizes the three categories and provides an example for each category based on the construction and use of windows in everyday life.

Table 3: Three Categories of Design Principles with Respect to User Activity.

Design principles about user activity
These principles state what (human) users can do with an artifact (i.e., what it should allow the user to do). **Example:** Build a window so people can see through it.
Design principles about an artifact
These principles state the features an artifact should have (i.e., shape/architecture and function). **Example:** Assemble a window with a frame and transparent material to fill the frame.
Design principles about user activity and an artifact
These principles combine the characteristics of the two previous ones and contain what users should be able to do with an artifact ***and*** the characteristics it should possess. **Example:** Assemble a window with a frame and transparent material to fill the frame, so people can see through it.

In sum, the review of design principles shows that human activities can be both aims and mechanisms to achieve aims.

5. Examination of Issues in Formulating Design Principles

In this section we examine in detail the issues identified in our analysis of specifications of design principles and present ideas on how to address them. The first issue we identified, inconsistency in specification, is an overarching issue that we address by examining separately the sub-issues related to the roles of human actors, the complexity of design principles, types of causation, means to achieve ends, and justification of design principles.

Design principles are theoretical abstractions that serve a purpose and have utility, a definition that is congruent with recent pragmatic perspectives on theory, in contrast to earlier syntactic and semantic perspectives (Gregor, 2017; Winther, 2016). The literature has considered the theorizing process necessary to arrive at these abstractions, including methodologies for design science and action design research, where reflection/abstraction and application/experimentation are shown as occurring in cycles until relatively stable design knowledge can be formalized. A detailed examination of the abstraction process can be found in Gregor, Müller, and Seidel (2013). Discussion of the application process was provided by, for example, van Aken (2004) with respect to technological rules in management. Lukyanenko and Parsons (forthcoming) discuss some of the difficulties in implementing design principles that relate to their potential lack of applicability in new contexts.

5.1 The Role of Actors

"Actors" may be humans or non-humans, that is, "automata" (Bunge, 2009). A significant difference between human and non-human actors lies in the process that underlies their task execution. Non-human actors usually execute an action only if a pre-determined condition is met, and only if the entire system is algorithmic or deterministic in nature. In contrast, human actors act in a non-deterministic, or probabilistic, manner; they do not always act in a logical and algorithmic manner, so they are more flexible in achieving a certain goal than non-human actors are. This observation is consistent with Gibson's (1994) explanation that, despite the affordances (i.e., potential for action) offered by the environment, humans still preserve their autonomy and control (Gibson, 1994; Reed, 1996). However, the distinction blurs as non-human actors come to have more human-like characteristics, such as the use of fuzzy logic programs or deep learning algorithms.

Figure 1 shows how design principles are represented in an abstract domain and interpreted and used in an instance domain. The figure is our adaptation of Lee, et al.'s (2011) design theorizing model by adding the roles of theorizer, implementer, user, and enactor. According to Lee, et al. (2011), design theorizing takes place through four activities: **abstraction** (extracting key ideas from problem instances and conceptualizing a problem class), **de-abstraction** (contextualizing a conceptual solution to address a specific problem instance), **solution search** (finding connections between our perception of a problem, our imagination of the desired changes and the possible actions that we can undertake in order to realize the changes), and **registration** (evaluating, modifying, and registering a solution instance in relation to the problem instance).

In figure 1, we show that a theorizer brings knowledge from an instance domain to an abstract domain, while an implementer applies abstract knowledge to an instance domain. While theorizer and implementer translate design knowledge from one domain to the other, user and enactor perform mostly in the instance domain. Most of the time, both user and enactor deal with an instance of artifact to address a problem instance in order to achieve specific goals. In principle, all these actors could be human or non-human, although in the majority of the examples we consider they are human. Where it is necessary to make a distinction, we refer to "non-human actors." The following is a summary of the roles that formulations of design principles should distinguish:

1) *Implementer*, who applies the abstract specification to the concrete instance domain.
2) *Recipient User* (or simply *user*), for whom the aim is to be achieved.
3) *Enactor*, who performs actions as part of the mechanisms that are used to accomplish the aim. When there is decomposition, an enactor may also be a recipient user of an artifact at a lower (i.e., more detailed) level.
4) *Theorizer*, who captures the abstract design knowledge from a concrete instance domain for use in research and subsequent applications.

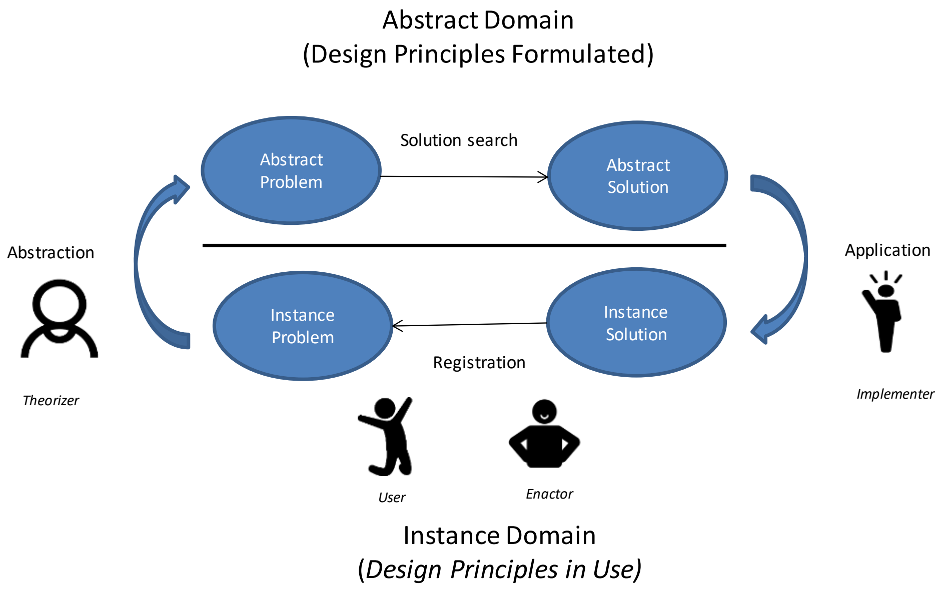

Figure 1: Design Principles in Use (adapted from Lee, Baskerville, and Pries-Heje, 2011)

5.2 Complexity and Decomposition

IT-based artifacts are often viewed as complex systems, explained as systems "made up of a large number of parts that have many interactions" (Simon, 1996, pp. 183-184). Simon (1996) also discusses how such complex systems are often viewed in terms of hierarchy to aid analysis and understanding, with systems at one level and their subsystems at a lower level. These ideas can be extended to the conceptualization of design principles that deal with complex systems and the level of generality at which a design principle can be considered. Scholars have dealt with this issue in several ways. Van Aken (2004, p. 238) saw technological rules as "mid-range theories of practice" and pointed out that a practitioner (the implementer) has to design a specific intervention based on his or her own experience, knowledge of the local context, and knowledge of the technological rule. On the other hand, some authors have advanced more general design principles. For example, Denning and Martell (2015, p. 200) proposed "Align the design [of an interactive system] with practices familiar to users" as a general design principle, while Shneiderman and Plaisant (2005, p. 74) proposed "Design dialogs to support closure" as one of the eight golden rules for interface design.

Both of these examples come from textbooks where there is considerable accompanying text to explain what the principle means in more detail, and where the audience for the texts is defined (e.g., as practitioners and researchers, not the general public). Shneiderman and Plaisant (2005) noted that, while principles tend to be fundamental, widely applicable, and enduring, they also need clarification by explaining parts of the principle at a lower level, that is, through decomposition. Therefore, Shneiderman and Plaisant's (2005, p. 75) example has a subordinate principle: "Sequences of actions [dialogs] should be organized into groups with a beginning, middle and end."

The short form of the principle enhances memorability, as our personal experience with students using the text has shown. To enhance a principle's memorability further and capture its essence, it can also be given a title or label. Moody (2009, p. 761) offered principles for designing visual notations in software engineering and provided each principle with a short title that helps explain its nature, for example: "Principle of Semiotic Clarity: There should be a 1:1 correspondence between semantic constructs and graphical symbols."

In summary, design principles are used by implementers who apply them in practice and theorizers who use them to capture knowledge. The nature of these actors should be considered in the formulation of design principles, especially in terms of the principle's level of generality and whether decomposition to lower levels is needed to make it understandable by the intended audience. Providing a title or label for a design principle can assist in conveying the principle's main point.

5.3 Types of Causation: Affordances and Non-Determinacy

IT-based artifacts often provide varying levels of freedom regarding their use, as the designer provides the artifact with some features, but the eventual use of these features depends on the users and may vary considerably, as has been expressed in views that highlight humans' role in enacting IT-based tools (e.g., DeSanctis & Poole, 1994; Orlikowski, 1992). Well-known examples are the IT-based tools used in everyday work, such as word-processing software with features that increase the efficiency of working with text, or mobile devices with features that allow connectivity, navigation, and so on. In some cases, even what is termed *secondary design* might occur. (See Germonprez, Hovorka, & Collopy (2007)). The original design could be purposely "generative," that is, designed in such a way that extension of the original design is encouraged, as is the case with IT-based platforms. (See Yoo, Boland Jr, Lyytinen, & Majchrzak (2012), Yoo, Henfridsson, & Lyytinen (2010), and

Zittrain (2006).) What these IT-based artifacts have in common is that humans use them in specific contexts and that this use often provides unpredictable results, which is in stark contrast with the premise that a specific design will deterministically lead to an anticipated, measurable result such as improved performance or lowered costs.

Congruent with this tension is a long-standing debate in the information systems field about whether IT-based artifacts are deterministic or non-deterministic, a debate that is often informed by the sociological discussion about dualities like objectivism and subjectivism (e.g., Bourdieu & Wacquant, 1992; Burrell & Morgan, 1979). Traditional views of information systems research have a clear preference for a deterministic view, where technology does what it is expected to do, and where "variance" schemas predominate. Critical of this view, scholars in the 1990s moved toward the individual and her interpretation of information technology (e.g., DeSanctis & Poole, 1994; Orlikowski, 1992), but this view has also been criticized as being overly voluntarist and as downplaying the role of technology (e.g., Orlikowski, 2010; Scott & Orlikowski, 2013). More recently, scholars have sought a middle ground between voluntarism and determinism (e.g., D'Adderio, 2011; Fayard & Weeks, 2014; Hultin & Mähring, 2014; Leonardi, 2012). In this view, there *are* regularities, and IT-based artifacts are used in similar ways across context and time, such that we design information systems in certain ways and expect those systems to meet a certain purpose. Examples include Enterprise Resource Planning, Customer Relationship Management, and Decision Support Systems. Still, as humans are involved, there is always indeterminacy, and humans can always choose to do otherwise, leading to change (Leonardi, 2011).

The various literatures deal with the underlying ideas here from various perspectives. The philosophy of science has discussed the idea of "propensities" or "dispositions" to behave in a certain way. Popper (1965) was concerned with the ontic nature of entities' properties and the link between the use of subjunctives in language and entities' disposition to behave in certain ways. For example, if we observe a glass full of clear liquid and suppose that it is water, then we expect it to quench thirst, extinguish fire, and so on. Describing the liquid as water "entails innumerable subjunctives about the kinds of responses it would display under a wide variety of test conditions" (Fetzer, 2017, p. 16).

Putting a pair of rabbits in the backyard will likely, but not necessarily, lead to more rabbits, because of their disposition or propensity to behave in certain ways (Fetzer, 2017) (the propensity to be "generative").

Similar ideas have been discussed in the information systems field using the notion of "affordance," which has been an influential way of thinking about how humans interact with IT-based artifacts (e.g., Fayard & Weeks, 2014; Leonardi, 2011; Markus & Silver, 2008; Seidel, Recker, & vom Brocke, 2013; Zammuto, Griffith, Majchrzak, Dougherty, & Faraj, 2007). Affordances describe relationships between humans and technical objects and have been used to describe what potential actions technologies provide to users and groups of users in their context of use (Leonardi, 2011; Markus & Silver, 2008). Affordances are both dispositions of technologies *and* relationships between users and technologies (Fayard & Weeks, 2014), so they provide a middle path between technology determinism (technologies *provide* affordances) and non-determinism (the eventual enactment of affordances depends on the human actor, her capabilities, and the context of use).

In summary, we propose that design principles should be capable of accounting for the *deterministic nature* of technologies—that is, when certain mechanisms are expected *to achieve* particular aims (e.g., in the case of an algorithm that transforms input into output)—and the *affordances* that technologies provide to certain groups of users, which *allow for* an action with more or less regularity.

5.4 Means to Achieve Ends: Design Mechanisms

Existing ways to formulate the design of design principles are not consistent in how they describe the "means" (or activities) to achieve an aim or in how they describe whether human activity is part of those means or part of the aim itself. Such formulations have focused primarily on the means to achieve an aim in terms of actors doing something or using something in one action or a series of actions. In the previous section we explored how an artifact can provide an affordance or propensity for an outcome, rather than the artifact's achieving that outcome directly. These mechanisms can include both impersonal material factors as well as the interpretations and understandings of the actors involved. We use the term *mechanisms* to refer to both *direct* agency through human and or machine activity and to the achievement of outcomes via *affordances* that artifacts offer to actors. Formulations of design principles should account for this distinction.

5.5 Justification of Design Principles

The definitions of design principles vary with respect to the need for justificatory knowledge that provides a rationale or reason for believing that the principle has validity. For example, van den Akker (1999, p. 9) suggested including words like "because of arguments P, Q and R" as part of the principle.

Justificatory knowledge can take several forms. In some cases, design knowledge is developed, at least in part, deductively from prior knowledge. At the extreme end of the spectrum, Bunge (2009) saw as relevant to scientific research only grounded rules (design principles), that is, rules that are "based on a set of law formulas capable of accounting for its effectiveness" (Bunge, 2009, p. 148). As an example of deductive development, Moody (2009) developed a well-cited design theory on the physics of visual notations from a synthesis of both theory and empirical evidence. Some research methods that have been proposed for developing design theory also emphasize the role of prior descriptive theory. For example, Kuechler and Vaishnavi (2012) proposed the development of "design-relevant explanatory/predictive theory (DREPT)" that "formally captures the translation of general theory constructs from outside IS to the design realm" (p. 400). When design knowledge has been developed deductively even in part from descriptive theory, then there is a ready source of justificatory knowledge.

However, in some situations, justificatory knowledge is not so readily available, at least when the principles are first developed, as some design knowledge is developed in projects that involve trial-and-error and experimentation and that use the reasoning processes of induction and abduction, rather than deduction (see Fischer, Gregor, & Aier, 2012). Simon (1996, p. 16) pointed to the "skyhook-skyscraper" construction of a science from the roof down and used the example of the first time-sharing computer, where the developers had only fragments of theory to guide them and to predict what demands an environment of users would place on the new systems. Simon made the important point that the problem of building a complex IT-based system involves finding a structure that works by allowing the interconnected components of the system to work reliably; in such a situation, having an "adequate micro-theory of the natural laws that govern the system components … might indeed be simply irrelevant" (p. 19). The justification for principles in this case are convincing demonstrations that the principles work when applied in practice.

This section described in brief how justifications for design knowledge can be provided in various ways. The key take-away is that justificatory knowledge (a rationale) for a design principle should be provided if possible, although its form may vary.

6. A New Conceptual Schema for Design Principles

Our new conceptual schema is based on the analysis presented in the previous section. Figure 2 shows the schema in a diagrammatic form and Table 4 in a textual form.

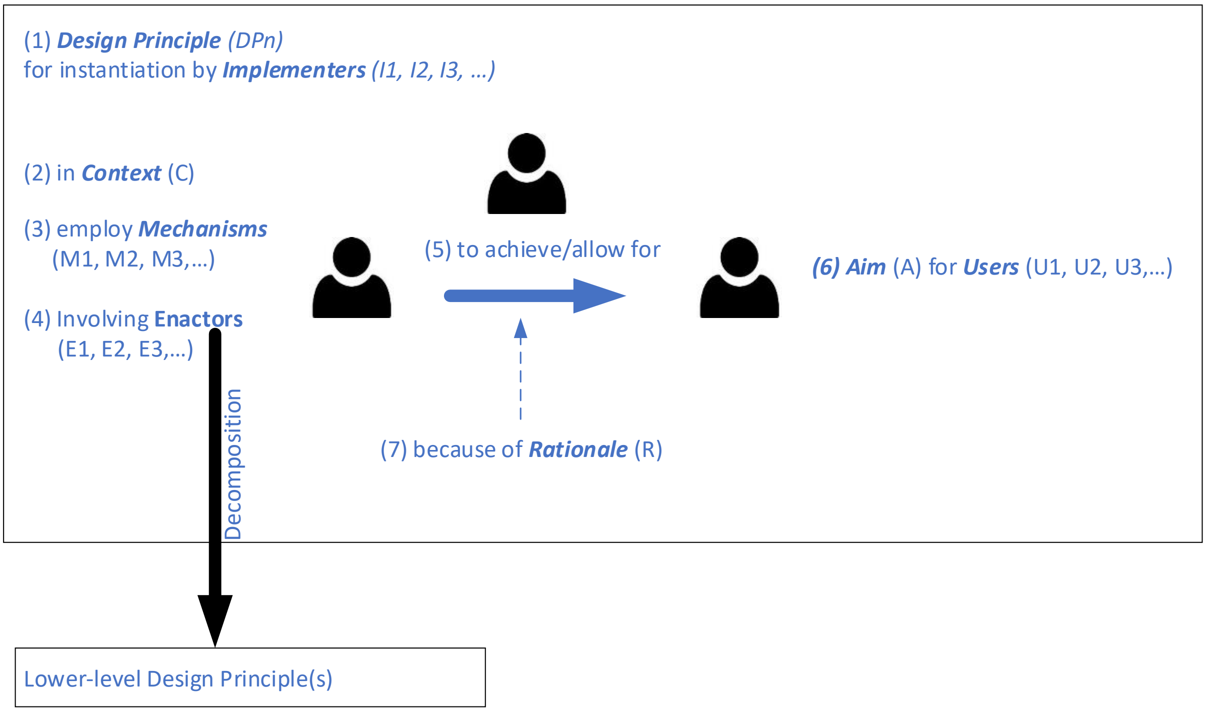

Figure 2: Diagram of the Design Principles Schema

Table 4. Components of the Design Principle Schema

Title: Design Principle Name **Structure**	**Components***
For **Implementer I** to achieve or allow **Aim A** for **User U**	Aim, Implementer, and User
in **Context C**	Context (Boundary conditions, implementation setting, further user characteristics)
Employ **Mechanisms** M1, M2, M3…. involving **Enactors** E1, E2, E3,…	Mechanisms (acts, activities, processes, form/architecture, manipulation of other artifacts) Subsidiary components/artifacts that can have their own design principles
because of **Rationale** R	Rationale Theoretical or empirical justification for the design principle

*Note: In many explications of design principles, some components are not made explicit.

The schema integrates the discussed prior definitions and retains their common components of Aim, Context, and Mechanism. Sometimes Rationale is also included, although it is not common in all the prior definitions.

Actors

The schema clarifies the role of all actors involved with the use of the design principle:

1. *Implementers* instantiate abstract specifications in a concrete design context.
2. *Users* are those whose aims are to be achieved.
3. *Enactors* perform actions as part of the mechanisms that are used to accomplish the aim. When there is decomposition, enactor may also be users if they rely on an artifact at a lower level.
4. *Theorizers* reflect on a concrete design context and try to capture the abstract design knowledge but are not part of the design principle. The theorizer and the implementer could be the same individual.

All of the actors who are involved can be either human or non-human. When the actor is a human, rather than a machine, indeterminacy is more likely. Tables 5 and 6 depict two examples of the distinctions among actors.

Table 5: Actors in the Medical Example

Actors	Instance
Implementer	Supervisor of medical staff member
User	Individual who receives treatment
Enactor	Individual who applies the cold compress. Also a user for the cold compress if that is regarded as an artifact at a lower level, and we are interested in how the artifact is constructed.
Theorizer	Medical researcher

Table 6: Actors in a Data Mining Algorithm

Actors	Instance
Implementer	The software developer
User	Program that receives retrieved results
Enactor	Algorithm that performs calculations
Theorizer	Computer scientist

Mechanisms

The schema includes actions, the use of other artifacts, and series of these actions and uses. The mechanisms have causal potential in that they either *lead to* or *allow users*—with the help of enactors that can themselves be systems that can be described in terms of design principles—to accomplish some aim. The schema thus recognizes that design principles can be more or less deterministic through the wording “achieves” or “allows for.”

Rationale

The schema recommends that each design principle include a “rationale,” that is, a justification for believing that the mechanisms will lead to achieving the aim.

Decomposition

The schema also shows that decomposition (Figure 2) can be used to provide detail about a design principle at a lower level to enhance implementers’ and enactors’ understanding.

7. Schema Application

We use three illustrations to demonstrate the applicability of our conceptual schema. The illustrations vary in terms of the nature of the IT-based artifact and the manner in which design knowledge is developed. We verified our analysis by communicating with authors of the second and third study we analyzed to understand and consider their views related to developing and formulating design principles. We asked them if we had depicted their studies appropriately and made changes where they saw misrepresentation.

Illustration 1: Designing Effective Visual Notations

Moody (2009) developed a design theory called the Physics of Notations to provide a scientific basis for the construction of visual notations in software engineering. He argued that, while visual notations are a key part of the language of software engineering, language's visual representation has been undervalued compared to its semantic understanding. In an effort to bridge this gap, he focused on the physical and perceptual properties of notations (syntax), rather than their logical (semantic) properties and showed the components of the design theory explicitly using Gregor and Jones' (2007) framework. The theory as a whole has nine design principles; Table 7 shows the first of these principles in terms of the new schema. The principle is not decomposed but has accompanying explanatory text that the intended implementers of the design theory, researchers, and designers in software engineering can understand. The article has been well cited in many fields. We inferred the human actors' roles from the text.

Table 7: Principle of Semiotic Clarity (from Moody, 2009)

Design PrincipleTitle	Principle of Semiotic Clarity
Aim, Implementer, and Users	For designers and researchers (implementers) "to design cognitively effective visual notations" (p. 773) (aim) for use by diagram creators and diagram users (users)...
Context	in software engineering…,
Mechanism	ensure there is a 1:1 correspondence between semiotic constructs and graphical symbols...
Rationale	because doing so avoids the anomalies of symbol redundancy, symbol overload, symbol excess, and symbol deficit, based on theory, including Goodman's (1968) theory of symbols.

Illustration 2: Designing for E-Government in a Developing Country

The second illustration is taken from Gregor et al.'s (2014) work on an action design research project that focused on a change strategy for e-government in a least developed country. The so-called "sweet spot" strategy was formulated to deal with the change's identified barriers, decision-makers' lack of fundamental knowledge and understanding of IT and lack of awareness of the strategic use and implications of IT systems for government business processes. The design principles sought to help fill the knowledge gap in e-government in a systematic way to achieve successful adoption of a broader e-government strategy for the public sector. The whole project was an intervention in which IT-based artifacts were part of the aim and the mechanisms for achieving the aim. The highest-level design principle was the "sweet spot" principle (Table 8).

Table 8: Principle of Sweet-Spot Change Strategy

Design Principle Title	Principle of Sweet Spot Change Strategy
Aim, Implementer, and User	To allow a change agent (enactor) to facilitate the uptake of e-government (aim) by public sector agencies (users) …
Context	In a least developed country with high power distance, political instability, and uncertainty,…
Mechanism	identify and act on the sweet spot(s) …
Rationale	because acting on a "sweet spot" can quickly deliver an effect or unlock a process of further rapid change with comparatively little effort, which is congruent with work on points of leverage and feedback loops in systems dynamics.

This general design principle has two mechanisms: identify sweet spot(s) and act on the sweet spot(s). Identifying sweet spot(s) involves several mechanisms as well, as the principle of local knowledge portrayed in Table 9 shows. Table 9 shows that a higher-level mechanism can be decomposed into separate lower-level mechanisms.

Table 9: Principle of Local Knowledge

Design Principle Title	Local Knowledge
Aim, Implementer, and User	To allow change agents (enactors) to identify sweet spot(s) and, thus, facilitate the uptake of e-government (aims) by public sector agencies (users)…
Context	in a least developed country with high power distance, political instability, and uncertainty,…
Mechanism	ensure the change agents have local knowledge, which is likely to occur only when the team includes one or more team members who are natives of the country....
Rationale	because "the issue that is underlying other inhibitors is …more often recognizable by members of the culture or region than by outsiders, no matter how earnest they are" (p. 665). The principle is congruent with Rogers' (1995) diffusion of innovation theory and the nature of change agents.

Illustration 3: Applying Text Analytics in Organizations

The third illustration is taken from Muller, Junglas, Debortolli, and vom Brocke's (2016) work on the use of text analytics in customer service management. The authors seek to provide simple, effective solutions that tech-savvy business people can use. The solution addressed how organizations can make sense of unstructured textual data (e.g., content of streams of incoming service requests) so they can understand their customers' problems and improve their customer service processes. In this case, the recommendations from the study were presented as "lessons learned," rather than as "design principles," yet they followed a similar pattern (Table 10).

Table 10: Principle of Business Positioning of Text Analytics

Design Principle Title	Principle of Business Positioning of Text Analytics
Aim, Implementer and User	To allow customer service managers (business users) in an organization (users) to understand customers' problems and improve customer service processes (aim),…
Context	when text mining is used to examine streams of incoming service requests, the business users are tech-savvy, they can learn the necessary skills, and the analytics tools are relatively easy to use,...
Mechanism	"Position text analytics in business units, not IT" (p. 255)
Rationale	because "analytical projects are less about rolling out IT tools and more about understanding how these tools might be used for creating business value. Business users know best which questions to ask, which datasets to explore and how to translate insights into actions" (pp. 255-256).

8. Discussion and Implications

This paper develops a schema for design principles to increase the effectiveness of formulating design principles and help in building cumulative bodies of design knowledge in information systems. Our analysis suggests that any formulation of design principles must attend to:

(1) the roles of the actors involved in developing and using these design principles,

(2) descriptions of complex IT-based artifacts that require design principles to allow for decomposition,

(3) representing various types of artifact-based actions, from affording user action to performing actions,

(4) the means to achieve the ends envisioned by the design principles, and

(5) the option of providing rationales that justify their formulation.

We use these five key issues to reflect on our research question: *How should design principles for technology-based artifacts in socio-technical systems be presented so that they are understandable and useful in real-world design contexts?* To do so, we discuss how our suggested framework contributes to the constituents of socio-technical artifacts, design practice and the practice of DSR by enhancing the understanding and usefulness of design principles for implementers, and the evaluation of such artifacts in DSR.

8.1 Design Principles and the Constituents of Socio-Technical Artifacts

We analyzed a number of publications that have described interventions or artifact designs from design domains including artificial intelligence, public policy, information systems, HCI, visual design, and international development and illustrated the schema's applicability by means of published design principles. Considering each of the five key issues in our formulation of design principles helps us to decompose important aspects of the formulation of knowledge about socio-technical artifacts in general terms (cf. Lee, Thomas, & Baskerville, 2015); even though the proposed structure for formulating design principles is similar to those of "veteran" concepts, such as technological rules (Bunge, 1967) and technical norms (Niiniluoto, 1993), it helps unpack these extant concepts in five ways.

First, with regard to the recipients of design principles (those who implement systems and those who enact them), we distinguish between the notion of effectiveness in formulating design principles in terms of completeness and the notion of effectiveness in terms of validity. A complete design principle spells out its aim, context, mechanism, and (if applicable) rationale and considers the roles of stakeholders in the relationships among these elements. What is important is their effectiveness from the point of view of stakeholders—implementers, users, enactors, and theorizers. For instance, implementers in business settings will implement only those design principles whose validity has been supported by repeated tests showing their prescriptive accuracy, so theorizers must provide evidence that shows their principles' prescriptive accuracy under defined boundary conditions (Seidel and Watson 2014).

Second, we consider the action that non-human agents perform (in the sense of an "algorithmic agent") and argue that design principles in information systems are design principles about socio-technical systems that involve both human and machine actors. Thus, we contribute to debates on the role and interplay of human and machine agencies in socio-technical assemblages (e.g., Leonardi, 2011). Considering machine agents in design principles gains importance, as artificial intelligence and related methods such as machine learning, pattern recognition, and evolutionary algorithms have increasingly become part of socio-technical assemblages. Thus, we add clarity to how to distinguish and consider the material component (in terms of machine action) from the human component (in terms of user action) in designing IS artifacts.

Third, we account for the often non-deterministic nature of IT artifacts when human actors use them, as the effects that result from using socio-technical artifacts occurs as humans enact the artifacts in certain ways. We base our formulation of design principles on concepts that include the non-deterministic nature of artifacts (Faulkner & Runde, 2013; Leonardi, 2011; Markus & Silver, 2008; Strong et al., 2014). We highlight how design principles must account for the relationships between artifacts' features and their users, thereby stressing what the technology can be used under certain boundary conditions. While some researchers have asserted that the concept of affordances provides a suitable lens through which to study information systems' design (Markus & Silver, 2008), little guidance has been provided about how this can be done. We propose a formulation of design principles that can be used to develop prescriptive knowledge that takes into account the non-deterministic use of IT-based artifacts.

Fourth, we address the complexity of design principles for complex IT-based artifacts compared with the simpler forms that have been proposed to date (Bunge, 1967; Niiniluoto, 1993), which requires disentangling an overall aim from the mechanisms (acts, activities, processes, and forms/architectures) that are proposed to achieve it. Moreover, the schema clarifies the role of all actors involved with the use of the design principle (implementer, user, enactor, and theorizer), an analysis that has not been attempted before.

Finally, our conceptual schema considers that the boundary conditions (situation) relate not only to the implementation setting but also to the users' characteristics. One may need to think differently when designing an online teaching tool for school children than one does when designing a similar tool for executive MBA candidates. This perspective helps to clarify the boundary conditions of design knowledge in broader terms (cf., Gregor and Jones, 2007).

8.2 Design Principles in Design Practice and Design Research Practice: Understanding and Usefulness

One of DSR's key objectives is to complement work that seeks to understand, explain, and sometimes predict the development, use, and impact of information systems and related socio-technical artifacts in organizations and other social contexts (Baskerville et al., 2018; Hevner et al., 2004; Kuechler & Vaishnavi, 2012). The purpose of DSR is to develop prescriptive knowledge that may or may not build on explanatory and predictive knowledge and that needs to be conveyed in one way or another (Baskerville & Pries-Heje, 2010; Gregor & Hevner, 2013; Kuechler & Vaishnavi, 2012). It is against this background that we set out to devise a simple, understandable, and useful schema that helps us in formulating prescriptive knowledge.

The first requirement is that the design principles attend to the roles of human actors who are involved in their formulation and use. The distinction between the implementer, who instantiates the abstract specification of the design principle, and the user, who enacts that instantiation to bring about a goal, is of particular importance, as it requires the design theorizer, who develops the design principle, to formulate the principle in such way that considers both perspectives. This distinction avoids the development of design principles that provide guidance for implementers without considering the user's perspective and, therefore, without considering the practical consequences of implementing and then using the IT-based artifact in organizational and other practice. Research has highlighted the need to distinguish human roles. For example, in providing a framework for the use of explanations in data-driven document classification, Martens and Provost (2014) showed the importance of distinguishing the roles of people who interact with a decision system: in their case, developers, managers, and customers.

The second requirement is that that design principles attend to the complexity of IT-based artifacts through decomposition. The suggested formulation of design principles allows for formulating design principles at various levels of granularity. Design principles are abstractions (Gregor & Hevner, 2013), so they should be formulated in such way that their recipients can readily understand them, thus ensuring their usefulness. Our examples show how design principles can be devised that are sufficiently simple for enactors and users.

Third, for a formulation of design principles to be useful in a variety in contexts, they must accommodate both human and non-human actors, a requirement that becomes increasingly important with the advent of more distributed systems based on Internet of Things (IoT) technologies, where human and non-human actors are part of increasingly complex networked systems. Design principles that fail to recognize that human actors are part of socio-technical systems cannot be applied to many contemporary and emergent IT-based situations.

Fourth, attending to means in terms of acts, activities, processes, architectures, and artifact manipulations immediately opens up the formulation of design principles to a wide arena of applications, making them useful for a variety of IT-based artifacts and associated situations.

Fifth, the optional consideration of justificatory knowledge allows the suggested formulation of design principles to be applied when the design theorizer (and, consequently, the implementer and the user) can draw on a body of explanatory and predictive knowledge as well as when no such knowledge is available. Useful prescriptions can be conceived even without understanding the causes, although prescriptive accuracy benefits when we understand the underlying causal relationships.

Finally, the schema helps capture the essence of a design in a concise and straightforward manner. At the same time, however, it conveys comprehensive knowledge about the design essence by providing several contact points for implementers. Implementers will benefit from this form of communication in identifying similarities or associations between their design situation and that described in the schema (e.g., boundary condition, aim, user). Compare this approach with the common one that begins with reading individual design principle, understanding its scope, and guesstimating missing information (cf. Chandra Kruse, Purao, and Seidel, 2016). Imagine the target users are people with disabilities but they are still encouraged to operate a system for a specific goal. With the schema, implementers must not search for important pieces of information that is usually presented in different sections of a report and apply hermeneutics. Instead, they will find the pieces under “user", "mechanism", and "boundary condition" in our schema.

Taken together, addressing these issues renders our formulation of design principles a contribution to the ongoing effort in IS design knowledge production (cf. Baskerville, Kaul, & Storey, 2015). Discussions at conferences and workshops suggest that many scholars perceive that it is time to move from a debate that focuses on methodology and associated contributions to conducting DSR and developing a cumulative tradition. This view is also reflected the *Journal of the Association for Information Systems*’ recent editorial on the accumulation and evolution of knowledge in design science research (see vom Brocke, et al., 2019). We argue that a simple formulation of design principles that is open to a wide array of phenomena involving human and non-human actors and a variety of types of IT-based systems supports this next step in the development of DSR as a central element in the canon of information systems research and research on IT-based systems in other fields.

8.3 Design Principles and their Evaluation

Evaluation is a key component of DSR (Hevner et al., 2004; Venable et al. 2016). A strategic process for evaluating DSR studies involves explicating goals, selecting a strategy for evaluation, determining the properties to be evaluated, and designing individual evaluation episodes (Venable et al., 2016). The proposed schema for formulating design principles supports these steps.

By explicitly considering roles of the key stakeholders involved in formulating design principles, the schema allows DSR scholars to test a set of generated design principles in terms of their usability for a variety of user groups. Two key questions concern whether the design principles are understandable and useful for implementers and whether they are useful for achieving the goals of the users who enact the instantiations that result from applying the design principles. That is, the schema's distinction among stakeholders facilitates an evaluation that considers a design principle's appropriateness in both guiding implementation and deployment (e.g., Seidel et al., 2018) and in accomplishing organizational goals.

The proposed schema for formulating design principles also supports their evaluation by ensuring they are formulated in a way that accommodates decomposition so a set of design principles can be evaluated at multiple levels of abstraction. Generally, evaluating design principles at finer levels of granularity increases control and internal validity but at the cost of considering contextual factors that originate in the composition of the overall modular system and in its application in real-world contexts. The proposed formulation of design principles allows for several degrees of freedom in evaluating sets of design principles, as the team of researchers can chose among levels of abstraction and, therefore, also among levels of granularity.

Further, the proposed schema's consideration of types of actors is inclusive of both deterministic effects and probabilistic effects. For instance, for an algorithm, the researchers may conduct a set of experiments to determine the algorithm's performance under conditions including differences in inputs and hardware. The performance of a socio-technical artifact that, for instance, involves both a human actor and the actor's use of a technology artifact may be evaluated in an experiment with a set of human subjects, controlling for demographic aspects like gender and age as well as variables like experience.

With regard to an evaluation's consideration of means (e.g., mechanisms to bring about a certain result), the proposed formulation of design principles is open to a variety of evaluative scenarios that fit the respective means. An algorithm may be tested through a set of experiments, while a complex socio-technical artifact may be evaluated in a real-world context, perhaps through an action design research study (Sein et al., 2011) that may lead to subsequent refinements of the proposed set of design principles.

Our formulation also requires the definition of boundary conditions, and every evaluation of the resulting design principles must consider these boundary conditions. Design is a contextual activity (Dorst & Cross, 2001), but design principles may be applicable across contexts and time, although whether such is the case requires repeated application and testing of these principles in a variety of contexts.

8.4 Limitations and Future Research

This study has several limitations. We did not test our proposed formulation of design principles empirically and did not move beyond a proof of concept (comparable with the formulation of other approaches including Peffers et al., 2007, Gregor 2006, Gregor and Jones 2007, Hevner et al. 2004). Moreover, while we expect that the proposed conceptual foundation will provide useful guidance in the practice of DSR, we cannot claim it to be the only or even the best solution.

Notably, while our framework is a prescription and thus tells us how something should or could be done, it is tentative—as any scientific contribution is tentative. We thus understand that by suggesting this formulation of design principles we are rather contributing to the discourse on how prescriptive knowledge can be formulated than claiming to conclude this discourse by formulating a single way in which prescriptive knowledge should be formulated. The key here is that any framework—including the one that we are proposing—needs to be evaluated by how well it helps its users accomplish their goals—in our case this is the effective and efficient formulation of prescriptive knowledge about socio-technical artifacts. We can identify opportunities for future research to continue this discourse.

First, future research should investigate the schema empirically for completeness, validity, and other desirable properties, such as understandability. Our solution is conceptual as well as prescriptive in nature—we suggest how design principles should be formulated—so it will have to be evaluated by the same measures that we suggest for evaluating design principles. The criteria we propose could inform such empirical work.

Second, it will be particularly interesting to look at understanding how informal feedback we received about the usefulness of our schema could be extended to a formal evaluation. Future research could also investigate how DSR scholars in particular, and IS/IT designers in general, apply the framework in situ. We believe that applying the framework is different from following a recipe in a stepwise manner, and it is not intended to straight-jacket researchers who embark on developing prescriptive knowledge. It is therefore important to observe how designers understand and act upon the knowledge prescribed in the conceptual schema.

Finally, while we are expecting that the suggested formulation of design principles can lead to accomplishing desired goals, it is important to highlight that such abstract formulation necessarily applies to a broad variety of design situations that require a variety of different means. Future research will thus have to explore the boundary conditions under which the suggested formulation is useful.

9. Conclusions

To have maximum societal impact, the IS discipline must turn explanation and prediction into prescription (Bichler, Heinzl, & Winter, 2015; Seidel & Watson, 2014) and focus on applicable knowledge (Johannesson & Perjons, 2014; Rosemann & Vessey, 2008) and intervention (Davenport & Markus, 1999) and how such knowledge should be represented, communicated, and cumulatively built. IS development in organizations engages various parties and users and employs processes that include requirements engineering to identify the actions or processes that a system should support. Project failure that is due to poorly communicated requirements is a main challenge in IS development, so clearly formulated design principles are expected to support the process of developing and implementing IS artifacts and, thus, to improve practice in digital innovation.

It is against this background that we rigorously derived a schema for specifying design principles that is appropriate for research practice and that supports the application of design knowledge to professional practice. The context is that of IT-based systems that include human and non-human actors (i.e., socio-technical systems). Our conceptual schema attends to central issues in the formulation of prescriptive knowledge about IT-based artifacts, specifically, issues related to handling their complexity through decomposition, considering that the mechanisms for achieving aims can be accomplished by both human and non-human enactors, distinguishing deterministic and probabilistic types of causation, and allowing the justification of design principles. Thus, we provide a nuanced understanding of the notion of actors in design principle formulation and the nature of the mechanisms used to achieve aims, and we highlight that the generalizability of any design principle is limited to the contexts that share its boundary conditions.

We expect societal and scientific advancement from an evolving and accumulative process of forming a prescriptive body of knowledge for the design of IT-based artifacts. We are interested to see how we, as a discipline, adopt standards of formulating prescriptive knowledge in our editorial and review processes and hope that our work contributes to an important debate that is ultimately about the applicability and practical relevance of our discipline.

References

Alexander, C., Ishikawa, S., & Silverstein, M. (1977). *Pattern Languages: Towns, Buildings, Constructions* (Vol. 2). Oxford, UK: Oxford University Press.

Arthur, W. B. (2009). *The nature of technology: What it is and how it evolves*: New York, NY: Simon and Schuster.

Baskerville, R., Baiyere, A., Gregor, S., Hevner, A., & Rossi, M. (2018). Design Science Research Contributions: Finding a Balance between Artifact and Theory. *Journal of the Association for Information Systems, 19*(5): 358-376.

Baskerville, R. L., Kaul, M., & Storey, V. C. (2015). Genres of Inquiry in Design-Science Research: Justification and Evaluation of Knowledge Production. *MIS Quarterly*, 39(3), 541-564.

Baskerville, R., & Pries-Heje, J. (2010). Explanatory design theory. *Business & Information Systems Engineering, 2*(5), 271–282.

Bichler, M., Heinzl, A., & Winter, R. (2015). Practice impact of IS research. *Business & Information Systems Engineering, 57*(2), 87-89.

Bider, I., Johannesson, P., & Perjons, E. (2013). Design science research as movement between individual and generic situation-problem–solution spaces. In Designing Organizational Systems (pp. 35-61). Berlin, Germany: Springer.

Bourdieu, P., & Wacquant, L. J. (1992). *An invitation to reflexive sociology*. Chicago, IL: University of Chicago Press.

Bunge, M. (1967). *Scientific Research I, The Search for System*. New York, NY: Springer Verlag.

Bunge, M. (2009). *Philosophy of Science: From Explanation to Justification* (Vol. 2). New Brunswick, NJ: Transaction Publishers.

Burrell, G., & Morgan, G. (1979). *Sociological paradigms and organisational analysis. Elements of the sociology of corporate life*. London, UK: Heinemann Educational Books.

Burton-Jones, A. and Volkoff, O. (2017). How can we develop contextualized theories of effective use? A demonstration in the context of community-care electronic health records. *Information Systems Research, 28*(3), 468-489.

Butler, T., & Murphy, C. (2007). Understanding the Design of Information Technologies for Knowledge Management in Organizations: A Pragmatic Perspective. *Information Systems Journal, 17*, 143–163.

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 (HICSS)* (pp. 4039–4048). Kauai, USA.

Chandra Kruse L., Seidel S., Purao S. (2016) Making Use of Design Principles. In: Parsons J., Tuunanen T., Venable J., Donnellan B., Helfert M., Kenneally J. (eds) *Tackling Society's Grand Challenges with Design Science*. DESRIST 2016. Lecture Notes in Computer Science, vol 9661. Springer, Cham

Chen, H., Chiang, R. H. L., & Storey, V. C. (2012). Business Intelligence and Analytics: From Big Data to Big Impact. *MIS Quarterly, 36*(4), 1165-1188.

D'Adderio, L. (2011). Artifacts at the centre of routines: Performing the material turn in routines theory. *Journal of Institutional Economics, 7*(02), 197-230.

Dahlbom, B. (2002). The Idea of an Artificial Science, in B. Dahlbom, S. Beckman, and G. Nilsson (eds.), *Artifacts and Artificial Science,* Stockholm, Sweden: Almqvist & Wiksell International, pp. 9-44.

Davenport, T. H., & Markus, M. L. (1999). Rigor vs. relevance revisited: Response to Benbasat and Zmud. *MIS Quarterly, 23*(1), 19-23.

Denning, P., & Dargan, P. (1996). Action-centered design. *Bringing design to software*, 105-119.

Denning, P. J., & Martell, C. H. (2015). *Great principles of computing*. Cambridge, MA: MIT Press.

DeSanctis, G., & Poole, M. S. (1994). Capturing the complexity in advanced technology use: Adaptive structuration theory. *Organization Science, 5*(2), 121-147.

Dorst, K. and N. Cross (2001). "Creativity in the design process: Co-evolution of problem–solution." *Design Studies, 22*(5): 425-437.

Faulkner, P., & Runde, J. (2013). The Social, the material, and the ontology of non-material technological objects. *MIS Quarterly, 37*(3), 803-818.

Fayard, A.-L., & Weeks, J. (2014). Affordances for practice. *Information and Organization, 24*(4), 236-249.

Fetzer, J. (Fall 2017 Edition). "Carl Hempel", *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.), URL = <https://plato.stanford.edu/archives/fall2017/entries/hempel/>. (Accessed May 2018).

Fischer, C., Gregor, S., & Aier, S. (2012). *Forms of discovery for design knowledge.* Paper presented at the the 20th European Conference on Information Systems (ECIS 2012), Barcelona, Spain.

Fowler, M. (2007). Accounting patterns. *Analysis Patterns: Reusable Object Schemas*.

Gamma, E. (1995). *Design patterns: Elements of reusable object-oriented software*: Pearson Education India.

Germonprez, M., Hovorka, D., & Collopy, F. (2007). A Theory of Tailorable Technology Design. *Journal of the Association for Information Systems, 8*(6), 351–367.

Goes, P. B. (2014). Editor's comments: design science research in top information systems journals. *MIS Quarterly, 38*(1), iii-viii.

Goodman, N. (1968). *Languages of art: An approach to a theory of symbols.* Indianapolis, IN: Bobbs-Merrill Co.

Gregor, S. (2006). The nature of theory in information systems. *MIS Quarterly, 30*(3), 611-642.

Gregor, S. (2017). On Theory In R. Galliers & M. K. Stein (Eds.), *The Routledge Companion to Management Information Systems.* (pp. 57-72). London, UK: Routledge.

Gregor, S., & Benbasat, I. (1999). Explanations from intelligent systems: Theoretical foundations and implications for practice. *MIS Quarterly*, *23*(4), 497-530.

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

Gregor, S., Imran, A., & Turner, T. (2014). A 'sweet spot'change strategy for a least developed country: leveraging e-Government in Bangladesh. *European Journal of Information Systems, 23*(6), 655-671.

Gregor, S., & Jones, D. (2007). The anatomy of a design theory. *Journal of the Association for Information Systems*, *8*(5), 313-335.

Gregor, S., Müller, O., & Seidel, S. (2013). Reflection, Abstraction, and Theorizing in Design and Development Research. *Proceedings of the European Conference on Information Systems*, Utrecht, The Netherlands.

Heidegger, M. (1993). *Martin Heidegger: Basic Writings from Being and Time (1927) to The Task of Thinking (1964)*. New York, NY: Harper Perennial Modern Classics.

Houkes, W. (2009). The nature of technological knowledge. *Philosophy of technology and engineering sciences*, *9*, 309-350.

Houkes, W., & Vermaas, P. (2004). Actions versus functions: A plea for an alternative metaphysics of artifacts. *The Monist, 87*(1), 52-71.

Hultin, L., & Mähring, M. (2014). Visualizing institutional logics in sociomaterial practices. *Information and Organization, 24*(3), 129-155.

Iivari, J. (2007). A Paradigmatic Analysis of Information Systems As a Design Science. *Scandinavian Journal of Information Systems, 19*(2), 39-64.

Johannesson, P., & Perjons, E. (2014). *An Introduction to Design Science.* Berlin, Germany: Springer.

Kenny, D. A. (1996). Schemas of non-independence in dyadic research. *Journal of Social and Personal Relationships*, *13*(2), 279-294.

Ketter, W., Peters, M., Collins, J., and Gupta, A. (2016). A multiagent competitive gaming platform to address societal challenges. *MIS Quarterly, 40*(2), 447-460.

Kuechler, W., & 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.

Law, J., & Urry, J. (2004). Enacting the Social. *Economy and Society, 33*((3), 390-410.

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, A. S. (2010). Retrospect and prospect: information systems research in the last and next 25 years. *Journal of Information Technology*, 25(4), 336-348.

Lee, J.S., Pries-Heje, J., & Baskerville,go R. (2011). Theorizing in Design Science Research. In: Jain H., Sinha A.P., Vitharana P. (eds). *Service-Oriented Perspectives in Design Science Research*. DESRIST 2011. Lecture Notes in Computer Science, vol 6629. Springer, Berlin, Heidelberg

Leonardi, P. M. (2011). When flexible routines meet flexible technologies: Affordance, constraint, and the imbrication of human and material agencies. *MIS Quarterly, 35*(1), 147–168.

Leonardi, P. M. (2012). Materiality, sociomateriality, and socio-technical systems: What do these terms mean? How are they related? Do we need them? In P. M. Leonardi, B. A. Nardi, & J. Kallinikos (Eds.), *Materiality and organizing: Social interaction in a technological world* (pp. 25-48). Oxford: Oxford University Press.

Lin, Y., Chen, H., Brown, R. A., Li, S., & Yang, H. (2017). Healthcare predictive analytics for risk profiling in chronic care: A Bayesian multitask learning approach. *MIS Quarterly*

Lukyanenko, R. & Parsons, J. (forthcoming). Design theory indeterminacy: What is it, how can it be reduced, and why did the polar bear drown?. *Journal of the Association for Information Systems*

Lukyanenko, R., Parsons, J., and Wiersma, Y. (2014). The IQ of the crowd: Understanding and improving information quality in structured user-generated content. *Information Systems Research, 25*(4), 669-689.

March, S. T., & Storey, V. C. (2008). Design science in the information systems discipline: an introduction to the special issue on design science research. *MIS Quarterly*, 725-730.

Markus, M. L., & Silver, M. S. (2008). A Foundation for the Study of IT Effects: A New Look at DeSanctis and Poole's Concepts of Structural Features and Spirit. *Journal of the Association for Information Systems, 9*(10), 609–632.

Menon, S., and Sarkar, S. (2016). Privacy and big data: Scalable approaches to sanitize large transactional databases for sharing. *MIS Quarterly, 40*(4), 963-981.

Mertens, D. & Provost, F. (2014). Explaining data-driven document classifications. *MIS Qarterly, 38*(1), 73-99.

Moody, D. L. (2009). The “physics” of notations: toward a scientific basis for constructing visual notations in software engineering. *Software Engineering, IEEE Transactions on, 35*(6), 756-779.

Müller, O., Junglas, I., Debortoli, S., & vom Brocke, J. (2016). Using text analytics to derive customer service management benefits from unstructured data. *MIS Quarterly Executive, 15*(4), 243-258.

Niehaves, B., & Ortbach, K. (2016). The inner and the outer model in explanatory design theory: The case of designing electronic feedback systems. *European Journal of Information Systems, 25*(4), 303–316.

Niiniluoto, I. (1993). The Aim and Structure of Applied Research. *Erkenntnis, 38*, 1-21.

Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., & Sinz, E. J. (2011). Memorandum on design-oriented information systems research. *European Journal of Information Systems, 20*(1), 7–10.

Orlikowski, W. J. (1992). The duality of technology: Rethinking the concept of technology in organizations. *Organization Science, 3*(3), 398-427.

Orlikowski, W. J. (2010). The sociomateriality of organisational life: Considering technology in management research. *Cambridge Journal of Economics, 34*, 125-141.

Pawson, R., & Tilley, N. (1997). *An introduction to scientific realist evaluation*. London, UK: SAGE.

Peffers, K., Tuunanen, T., & Niehaves, B. (2018). Design science research genres: Introduction to the special issue on exemplars and criteria for applicable design science research. *European Journal of Information Systems, 27*(2), 129-139.

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.

Plsek, P., Bibby, J., & Whitby, E. (2007). Practical methods for extracting explicit design rules grounded in the experience of organizational managers. *The Journal of Applied Behavioral Science, 43*(1), 153-170.

Rai, Arun. (2017). Editor's comments: Diversity of Design Science Research. *MIS Quarterly, 41*(1), iii-xviii.

Rogers, E. M. (1995). *Diffusion of innovations* (4th ed.). New York, NY: Free Press.

Rosemann, M., & Vessey, I. (2008). Toward improving the relevance of information systems research to practice: The role of applicability checks. *MIS Quarterly, 32*(1), 1-22.

Scott, S. V., & Orlikowski, W. J. (2013). Sociomateriality—taking the wrong turning? A response to Mutch. *Information and Organization, 23*(2), 77-80.

Segerberg, K., Meyer, J. J., & Kracht, M. (2016). The Logic of Action. In E. N. Zalta, Nodelman, U., Allen, C., & Anderson, R.L. (Ed.), *Stanford Encyclopedia of Philosophy*. Stanford, CA: The Metaphysics Research Lab.

Seidel, S., Chandra Kruse, L., Szekely, N., Gau, M., & Stieger, D. (2018). Design Principles for Sensemaking Support Systems in Environmental Sustainability Transformations. *European Journal of Information Systems 27*(2): 221-247.

Seidel, S., Recker, J., & vom Brocke, J. (2013). Sensemaking and sustainable practicing: Functional affordances of information systems in green transformations. *MIS Quarterly, 37*(4), 1275-1299.

Seidel, S., & Watson, R. T. (2014). Improving the Societal Effectiveness of IS Research: The Pursuit of Prescriptive Accuracy. *Available at SSRN 2477917*.

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

Shneiderman, B., & Plaisant, C. (2005). *Designing the user interface*. Boston, MA.: Pearson.

Simon, H. A. (1996). *The sciences of the artificial*. Cambridge, MA: MIT Press.

Strong, D. M., Volkoff, O., Johnson, S. A., Pelletier, L. R., Tulu, B., Bar-On, I., . . . Garber, L. (2014). A Theory of Organization-EHR Affordance Actualization. *Journal of the Association for Information Systems, 15*(2), 53-85.

van Aken, J., Chandrasekaran, A., & Halman, J. (2016). Conducting and publishing design science research: Inaugural essay of the design science department of the Journal of Operations Management. *Journal of Operations Management, 47*, 1-8.

van Aken, J. E. (2001). Improving the relevance of management research by developing tested and grounded technological rules. *Eindhoven Centre for Innovation Studies, Eindhoven*.

Van den Akker, J. (1999). Principles and methods of development research. In J. J. H. v. d. Akker, Branch, R., Gustafson, K., Nieveen, N.M. & Plomp, T. (Ed.), *Design approaches and tools in education and training* (pp. 1-14): Springer.

Venable, J., et al. (2016). FEDS: a framework for evaluation in design science research. *European Journal of Information Systems 25*(1): 77-89.

vom Brocke, J., Winter, R., Hevner, A., Maedche, A. (2019). Accumulation and Evolution of Design Knowledge in Design Science Research – A Journey Through Time and Space. *Journals of the Association for Information Systems*.

von Wright, G. H. (1963). *Norm and Action: A Logical Enquiry*. London, UK: Routledge & Kegal Paul.

Walls, J. G., Widermeyer, 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 (JITTA), 6*(2), 43-58.

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. doi:10.1287/isre.3.1.36

Winter, R. (2008). Design science research in Europe. European Journal of Information Systems, 17(5), 470–475.

Winther, R. (2016). The Structure of Scientific Theories. In Zalta, E., Nodelman, U., Allen, C. and Perry, J. (Eds.). *Stanford Encyclopaedia of Philosophy.* URL: https://stanford.library.sydney.edu.au/archives/fall2016/entries/structure-scientific-theories/ (Accessed Dec. 2016)

Wong, N., Ray, P., Stephens, G., & Lewis, L. (2012). Artificial immune systems for the detection of credit card fraud: an architecture, prototype and preliminary results. *Information Systems Journal*, *22*(1), 53–76.

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

Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). Research commentary—the new organizing logic of digital innovation: an agenda for information systems research. *Information Systems Research*, *21*(4), 724-735.

Zammuto, R. F., Griffith, T. L., Majchrzak, A., Dougherty, D. J., & Faraj, S. (2007). Information technology and the changing fabric of organization. *Organization Science*, *18*(5), 749-762.

Zhang, X., Venkatesh, V., & Brown, S. A. (2011). Designing Collaborative Systems to Enhance Team Performance. *Journal of the Association for Information Systems, 12*(8), 556–584.

Zittrain, J. L. (2006). The generative internet. *Harvard Law Review*, 1974-2040.

Appendix A: Examining Design Principles Formulation in Information Systems Research Practice

The practice of formulating and specifying design principles in information systems (IS) design science research (DSR) with respect to their focus on the human user was investigated in a literature review. We conducted a first review in 2014[2] and we updated the review in February/March 2020.[3] The sample of this last review consisted of 69 articles based on a Google Scholar search of articles published in *European Journal on Information Systems, Information Systems Journal, Information Systems Research, Journal of Information Technology, Journal of Management Information Systems, Journal of Strategic Information Systems, Journal of the Association for Information Systems,* and *Management Information Systems Quarterly*.

Literature Search: Sample

In a first step, we performed a Google Scholar search using the following search string:

"Design Principle" OR "Design Principles" OR "Design Theory" source: "Journal Name"

We did not limit the time frame—our analysis thus includes articles that were published between the inception of the respective journals and when the search was performed. This exercise produced more than 500 articles from the eight journals. Our selection of articles for further analysis was based on whether the article proposed explicit design principles—we thus excluded articles that just referenced or used design principles that were published elsewhere. In our search, we did not consider further synonyms under which design principles might have been published. However, if an article that our search yielded included principles under different names (such as "principles of form and function"—as was the case in papers that formulate design theories), we considered this article and the respective design principles. The goal of our literature search was not comprehensiveness, but to produce a

[2] Chandra, L., Seidel, S., & Gregor, S. (2015). Prescriptive Knowledge in IS Research: Conceptualizing Design Principles in Terms of Materiality, Action, and Boundary Conditions. *Proceedings of the 48th Hawaii International Conference on System Sciences*, pp. 4039–4048.

[3] Note that we slightly adjusted the search process in the new review compared with our original review that was published in 2015. In our first review we included those articles that used design principles provided elsewhere, while in the updated process we considered only those papers that introduced new design principles. The sample of our second review is still significantly larger compared to that of the first review, which can be explained by a general uptake in developing design principles in the IS field—there are now simply more papers developing design principles.

sample that would provide a good overview of how design principles have been formulated in our field. Table A.1 shows an overview of our sample in terms of the number of articles retrieved for each of the eight journals.

Table A1: Sample for the Content Analysis

Journal	*n*
European Journal of Information Systems (EJIS)	14
Information Systems Journal (ISJ)	8
Information Systems Research (ISR)	7
Journal of Association of Information Systems (JAIS)	19
Journal of Information Technology (JIT)	3
Journal of Management Information Systems (JMIS)	7
Journal of Strategic Information Systems (JSIS)	2
Management Information Systems Quarterly (MISQ)	9
Total	69

Content Analysis

To analyze our sample of 69 papers, we applied a coding scheme we derived in the first review we conducted in 2014. This previous analysis had suggested that (1) some of the design principles focused attention on users' use of artifacts; (2) some talked mainly about the artifacts and little about the users; and (3) the remainder attended to both (i.e., focused on both artifact and action). We used this simple coding scheme as the basis for our analysis.

To this end, two of the authors coded each set of design principles using the identified three orientations and then compared their results to attain an inter-rater agreement. We decided to code sets of design principles instead of individual design principles (a set could also comprise a single design principle). Both raters agreed on the majority of the coding results. Differing views could be resolved through discussion. That is, the sets of design principles where categorized as either prescribing use (i.e., they are about user activity) or prescribing features (i.e., they are about the artifact), or both—lending evidence to the suitability of the three categories and supporting the results of our previous analysis in 2014 (see Table A2). The results showed that, out of 69 sets of design principles, 11 sets were about user-activity, 27 sets were about artifacts, and 31 sets were about both user-activity and artifact.

Table A2: Summary of Code per Design Principle Set

No.	Design Principle Set	Final Code	Inter-Rater Agreement	Reference
1	Design principles for text analysis of computer-mediated communication	**About artifact**	*Initial agreement*	Abbasi, A., & Chen, H. (2008). CyberGate: A design framework and system for text analysis of computer-mediated communication. MIS Quarterly, 32(4), 811-837.
2	Design principles for tailoring database training to end users	**About artifact**	*Initial agreement*	Ahrens, J. D., & Sankar, C. S. (1993). Tailoring database training for end users. MIS Quarterly, 17(4), 419-439.
3	Design principles for social recommender systems	**About both**	*Initial agreement*	Arazy, O., Kumar, N., & Shapira, B. (2010). A theory-driven design framework for social recommender systems. Journal of the Association for Information Systems, 11(9), 455-490.
4	Design principles for collaborative ERP systems	**About both**	*Initial agreement*	Babaian, T., Xu, J., & Lucas, W. (2018). ERP prototype with built-in task and process support. European Journal of Information Systems, 27(2), 189-206.
5	Design principles for IT-enabled knowledge management systems	**About user activity**	*Initial agreement*	Butler, T., & Murphy, C. (2007). Understanding the design of information technologies for knowledge management in organizations: a pragmatic perspective. Information Systems Journal, 17(2), 143-163.
6	Design principles for loosely coupling lightweight and heavyweight IT	**About artifact**	*Resolved*	Bygstad, B. (2017). Generative innovation: a comparison of lightweight and heavyweight IT. Journal of Information Technology, 32(2), 180-193.
7	Design principles for blockchain-based sensor data protection system	**About artifact**	*Initial agreement*	Chanson, M., Bogner, A., Bilgeri, D., Fleisch, E., & Wortmann, F. (2019). Blockchain for the IoT: privacy-preserving protection of sensor data. Journal of the Association for Information Systems, 20(9), 1274-1309.
8	Deesign principles for IoT and sensor-based in-home monitoring system for assisting diabetes patients	**About both**	*Initial agreement*	Chatterjee, S., Byun, J., Dutta, K., Pedersen, R. U., Pottathil, A., & Xie, H. (2018). Designing an Internet-of-Things (IoT) and sensor-based in-home monitoring system for assisting diabetes patients: iterative learning from two case studies. European Journal of Information Systems, 27(6), 670-685.
9	Design principles for ethical collaboration	**About user activity**	*Initial agreement*	Chatterjee, S., Sarker, S., & Fuller, M. A. (2009). A deontological approach to designing ethical collaboration. Journal of the Association for Information Systems, 10(3), 138-169.
10	Design principles for virtual worlds	**About both**	*Resolved*	Chaturvedi, A. R., Dolk, D. R., & Drnevich, P. L. (2011). Design principles for virtual worlds. MIS Quarterly, 673-684.
11	Design principles for the assessment of human competences	**About artifact**	*Initial agreement*	Coenen, T., Coertjens, L., Vlerick, P., Lesterhuis, M., Mortier, A. V., Donche, V., ... & De Maeyer, S. (2018). An information system design theory for the comparative judgement of competences. European Journal of Information Systems, 27(2), 248-261.
12	Design principles for carbon management systems	**About both**	*Initial agreement*	Corbett, J. (2013). Designing and using carbon management systems to promote ecologically responsible behaviors. Journal of the Association for Information Systems, 14(7), 339-378.
13	Value-sensitive design principles	**About both**	*Initial agreement*	Dadgar, M., & Joshi, K. D. (2018). The role of information and communication technology in self-management of chronic diseases: an empirical investigation through value sensitive design. Journal of the Association for

				Information Systems, 19(2), 86-112.
14	Design principles for secure collaborative process	**About artifact**	*Initial agreement*	D'Aubeterre, F., Singh, R., & Iyer, L. (2008). A Semantic Approach to Secure Collaborative Inter-Organizational eBusiness Processes (SSCIOBP). Journal of the Association for Information Systems, 9(3/4), 231-266.
15	Design principles for disaster relief supply chain	**About artifact**	*Initial agreement*	Day, J. M., Junglas, I., & Silva, L. (2009). Information flow impediments in disaster relief supply chains. Journal of the Association for Information Systems, 10(8), 637-660.
16	Design principles for virtual business model innovation	**About user activity**	*Initial agreement*	Ebel, P., Bretschneider, U., & Leimeister, J. M. (2016). Leveraging virtual business model innovation: a framework for designing business model development tools. Information Systems Journal, 26(5), 519-550.
17	Design guidelines for DSS	**About both**	*Initial agreement*	Elam, J. J., & Mead, M. (1990). Can software influence creativity?. Information Systems Research, 1(1), 1-22.
18	Design principles for information infrastructures	**About both**	*Resolved*	Eriksson, O., & Ågerfalk, P. J. (2010). Rethinking the Meaning of Identifiers in Information Infrastructures. Journal of the Association for Information Systems, 11(8), 433-454.
19	Design principles for a meta model of a generic could migration process model	**About artifact**	*Resolved*	Fahmideh, M., Daneshgar, F., Rabhi, F., & Beydoun, G. (2019). A generic cloud migration process model. European Journal of Information Systems, 28(3), 233-255.
20	Design principles for preventing IT failures	**About artifact**	*Initial agreement*	Ferioli, C., & Migliarese, P. (1996). Supporting organizational relations through information technology in innovative organizational forms. European Journal of Information Systems, 5(3), 196-207.
21	Design principles for tailorable technology design	**About artifact**	*Initial agreement*	Germonprez, M., Hovorka, D., & Collopy, F. (2007). A theory of tailorable technology design. Journal of the Association for Information Systems, 8(6), 351-367.
22	Design principles for interenterprise systems to foster supply chain flexibiltiy	**About both**	*Initial agreement*	Gosain, S., Malhotra, A., & El Sawy, O. A. (2004). Coordinating for flexibility in e-business supply chains. Journal of Management Information Systems, 21(3), 7-45.
23	Design principles for the design of online selling mechanisms	**About artifact**	*Initial agreement*	Granados, N., Gupta, A., & Kauffman, R. J. (2010). Research commentary—information transparency in business-to-consumer markets: concepts, framework, and research agenda. Information Systems Research, 21(2), 207-226.
24	Design Principles for a "Sweet Spot Change Strategy"	**About user activity**	*Initial agreement*	Gregor, S., Imran, A., & Turner, T. (2014). A 'sweet spot'change strategy for a least developed country: leveraging e-Government in Bangladesh. European Journal of Information Systems, 23(6), 655-671.
25	Information systems use principles	**About both**	*Resolved*	Hales, M. (1991). A human resource approach to information systems development—the ISU (information systems use) design model. Journal of Information Technology, 6(3-4), 140-161.
26	Design principles for dynamic complexity	**About artifact**	*Initial agreement*	Hanseth, O., & Lyytinen, K. (2010). Design theory for dynamic complexity in information infrastructures: the case of building internet. Journal of Information Technology, 25(1), 1-19.
27	Design principles for user involvement in designing mobile and temporarily interconnected systems	**About both**	*Initial agreement*	Henfridsson, O., & Lindgren, R. (2010). User involvement in developing mobile and temporarily interconnected systems. Information Systems Journal, 20(2), 119-135.
28	Design principles for communications for group report authoring	**About user activity**	*Initial agreement*	Heng, M. S., & De Moor, A. (2003). From Habermas's communicative theory to practice on the internet. Information Systems Journal, 13(4), 331-352.
29	Teaching framework for reflective Enterprise Systems practitioners	**About both**	*Resolved*	Hustad, E., & Olsen, D. H. (2014). Educating reflective Enterprise Systems practitioners: a design research study of the iterative building of a teaching framework. Information Systems

				Journal, 24(5), 445-473.
30	Design principles for service network effects (as part of a design theory)	**About both**	*Initial agreement*	Janiesch, C., Rosenkranz, C., & Scholten, U. (2019). An Information Systems Design Theory for Service Network Effects. Journal of the Association for Information Systems: forthcoming.
31	Design principles for dual IS-supported work	**About both**	*Initial agreement*	Käkölä, T. K., & Koota, K. I. (1999). Redesigning computer-supported work processes with dual information systems: the work process benchmarking service. Journal of Management Information Systems, 16(1), 87-119.
32	Design principles for user calibration	**About artifact**	*Initial agreement*	Kasper, G. M. (1996). A theory of decision support system design for user calibration. Information Systems Research, 7(2), 215-232.
33	Design principles for service-oriented systems development	**About artifact**	*Initial agreement*	Keith, M., Demirkan, H., & Goul, M. (2013). Service-oriented methodology for systems development. Journal of Management Information Systems, 30(1), 227-260.
34	Design principles for virtual co-creation	**About both**	*Initial agreement*	Kohler, T., Fueller, J., Matzler, K., Stieger, D., & Füller, J. (2011). Co-creation in virtual worlds: The design of the user experience. MIS Quarterly, 773-788.
35	Design principles for value-based compliance analysis	**About user activity**	*Initial agreement*	Kolkowska, E., Karlsson, F., & Hedström, K. (2017). Towards analysing the rationale of information security non-compliance: Devising a Value-Based Compliance analysis method. The Journal of Strategic Information Systems, 26(1), 39-57.
36	Design theory for cognitively enhanced process model presentation	**About artifact**	*Initial agreement*	Kuechler, B., & Vaishnavi, V. (2008). On theory development in design science research: anatomy of a research project. European Journal of Information Systems, 17(5), 489-504.
37	Design principles for enterprise architecture management	**About both**	*Resolved*	Lange, M., Mendling, J., & Recker, J. (2016). An empirical analysis of the factors and measures of Enterprise Architecture Management success. European Journal of Information Systems, 25(5), 411-431.
38	Design principles for market surveillance systems	**About artifact**	*Resolved*	Li, X., Sun, S. X., Chen, K., Fung, T., & Wang, H. (2015). Design theory for market surveillance systems. Journal of Management Information Systems, 32(2), 278-313.
39	Design principles for competence management systems	**About both**	*Initial agreement*	Lindgren, R., Henfridsson, O., & Schultze, U. (2004). Design principles for competence management systems: a synthesis of an action research study. MIS quarterly, 435-472.
40	Design principles for gamification	**About artifact**	*Resolved*	Liu, D., Santhanam, R., & Webster, J. (2017). Toward Meaningful Engagement: A Framework for Design and Research of Gamified Information Systems. MIS Quarterly, 41(4), 1011-1034.
41	Guidelines for conceptual modeling of user-generated content	**About artifact**	*Initial agreement*	Lukyanenko, R., Wiersma, Y., Huber, B., Parsons, J., Wachinger, G., & Meldt, R. (2017). Representing crowd knowledge: Guidelines for conceptual modeling of user-generated content. Journal of the Association for Information Systems, 18(4), 297.
42	Design theory for systems that support emergent knowledge processes	**About both**	*Initial agreement*	Markus, M. L., Majchrzak, A., & Gasser, L. (2002). A design theory for systems that support emergent knowledge processes. MIS Quarterly, 179-212.
43	Design theory principles for emergent knowledge processes	**About both**	*Initial agreement*	Markus, M. L., Majchrzak, A., Gasser, L. (2002). A design theory for systems that support emergent knowledge processes. MIS quarterly, 179-212.
44	Design principles for requirement mining systems	**About both**	*Resolved*	Meth, H., Mueller, B., & Maedche, A. (2015). Designing a requirement mining system. Journal of the Association for Information Systems, 16(9), 799-837.
45	Design principles for tailored DSS	**About both**	*Initial agreement*	Miah, S. J., Gammack, J. G., & McKay, J. (2019). A Metadesign Theory for Tailorable Decision Support. Journal of the Association for Information Systems, 20(5), 570-603.
46	Design principles for	**About both**	*Initial*	Morana, S., Kroenung, J., Maedche, A., &

	process guiding system		*agreement*	Schacht, S. (2019). Designing process guidance systems. Journal of the Association for Information Systems, 20(5), 499-535.
47	Design theory for creativity support systems	**About both**	*Initial agreement*	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
48	Design Principles for Electronic Feedback Systems	**About both**	*Resolved*	Niehaves, B., & Ortbach, K. (2016). The inner and the outer model in explanatory design theory: the case of designing electronic feedback systems. European Journal of Information Systems, 25(4), 303-316.
49	Design principles for program generators	**About artifact**	*Initial agreement*	Norman, M., & Muriel, A. (1984). Writing simple program generators: a case study in building productivity tools. Journal of Management Information Systems, 1(1), 102-111.
50	Principles for designing class structures	**About artifact**	*Initial agreement*	Parsons, J., & Wand, Y. (2013). Extending classification principles from information modeling to other disciplines. Journal of the Association for Information Systems, 14(5), 245-273.
51	Design principles for socio-technical artefacts to provide performance feedback at scale	**About both**	*Initial agreement*	Piccoli, G., Rodriguez, J., Palese, B., & Bartosiak, M. L. (2019). Feedback at scale: designing for accurate and timely practical digital skills evaluation. European Journal of Information Systems, 1-20.
52	Principles for the design of design of data integration requirements	**About user activity**	*Initial agreement*	Rosenkranz, C., Holten, R., Räkers, M., & Behrmann, W. (2017). Supporting the design of data integration requirements during the development of data warehouses: a communication theory-based approach. European Journal of Information Systems, 26(1), 84-115.
53	Design principles for sensemaking support systems	**About both**	*Initial agreement*	Seidel, S., Chandra Kruse, L., Székely, N., Gau, M., & Stieger, D. (2018). Design principles for sensemaking support systems in environmental sustainability transformations. European Journal of Information Systems, 27(2), 221-247.
54	Design principles for administrative DSS	**About user activity**	*Initial agreement*	Sena, J. A., & Olson, D. H. (1996). Decision support for the administrative man: a prototype DSS case. European Journal of Information Systems, 5(1), 10-23.
55	Design principles for gamified security training system	**About both**	*Initial agreement*	Silic, M., & Lowry, P. B. (2019). Using Design-Science Based Gamification to Improve Organizational Security Training and Compliance. Journal of Management Information Systems, 37(1), 129-161.
56	Design principles for blog communities	**About both**	*Initial agreement*	Silva, L., Goel, L., & Mousavidin, E. (2009). Exploring the dynamics of blog communities: the case of MetaFilter. Information Systems Journal, 19(1), 55-81.
57	Design principles for blog communities	**About both**	*Initial agreement*	Silva, L., Goel, L., & Mousavidin, E. (2009). Exploring the dynamics of blog communities: the case of MetaFilter. Information Systems Journal, 19(1), 55-81.
58	Design theory for IS security policies and guidelines	**About artifact**	*Resolved*	Siponen, M. and Ivari, J. (2006). Six design theories for IS security policies and guidelines. Journal of the Association for Information systems, 7(1), 445-472.
59	Design theory for secure ISD methods	**About artifact**	*Initial agreement*	Siponen, M., Baskerville, R., & Heikka, J. (2006). A design theory for secure information systems design methods. Journal of the Association for Information Systems, 7(1), 725-770.
60	Design principles for mapping routing decisions	**About both**	*Resolved*	Soffer, P., Wand, Y., & Kaner, M. (2015). Conceptualizing routing decisions in business processes:Theoretical analysis and empirical testing. Journal of the Association for Information Systems, 16(5), 345-393.
61	Design principles for social	**About user**	*Resolved*	Spagnoletti, P., Resca, A., & Sæbø, Ø. (2015). Design for social media engagement: insights

	media engagement in elderly care assistance	**activity**		from elderly care assistance. Journal of Strategic Information Systems, 24(2), 128-145.
62	Design principles for Organizational Memory Information Systems	**About both**	*Initial agreement*	Stein, E. W., & Zwass, V. (1995). Actualizing organizational memory with information systems. Information Systems Research, 6(2), 85-117.
63	Design principles for service-oriented E-Government	**About user activity**	*Initial agreement*	Tan, C. W., Benbasat, I., & Cenfetelli, R. T. (2013). IT-mediated customer service content and delivery in electronic governments: An empirical investigation of the antecedents of service quality. MIS Quarterly, 77-109.
64	Design theory for for population targeted requirements acquisition	**About user activity**	*Resolved*	Tuunanen, T., & Peffers, K. (2018). Population targeted requirements acquisition. European Journal of Information Systems, 27(6), 686-711.
65	Design requirements and design for Scientifically Controlled Screening Systems	**About artifact**	*Initial agreement*	Twyman, N. W., Lowry, P. B., Burgoon, J. K., & Nunamaker Jr, J. F. (2014). Autonomous scientifically controlled screening systems for detecting information purposely concealed by individuals. Journal of Management Information Systems, 31(3), 106-137.
66	Design theory for auto-ID enabled shopping assistance artifacts	**About artifact**	*Initial agreement*	Venkatesh, V., Aloysius, J. A., Hoehle, H., & Burton, S. (2017). Design and evaluation of auto-id enabled shopping assistance artifacts in customers' mobile phones: two retail store laboratory experiments. MIS Quarterly, 41(1), 83-113.
67	Design principles for vigilant information systems	**About artifact**	*Initial agreement*	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.
68	Design principles for artificial immune systems	**About artifact**	*Initial agreement*	Wong, N., Ray, P., Stephens, G., & Lewis, L. (2012). Artificial immune systems for the detection of credit card fraud: an architecture, prototype and preliminary results. Information Systems Journal, 22(1), 53-76.
69	Design principles of integrated information platform for emergency response	**About artifact**	*Initial agreement*	Yang, L., Su, G., & Yuan, H. (2012). Design principles of integrated information platform for emergency responses: the case of 2008 Beijing Olympic Games. Information Systems Research, 23, 761-786.