

2022

## How Designers Use Design Principles: Design Behaviors and Application Modes

Leona Chandra Kruse

*Universitat Liechtenstein, leona.chandra@uni.li*

Sandeep Purao

*Bentley University, spurao@bentley.edu*

Stefan Seidel

*University of Liechtenstein, stefan.seidel@uni.li*

Follow this and additional works at: <https://aisel.aisnet.org/jais>

---

### Recommended Citation

Chandra Kruse, Leona; Purao, Sandeep; and Seidel, Stefan (2022) "How Designers Use Design Principles: Design Behaviors and Application Modes," *Journal of the Association for Information Systems*, 23(5), 1235-1270.

DOI: 10.17705/1jais.00759

Available at: <https://aisel.aisnet.org/jais/vol23/iss5/3>

This material is brought to you by the AIS Journals at AIS Electronic Library (AISeL). It has been accepted for inclusion in *Journal of the Association for Information Systems* by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

## How Designers Use Design Principles: Design Behaviors and Application Modes

Leona Chandra Kruse,<sup>1</sup> Sandeep Purao,<sup>2</sup> Stefan Seidel<sup>3</sup>

<sup>1</sup>University of Liechtenstein, Liechtenstein, [leona.chandra@uni.li](mailto:leona.chandra@uni.li)

<sup>2</sup>Bentley University, USA, [spurao@bentley.edu](mailto:spurao@bentley.edu)

<sup>3</sup>University of Liechtenstein, Liechtenstein, [stefan.seidel@uni.li](mailto:stefan.seidel@uni.li)

### Abstract

This paper investigates how information systems design professionals use design principles (extracted from a prior design science research project) in a new design situation. We do this by capturing think-aloud protocols from experienced design professionals who are given access to potentially useful design principles. Our analysis identifies two dimensions of use: design behaviors (what designers do) and application modes (how designers apply the principles). Mapping across the dimensions suggests two use pathways: forward chaining and backward chaining. Our study shows how empirically studying expert designers can shed light on the microprocesses of design principles in use, and how an empirical turn in the investigation can contribute to clarifying the fundamental nature of design principles. We conclude by highlighting the implications of these insights for crafting more useful design principles.

**Keywords:** Design Principles, Knowledge Reuse, Think-Aloud Method, Design Science Research

Dorothy E. Leidner was the accepting senior editor. This research article was submitted on April 11, 2020 and underwent three revisions.

## 1 Introduction

Designing information systems is a complex and knowledge-intensive enterprise. It requires designers to be creative in the moment while at the same time drawing on a set of different knowledge. The knowledge includes personal experience (Schön, 1983), knowledge of technologies (Iivari et al., 2004), heuristics (Parsons & Saunders, 2004), rules (Hanseth & Lyytinen, 2010), and patterns (Gamma, 1995). The design science research (DSR) community suggests adding design principle to this set, defining it as “knowledge about creating other instances of artifacts that belong to the same class” (Sein et al., 2011, p. 39).

Design principles are extracted from DSR efforts (Baskerville & Pries-Heje, 2010; Sein et al., 2011), stated as: “in condition C, to achieve outcome A, do action B” (Romme & Endenburg, 2006, p. 288), and justified by appealing to established theory<sup>1</sup> (Gregor & Jones, 2007; Walls et al., 1992). Examples include design principles for designing sensemaking support systems (Seidel et al., 2018) and for designing virtual worlds (Chaturvedi et al., 2011).

Design principles have a dual audience. They contribute to a cumulative body of knowledge (Wieringa, 2009) about the design of different classes of IT artifacts (Baskerville & Pries-Heje, 2010;

<sup>1</sup> In so doing, they do not only capture best practices as what design patterns do in software engineering (e.g., Alexander, 1979; Gamma, 1995).

Fischer et al., 2010). They are also intended to provide design professionals with actionable knowledge useful in building new versions of similar artifacts. In spite of this acknowledgment, we know little about how design professionals use the design principles from prior DSR efforts in their design practice. One may assume the principles to be straightforwardly reused by design professionals. However, Markus (2001) and others point out that reuse is not a natural adjunct to creative problem-solving. Moreover, the “not-invented-here” (NIH) syndrome continues to hamper the reuse of prior knowledge in new situations (Favaro, 1991; Katz & Allen, 1982; Rech et al., 2007). Therefore, it may be naive to expect a simple reuse. These observations motivate our research question: *How do design professionals use design principles in new design situations?*

In this investigation, we create a design situation in which a design professional has access to potentially useful design principles. Our data-gathering relies on the think-aloud method (van Someren et al., 1994), which allows us to gain direct access to the “doings and thinkings” of design professionals as they occur. The analysis uses vocabularies from multiple research streams, including design behaviors (e.g., Atman & Bursic, 1998; Cardella, Atman, & Adams, 2006; Newell & Simon, 1972; Purao et al., 2002; Ullman, 2010), design knowledge reuse (e.g., Rosson & Carroll, 1996; Sen, 1997), and knowledge sharing (e.g., Salovaara & Tuunainen, 2015; 2013).

We clarify the scope of our research by highlighting what this study *is not* and by stating what it *is*. It does not test assumptions about concerns such as the nature of the indeterminacy of design principles (Lukyanenko & Parsons, 2020). Neither does it explore different formulations of design principles (Gregor et al., 2020). Further, it does not purport to develop a philosophical treatise or an authoritative guide for the construction of design principles. Instead, our study focuses on generating empirical evidence to complement the ongoing discourse on the reuse of design principles (e.g., Iivari et al., 2020).

By doing so, we hope to offer five contributions. First, we unpack the use of design principles along two dimensions: what designers *do* (design behaviors) and how they *apply* the design principles

(application modes). Second, we identify pathways for design principles use at the intersection of these two dimensions. Third, we derive initial lessons for articulating more useful design principles. Fourth, we demonstrate how the think-aloud method can provide insights into designers’ micropractices and cognitive activities.<sup>2</sup> Finally, we hope this effort can bridge the two research communities (design studies and design science research) that have, by and large, remained separate until now.<sup>3</sup> Our work will contribute to the nascent stream of work in DSR that has only recently begun to explore the application of design principles in practice (Lukyanenko & Parsons, 2020; Tuunainen & Peffers, 2018).

We proceed as follows. The following section summarizes prior research in multiple streams to highlight various perspectives of the use of design principles. We then describe the research approach, the research design, and data collection and analysis procedures. Next, we detail the findings across the two dimensions of design behaviors and application modes for design principles and map the two. Finally, we discuss the study’s contributions and implications and conclude with suggestions for future work.

## 2 Prior Work

We identify several perspectives from prior work to inform our investigation of the phenomenon—the *use of design principles by design professionals*. The first concerns the nature of knowledge contained in design principles (Sein et al., 2011). The second perspective conceptualizes the use of design principles as part of a move from method-as-espoused to method-in-use (Argyris & Schön, 1974). The third perspective views the use of design principles as an effort of design professionals to access the expertise they lack (Cross, 2004). The fourth perspective frames the application of design principles as a special case of knowledge reuse, similar to code reuse (Krueger, 1992) or pattern reuse (Purao et al., 2002). The fifth perspective views the use of design principles in a manner similar to studies of design behaviors (e.g., Cross et al., 1994; Gero & McNeill, 1998). We present our review in three clusters: (1) design principles as (a form of) design knowledge, (2) studies of design behaviors and approaches, and (3) designing with design knowledge.

<sup>2</sup> Todd and Benbasat (1987, p. 501) advocate the use of verbal protocols in decision support system research. They consider the method superior to interviewing and argue that verbal protocols are worth the labor-intensive coding activities, which “are estimated to take several hours for each minute of verbalized information.”

<sup>3</sup> The design studies community has engaged in empirical studies of design behaviors and processes (e.g., Cross, 2001a; 2001b; Dorst & Cross, 2001), and the design science research community has focused on the formulation (Gregor & Jones, 2007; Gregor et al., 2020) and derivation (Hevner & Chatterjee, 2010; Sein et al., 2011) of prescriptive knowledge as design principles.

## 2.1 Design Principles as Design Knowledge

The very idea of design knowledge can be problematic because design remains difficult to decontextualize. A design “never ... begins from scratch” (Latour, 2008, p. 5) and addresses a wicked problem (Rittel & Webber, 1973). These characteristics make the creation and use of codified design knowledge a challenge. Even so, both scholars and practitioners acknowledge the importance of codified design knowledge (Garud, 1997) and have proposed several forms of design knowledge such as design patterns (e.g., Borchers, 2001; Denning & Dargan, 1996), technological rules (e.g., van Aken, 2001), and analysis patterns (Fowler, 1996). They represent the outcomes of moving from tacit to explicit knowledge (Nonaka & Konno, 1998; Nonaka & Toyama, 2003), drawing on best practices in software engineering (e.g., Gamma, 1995). Design principles follow a similar path but draw on the outcomes of DSR efforts (instead of industry expertise), are backed by prior theory (instead of best practices), and contain knowledge “about creating other instances of the same class” (Sein et al., 2011, p. 39). Inherent in the formulation of design principles are tensions between the nomothetic and the idiosyncratic approaches (Baskerville et al., 2015). These tensions point to the need to explore design approaches and behaviors during the use of design principles.

## 2.2 Design Approaches and Behaviors

Scholarly understanding of design has evolved past its initial characterizations as something comparable to decision-making and problem-solving (Simon, 1996), with an emphasis on objectivity and rationality (Cross, 2001a, p. 1). We now know that design problems are wicked (Rittel & Weber, 1973) and require iterative and nonrational approaches (Mathiassen, 1998; Mathiassen & Purao, 2002) including reflective conversations with the design materials and situations (Dorst, 2006; Schön, 1983, 1987). New conceptualizations of design extend Simon’s (1996) bounded rationality, highlighting how one cannot reduce designing to problem-solving (Hatchuel & Weil, 2002). Others describe the outcome of design as “the resolution of paradoxes between discourses” (Dorst, 2006, p. 17). These varied conceptualizations share one essential trait (Le Masson et al., 2013, p. 4): “recognition of the unknown, [investigation] ... based on available knowledge, and generation of new concepts.” Table 1 summarizes these perspectives and derives implications to understand the designers’ activities and behaviors.

Designing, thus, represents a nonroutine effort in a new domain,<sup>4</sup> confronting the unfamiliar and drawing on prior knowledge to conceive a solution in order to achieve some envisioned outcome or goal. Prior work in design studies unpacks the work of “designing” into specific design behaviors, i.e., activities from designers, as the outcomes of their cognitive moves. Examples include the function, behavior, and structure model (Gero, 1990; Gero & Kannengiesser, 2014; Gero & McNeill, 1998) and behavior clusters of concept generation, refinement and evaluation, note-taking, and retrieving prior knowledge (see Ullman, 2010). Table 2 summarizes this stream of work. Another stream of work examines designing with design knowledge, which we examine next.

## 2.3 Designing with Design Knowledge

The impetus for designing with design knowledge comes from the idea of not reinventing the same artifacts (McIlroy, 1968; Rech et al., 2007). Designers can incorporate well-tested components to design larger systems (Rech et al., 2007). However, much research shows how reuse can be challenging because of individual barriers related to self-interest and self-image (Judicibus, 1996; Katz & Allen, 1982; Rech et al., 2007), organizational barriers related to a short-term focus and policy (Favaro, 1991; Judicibus, 1996; Rech et al., 2007), and problems related to the producer-consumer knowledge gap (Markus, 2001) as well as knowledge ephemerality (Salovaara & Tuunainen, 2015, 2013). Table 3 summarizes the barriers.

Few studies have empirically investigated design knowledge reuse. Exceptions include Rosson and Carroll (1996), who investigate the reuse of interface classes in Smalltalk programming. According to them, reuse depends on how designers find, debug, and evaluate the reuse contexts. Another example is Sen (1997), who investigates the reuse of diagrams for database design and describes reuse in terms of retrieval, selecting the best artifact, adaptation, and evaluation. Purao et al. (2002) conceptualize reuse as retrieval, adaptation, and integration. Based on these studies, reuse does not occur as a single, atomic act. Instead, it requires multiple cognitive actions. These cognitive actions remain fraught with biases, such as anchoring (Parsons & Saunders, 2004) and fixation (Jansson & Smith, 1991). As design professionals engage in these actions, they convert prior design knowledge into design decisions (Mathiassen & Purao, 2002; Vitalari & Dickson, 1983).

---

<sup>4</sup> e.g., designing a platform for supporting cancer care decisions; the design situation we consider in this research

**Table 1. Contemporary Approaches to Conceptualizing Design**

Design...	Designers' behaviors	Source
As a rational activity	Identify a problem and determine the course of action to arrive at a solution.	Dorst, 2006; Purao et al., 2002; Simon, 1996
As a reflective practice	Reflect on their actions/situations, use their knowledge, and follow their artistry.	Dorst, 2006; Mathiassen, 1998; Schön, 1983
As recognition of the unknown	Realize that something is required that does not yet exist.	Le Masson et al., 2013
As drawing on the known	Use their knowledge and experience in various domains.	Iivari, 2016; Iivari et al., 2004; Le Masson et al., 2013; Vitalari, 1985

**Table 2. Clusters of Design Behaviors**

Cluster	Design behaviors (with sources)
Understand problem and scope	Establish the need/problem to be solved (Ullman, 2010); develop requirements (Atman & Bursic, 1998; Cardella et al., 2006; Newell & Simon, 1972; Ullman, 2010); gather information (Atman & Bursic, 1998; Cardella et al., 2006); plan to solve the problem (Ullman, 2010); represent constraint (Purao et al., 2002)
Retrieve prior knowledge	Retrieve domain knowledge, technique knowledge, experience base (Purao et al., 2002); uncover existing solutions for similar problems (Newell & Simon, 1972)
Look for alternatives	Generate and evaluate alternative solutions by comparing them to the design requirements and to each other (Atman & Bursic, 1998; Cardella et al., 2006; Newell & Simon, 1972; Ullman, 2010); conduct feasibility analyses (Atman & Bursic, 1998; Cardella et al., 2006)
Generate new concepts	Refine, extend, or recombine previous design concepts into something new (Hatchuel & Weil, 2002; Le Masson et al., 2013; Purao et al., 2002)
Propose solutions	Determine acceptable solutions (Atman & Bursic, 1998; Cardella et al., 2006; Newell & Simon, 1972; Ullman, 2010); form, expand, and simulate concepts (Purao et al., 2002)
Implement and communicate	Implement solution (Atman & Bursic, 1998; Cardella et al., 2006; Newell & Simon, 1972; Ullman, 2010); communicate results (Atman & Bursic, 1998; Cardella et al., 2006; Newell & Simon, 1972; Ullman, 2010); refocus on parts of the design (Purao et al., 2002); validate design (Purao et al., 2002; Ullman, 2010)

**Table 3. Barriers to Knowledge Reuse in Software Design and Development**

Barrier	Description
Artist's syndrome	Designers want to build something "beautiful" and avoid the reuse of external and "ugly" software, even at the cost of not fulfilling the requirements (Rech et al., 2007).
Feudal lord's syndrome	Managers tend to judge their importance by the size of their project teams and budgets. Reuse would lead to smaller teams and cheaper projects, and building reusable components would tend to benefit other departments most (Rech et al., 2007).
Not invented here (NIH) syndrome	Companies or departments often see others' products as inferior to what they themselves have or could create. The motivation to reuse them is negative to non-existent (Katz & Allen, 1982).
Manic-depressive syndrome	New approaches and technologies like software reuse are often introduced with high expectations that lead to an initial euphoria followed by disillusion (Favaro, 1991).

### 3 Research Approach

The investigation requires an appropriate research method to overcome recall problems (because of time elapsed between the activity of interest and data collection) and surface the fleeting and often complex reasoning associated with design decisions. Methods relying on self-report, observations, and interviews are, therefore, not suitable.

#### 3.1 Research Method: The Think-aloud Method

The think-aloud method allows such an immediate and direct access to the design professionals' "doings and thinkings" as they occur (van Someren et al., 1994). The method is appropriate for our investigation because (1) it supports us in understanding designers' cognitive moves when using design principles; (2) it generates insights into cognitive processes beyond what may be gleaned from interviews or observation (Ericsson & Simon, 1998); and (3) it is compatible with open coding, allowing us to draw on vocabularies from multiple research streams during analysis (e.g., Cramer-Petersen et al. 2019).

The method emphasizes "externalizing covert thinking without altering it" (Ericsson & Simon, 1998, p. 180). The closest connection between thinking and verbal reporting is achieved when the individual thinks aloud as a sequence of thoughts while completing the task (Ericsson & Crutcher, 1991; Ericsson & Simon, 1993; 1998), not when the individual describes how they performed the task at a later time. During a think-aloud session, participants are asked to verbalize their thoughts while performing a task. The researcher records the sessions, transcribes the recordings (called verbal protocol), and analyzes the transcripts. The verbal protocol captures in-the-moment deliberations too ephemeral (Salovaara & Tuunainen, 2015; 2013) to be revealed through self-reporting or interviews (Ericsson & Simon, 1993).

Applications of the method in IS research and adjacent disciplines<sup>5</sup> (e.g., Khatri & Vessey, 2016; Price & Shanks, 2011; Purao et al., 2002; Wang, 1996) emphasize the importance of choosing "the right" participants. Scholars who apply the method are less concerned with the number of participants (between two and 13 in prior studies. Because of "the high density of data that will be found in a single verbalization, samples are usually very small, commonly between 2 and 20" (Todd & Benbasat, 1987, p. 501). While the small number of participants may hinder generalizability (Tsang & Williams, 2012), the think-aloud method is one of the best methods for gaining access to individual cognitive activities (Todd & Benbasat, 1987).

<sup>5</sup> It has been used to analyze design behaviors (Cash et al., 2015; Suwa et al., 1998; Suwa & Tversky, 1997), usability (Cross, 2001a, 2001b), creative thinking (Pringle &

#### 3.2 Research Setting: Design Principles and Design Situation

We asked design professionals to perform a design task using a set of design principles from a completed DSR project. We chose the design situation and the design principles from a domain rather unfamiliar to our target participants: constructing a clinical decision support system (CDSS) to develop the course of treatment for breast cancer patients (Gaudioso et al., 2016). The design principles were peer-reviewed and published at a workshop affiliated with the premier conference in the IS discipline (see Table 4).

Some design principles use domain-specific terminology (e.g., Principle 9, *clinical practice guidelines*), some remain open to interpretation (e.g., Principle 5, *meaningful participation*), some use terminology from software design (e.g., Principle 4, *presentation modalities*), and some use scientific language (e.g., Principle 7, *evidence-based knowledge*). Together, they provide design professionals with guidance but also pose challenges. A design situation (brief description) provided the initial impetus for the designers (see Figure 1).

The design situation was hypothetical but grounded in practice (Gaudioso et al., 2016). Such meetings (multidisciplinary care conferences) are a weekly occurrence in community care hospitals, where a team of specialists (oncologists, radiologists, anesthesiologists, nurses, and others) come together to decide the course of treatment for multiple cancer patients.

#### 3.3 Participants

The criteria for recruitment were deep competencies in software design (Iivari, 2016; Vitalari, 1985), prior experience in using codified knowledge (e.g., code snippets and design patterns), and no experience in the design domain (CDSS for cancer care). With these criteria, we recruited seven participants, all with significant (14 to 16 years) experience. Each held a position requiring deep technical knowledge and nontrivial managerial responsibility. All had worked with design knowledge (patterns) in a corporate library and were aware of other knowledge repositories in their respective organizations (see Table 5). These participants were difficult to recruit because of the time demands of design sessions and the need to ensure a trusting relationship. Trust is a prerequisite for the participants to share their spontaneous, unedited thoughts and associations.

Snowden, 2017), collaboration (Sio et al., 2018), reasoning (Cramer-Petersen et al., 2019), and ideation (Laing et al., 2017).

**Table 4. Design Principles from a Completed DSR Project (Gaudioso et al., 2016)**

<b>Design principle</b>	<b>Specific action</b>
<b>Design Principle 1:</b> Design the artifact as an integral part of the organizational and technological fabric and work practices.	Provide integration points before and after the deliberation and decision window.
<b>Design Principle 2:</b> Embed researcher in the setting, create high-fidelity prototypes.	Evolve problem and solution definitions through rich interactions.
<b>Design Principle 3:</b> Make available and visible the knowledge required for the case.	Pre-populate knowledge that is relevant to the case instead of relying on search-on-demand.
<b>Design Principle 4:</b> Allow multiple presentation modalities with an integrated summary and user-directed access to details.	Allow user-directed navigation for the group as well as individuals.
<b>Design Principle 5:</b> Facilitate, support, and ensure all experts' meaningful participation in the case discussion.	Allow and record issue-based, non-anonymous discussions.
<b>Design Principle 6:</b> Allow directed access to group memory and similar cases based on issues.	Overcome the absence of some medical experts.
<b>Design Principle 7:</b> Facilitate and support the integration of expert and evidence-based knowledge.	Allow convergent and divergent discussions.
<b>Design Principle 8:</b> Preserve group memory.	Record deliberations and decisions for each case.
<b>Design Principle 9:</b> Plan for and ensure accurate and current content.	Consider patient data as well as clinical practice guidelines and relevant research.

Marco is a lead software designer. His company has just acquired a contract to develop a piece of software that will support doctors who are involved in making decisions about patient treatment at the local Cancer Care Center (CCC). The patient treatment decision is difficult because it requires a combination of perspectives and expertise from a number of specialists. These specialists are only able to meet about once every month because they travel from different locations. At CCC, their meetings tend to run for about two hours, and at each meeting, they end up considering as many as 15 patients. The meetings are attended by about 10-12 specialists in different fields who discuss each case and deliberate to finalize the treatment for each patient before moving on to the next patient. These decisions can be very critical because they have consequences that may be irreversible.

**Figure 1. A Design Situation****Table 5. Participant Profiles**

<b>Name</b>	<b>Expertise</b>	<b>Experience</b>	<b>Leadership responsibilities</b>
P	Software development, innovation, business analysis, software integration	16 years	Innovation lead at a large financial institution
Q	Software development, digital entrepreneurship, IT project management	14 years	Co-founder of a large IT service provider
R	IT project management, business analysis	15 years	Program lead at a large financial institution
S	Leading IT support for a large corporation	15 years	Head of IT support at a multinational financial institution
T	Software development, software integration, systems architecture	16 years	Head of IT support at a public service institution
U	IT architecture, business deployment, line management, software integration	15 years	Head of IT security for a multinational financial institution
V	Software development, software integration, testing	16 years	Testing and integration lead at a large IT service provider

### 3.4 Pilot Session, Design Sessions, and Collecting the Verbal Protocols

Before collecting the data, we conducted two pilot studies. The first pilot study aimed to improve the clarity of the design situation and instructions with the help of two design science researchers. The second pilot study simulated the data-collection procedure with an expert designer. Insights from the second pilot study helped us further improve the clarity of the documents, and the study procedure. The verbal protocol from the pilot was not included in the analysis.

The actual study involved one-on-one design sessions with each participant. Each session started with a practice task for the participant (solve a mathematical problem while verbalizing their thoughts) following prior work (van Someren et al., 1994). The session coordinator (the first author) provided feedback (e.g., encouraging verbalization instead of explaining actions). After the practice task, the participants received all documents (the design situation and the design principles, translated from the original English to German) along with a sketchbook and writing instruments. The session coordinator read the design session script (Appendix C), asked the participant to read the design situation and design principles aloud, and reminded the participant to verbalize their thoughts. She initiated the audio recording and receded into the background. The recordings were transcribed “true verbatim” (including pauses, stutters, murmurs, and other unintelligible or incomplete words) to generate verbal protocols of each session (van Someren et al., 1994). Table 6 summarizes all sessions.

### 3.5 Coding and Analysis

The research process moved forward with protocol segmentation: dividing the transcribed protocols into segments such as multiple sentences, a single sentence, and single phrases, each representing an intelligible idea. As Todd and Benbasat (1987) describe: “coding and analysis of a protocol ... [are] ... labor-intensive ... [often requiring] ... several hours for each minute of verbalized information.” The segmented protocols were coded and analyzed. Figure 2 summarizes (with additional details in Appendix D).

The coding was guided by the data. During open coding (Strauss & Corbin, 1998), we first examined two (out of seven) verbal protocols (independently by two coders, followed by a discussion to reach consensus). We identified initial codes along two dimensions: (1) what designers do (design behaviors),

and (2) how they apply the design principles (application modes). With this set of codes as the starting point, the first author coded the remaining protocols. As new codes emerged, she refined the initial codes, and finally, returned to recode the first two protocols. All of us discussed and refined the codes before the next step, which involved developing code clusters. We returned to the protocols to ensure that each segment was coded with (one or more) design behaviors, application modes (if any), and the design principles referenced (if any). Table 7 shows some examples of coding. A detailed description appears in Appendices D1 and D2.

The examples in Table 7 illustrate how each segment exhibits (one or more) design behaviors but may not involve the application of design principles. On the other hand, some segments indicate multiple application modes with one or multiple principles.<sup>6</sup> In the final stage, we generated tabular and visual representations to explore patterns within and across the two dimensions. Then we revisited the vocabularies from different research streams to develop further interpretations.

### 3.6 Ensuring Robustness of Findings

We took several measures to ensure the robustness (credibility, emergence, and generalizability) of our analysis and findings. The first was to provide a clear chain of evidence from data collection to data analysis to articulation of findings. We distributed the study protocol, documents, transcripts, and analytical documents among all authors. With the materials in hand, we could revisit all relevant data points at any stage. We also enhanced credibility through the participation of all co-authors in the analysis, meeting regularly over several months to discuss codes, code clusters, and interpretations.

The second measure to ensure the robustness of our analysis was to avoid force-fitting to prior concepts, a known threat to qualitative exploratory inquiry (Glaser, 1978), by starting our analysis without a specific coding scheme. The segmentation of protocols (Appendix D1) and the identification of codes from data were driven primarily by the research question concerning how designers apply the design principles made available to them. Acknowledging our familiarity with prior research in this stream, we strove to minimize potential bias (Seidel & Urquhart, 2013) by focusing on the data during our analyses (Appendix D2). As the analysis progressed, we consulted the literature to substantiate our emergent concepts. We returned to the data to ensure that our findings remained firmly grounded in the data (Saldaña, 2015).

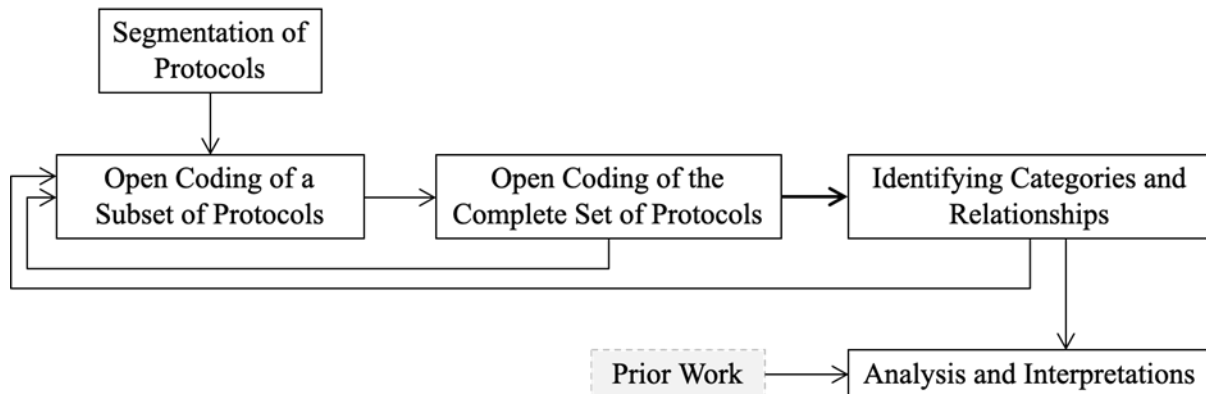
---

<sup>6</sup> This faithful representation, however, complicated the calculation of frequencies and fractions.



**Table 6. Design Sessions and Transcribed Protocols**

Participant	Session length (minutes)	Transcribed protocol (words)
P	35	3234
Q	31	2219
R	29	3261
S	23	2625
T	86	8302
U	25	2748
V	24	3585
	Range [24-35, outlier 86]; Average: 36	Range [2219-3585, outlier 8302]; Average: 3710.57

**Figure 2. Coding and Analysis of Verbal Protocols****Table 7. Coding of Protocol Segments: Examples**

Segment from think-aloud protocol	Design behavior*	Application mode**	Design principles considered***
<i>One needs only a good search function for similar cases ... it happens online... thus, in these ten minutes ... it can hardly take place before that ... except if the assistant ... who does the preparation, who does it here, who prepares each case ... then says ... there are already similar cases in the past. (Participant P, Segment P14)</i>	(1) The designer is making assumptions.	(1) The designer is using a principle directly.	DP 6
<i>This is principle 1, for example, so really that you have a work process. That's really it, that's really the process you have, to, to, um to come to a conclusion. It really has to be um in there. (Participant U, Segment U5)</i>	(1) The designer is drawing inferences. (2) The designer is articulating workflow.	(1) The designer is using a design principle directly.	DP 1
<i>Then you, um, welcome this feedback [laughs], I'll say, and, um [short pause], and then, um, is now, yes, there are many different approaches, if you are going to create a catalog with questions where the participants can, um, respond accordingly, or which method you choose. But I'll say generally it is extremely important that those, those who use this, um, software later, that those have the opportunity to be involved. (Participant R, Segment R18)</i>	(1) The designer is recognizing stakeholders.	None	None

<p><i>I mean, usually the techno, um, the hospital systems are cut off from the outside world, somehow he has to look from outside, from his ... look at his cases, based on his role, and he has to be able to transmit the information, even if, if he discusses it again, it has to be there, because there is principle 3, and prepare knowledge relevant to the case. You have to prepare; otherwise you can't say, in 8 minutes, right. (Participant T, Segment T67)</i></p>	<p>(1) The designer is verifying a design decision.</p>	<p>(1) The designer is combining multiple principles. (2) The designer is contextualizing a principle.</p>	<p>DP 3, DP 4</p>
<p>Note: *one or more design behaviors, **zero or more application modes, ***design principles</p>			

Finally, we moved conceptually from the specific context we studied (application of design principles for CDSS by expert designers who are not familiar with that context) to the broader context (application of design principles by designers in a new design situation). We explored how emergent concepts from our analysis mapped against established concepts in IS and design studies. This mapping helped us position our findings in multiple streams of research (Urquhart et al., 2010) and move toward abstract concepts.

## 4 Findings

We elaborate our findings along the two dimensions of *design behaviors* and *application modes*. The first dimension, *design behaviors*, refers to what designers do. They are the focus of considerable work in design studies in IS and elsewhere (e.g., Cross, 2001b; Gero & Kannengiesser, 2014). Our findings add to this stream of research by using a specific context: the presence of design principles. The second dimension, *application modes*, refers to how designers apply design principles. This dimension has received much less attention with a few exceptions focusing on artifact reuse (Rosson & Carroll, 1996; Sen, 1997). Our findings identify new ways in which design professionals apply design principles. Figure 3 illustrates this structure of our findings.

### 4.1 Dimension: Design Behaviors

This set of findings responds to the following subquestion derived from our overall research question: *What design behaviors do design professionals exhibit when they are in possession of potentially useful design principles?* We found ten design behaviors across four categories: (1) *determining the scope and challenges*, which describes what designers do to determine boundaries and concerns; (2) *guesstimating missing information*, which describes what they do to explore the design situation; (3) *projecting into the solution space*, which describes what they do to conceptualize design solutions; and (4) *implanting into design processes*, which describes how they structure their work. Table 8 portrays these design behaviors.

The following quotations illustrate the four categories of design behaviors we identified.

#### Category 1: Determining the scope and challenges

*That they get through each case in these 8 minutes ... if they don't have to, um, follow a protocol ... then these are still extremely short times.... (Participant T; behavior: identifying challenges and risks)*

*The central object besides a list ... of course some kinds of patient files that ... and the notes, ... so that you can also retrieve them again. (Participant Q; behavior: establishing design scope)*

#### Category 2: Guesstimating missing information

*Okay, that is, no matter where they come from, the meetings last two hours each, therapies 15, so time constraint ... they can only do two hours. (Participant R; behavior: drawing inferences)*

*I mean, if 20 people come again and again and look at it, then of course not everyone has to see the name. They are only interested in the case, right. (Participant T; behavior: making assumptions)*

*Yes, well, yes, are there any technological ... possibilities ... so that the, the rather shy people are also heard and can make their contribution? (Participant V; behavior: posing rhetorical questions)*

#### Category 3: Project into the solution space

*You would also have a GUI ... Then you have some software on it that represents the whole logic... access to the GUI via user ID and password. (Participant S; behavior: describing features)*

*An assistant will have to prepare everything; it can't be done automatically, ... so ... this person goes through all cases and the presentation. (Participant P; behavior: articulating workflow)*

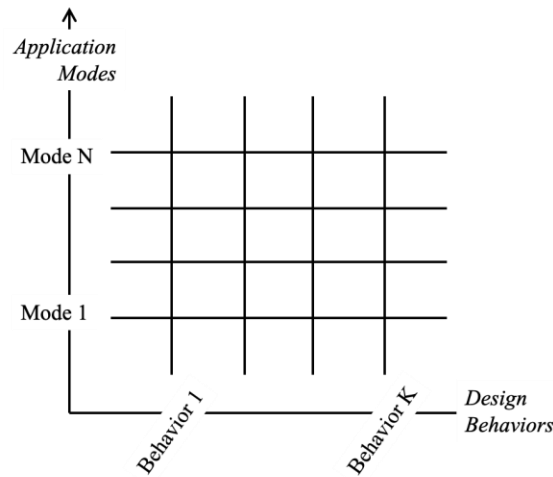


Figure 3. The Structure of Findings

Table 8. Design Behaviors Identified

Design behavior	Description
<b>Category 1: Determining the scope and challenges</b>	
Identifying challenges and risks	Designer frames the situation to identify potential challenges and risks.
Establishing design scope	Designer identifies design needs and establishes the scope of the design effort.
<b>Category 2: Guesstimating missing information</b>	
Drawing inferences	Designer finds new information in light of related information
Making assumptions	Designer takes for granted that something is true.
Posing rhetorical questions	Designer fills a gap by posing questions but leaving them unanswered.
<b>Category 3: Projecting into the solution space</b>	
Describing features	Designer describes some features of the system.
Articulating the workflow	Designer describes how users will complete tasks using the system.
Verifying the solution	Designer assesses the feasibility of the proposed solution.
<b>Category 4: Implanting into the design process</b>	
Recognizing stakeholders	Designer recognizes how to engage with stakeholders in the design process.
Outlining task sequences	Designer outlines the process of designing artifacts.

*For sure they need possibilities to upload each of the cases there .... It's very rare that one gets access to the database of another hospital. (Participant P; behavior: verifying solutions)*

**Category 4: Implant into the design process**

*Um, of course you can make high-level plans, but how that works out in detail, um, you'd have to get that information from the stakeholders. (Participant U; behavior: recognizing stakeholders)*

*You analyze the data that exists. And ... in the ideal case you now have the application available with the correct contents and then the next step is the maintenance or ... how*

*it is now used by participants in future meetings. (Participant R; behavior: outlining task sequence)*

The frequency of the behavior categories varied significantly across the design professionals. For example, more than half (54%) of Participant R's behaviors belonged to Category 4 (*implanting into design process*), while Participant S exhibited no behavior in this category. The opposite was the case for Category 2 (*guesstimating missing information*). About 60% of Participant S's behaviors fell into this category compared to only 3% for Participant R. These variations may be attributed to style differences or individual idiosyncrasies inherent in design (Akin, 2001; Visser, 2009).

Two important observations could be made from these findings. First, the design behaviors did not fundamentally change (see Table 2) as a result of the availability of design principles. Second, even with design principles at hand, the design professionals still exhibited differences in design behaviors, indicating how design professionals did not simply follow available design principles to construct new designs.

Participants referenced the design principles as they engaged in the design effort. In 82% of the observed behaviors (220 out of 268 aggregated behaviors) the participants referred to design principles. In the aggregate, behaviors in Category 2 (*guesstimating missing information*) were most frequent (102 out of 220 segments, ~46%), followed by behaviors in Category 3 (*projecting into solution space*, 68 out of 220 segments, ~31%), then Category 1 (*determining scope and challenges*, 32 out of 220 segments, ~15%), and finally Category 4 (*implanting into design process*, 18 out of 220 segments, ~8%). We examined this finding further by identifying the occurrence of design behaviors in conjunction with each design principle. Figure 4 shows the frequency of design behaviors in each category (aggregated across design professionals) for each design principle.

The figure shows a significant variation in the number of times each design principle was referenced as part of the design behaviors (from 11 to 43, average ~24). The most frequently referenced was Design Principle 4 (43 out of 220 segments, ~20%), and the least frequently referenced was Design Principle 9 (11 out of 220 segments, 5%). Across all principles, behaviors from Category 2 (*guesstimating missing information*) were either the most or second-most frequent category. For example, behaviors from this category represented 25 of 43 segments for Design Principle 4, and 3 out of 11 segments for Design Principle 9. In contrast, behaviors from Category 4 (*implanting into design process*) were often the least frequent across all principles. Two design principles (8 and 9) were, in fact, not referenced in this manner. Only Design Principle 2 was referenced five times in this manner with some other design principles referenced less than five times.

Seeing this data from a different perspective, we note how the participants focused on different design principles as part of their behaviors in each category. For example, behaviors in Category 1 (*determining scope and challenges*) focused on Design Principle 1 (~31% of occurrences, 10 out of 32). Behaviors in Category 2 (*guesstimating missing information*) focused on Design Principles 4, 1, and 2 (~57% of occurrences, 25, 17 and 16 respectively out of 102). Behaviors in category 3 (*projecting into solution space*) focused on Design Principles 3, 5, and 4 (~57% of occurrences, 14, 13 and 12 respectively out of 68). Behaviors in Category 4 (*implanting into design*

*process*) focused on Design Principles 2 and 5 (~28% of occurrences, 5 and 4 respectively out of 32).

Based on these findings, design principles can support design efforts as part of many different behaviors, a consideration rarely made transparent in DSR literature when articulating design principles (e.g., Gaudio et al., 2016; Tuunanen & Peffers, 2018). However, the design professionals' decisions to focus on certain design principles as part of their behaviors does not tell how they applied the principles, e.g., whether they followed the principles verbatim, extended the ideas contained therein, ignored the prescriptions offered, or applied them differently. These questions lead us to investigate the second dimension: *application modes*.

## 4.2 Dimension: Application Modes

This set of findings responds to the following subquestion: *How do design professionals apply the potentially useful design principles in a new design situation?* Through our analysis, we identified nine modes of application across three categories: *simple application*, which describes the direct and straightforward use of design principles; *selective application*, which describes some modification of the design principle during its application; and *implicit application*, which includes modes such as ignoring and disregarding. Table 9 describes these application modes. Appendix A provides illustrative quotes.

**Category 1: Simple application** describes a designer's focus on one design principle or orchestration of multiple design principles, as illustrated in these quotes:

*Yes, it means someone must then take the minutes. Well, usually one has to [inhale]... audio-record the whole thing. Someone must do it ... An assistant must assume the task. This assistant surely has to write ... hmm ... so this expert one said this and that ... Then expert two this and that ... One has to record all that in the minutes. It must happen separately. Can also be done online ... there in the presentation view so to say ... it's ... usually preferable. (Participant P; application mode: focus; Design Principle 5)*

*One has to consider that, together with principle three ... In principle three it's required that relevant knowledge that's demanded or requested be made available, and the old cases also belong to that. They can be, so to say, linked, and then directly retrieved. (Participant P; application mode: orchestration; Design Principles 3 and 6)*

**Category 2: Selective application** describes the designer modifying, extending, refining, enhancing, or elaborating a design principle, as illustrated in these quotes:

*I would say, if you had to make a decision a little bit faster, then you have to look at the worst-case scenario, basically the most extensive one, because a hospital operation always contains ... the simple case medical practice, and in the hospital, you have the whole hierarchy of doctors with different, well, supervisors. ... [short pause] That's important, right.* (Participant T; application mode: contextualization; Design Principle 1)

*Of course, there is also the question whether the doctors put their contributions in there themselves, or does someone write a protocol and [short pause] does it for them? [short pause] Will it also be integrated with other systems?* (Participant V; application mode: extension; Design Principle 4)

*Of course, you would have to clarify that as well to come to a solution so you really can adjust the system to the processes in the best possible way.* (Participant V; application mode: extension; Design Principle 4)

*The way I see it, there aren't [short pause] representatives from the sciences are not necessarily there. Well, of course doctors are probably [short pause] more or less up to date when it comes to technology.* (Participant V; application mode: refinement; Design Principle 7)

**Category 3: Implicit application** describes the designer considering a design principle in different ways, such as referring to it ex post to rationalize a decision, ignoring it, or willfully disregarding it, as illustrated in these quotes:

*Ah, it occurs to me now. You need another thing, of course ... Each person, each of the experts, also needs access. So, personal view [short pause] so what's clear here is that the group database needs various views ... presentation view and also each personal view of each party.* (Participant P; application mode: internalization; Design Principle 4)

*I think you always need a good stakeholder-management, um, how it is in principle 5 too.* (Participant U; application mode: rationalization; Design Principles 3 and 5)

*And, um, [short pause] exact and current content, um, is always what you put in, um. There is a saying in IT—shit in, shit out—you really have to make sure that the simplicity of, um, mode of presentation, input mask is simple, because only then you can also, um, guarantee quality of data within the tool.* (Participant U; application mode: willful disregard; Design Principle 9)

The participants considered the design principles as part of their design behaviors in 82% of the segments (simple applications 37%, selective applications 28%, and implicit applications 17%). Figure 5 summarizes these application modes, aggregated across all participants. A comparison of design professionals (Table 10) shows how they were different.

Several observations are evident from the table. First, the fraction of occurrences *not involving* the application of design principles (top row) varies across participants (46% for participant R and 7% for Participant T). The fraction of occurrences *involving* simple or selective application of design principles also varies across participants, but it shows the exact opposite (18% and 9% for Participant R compared to 45% and 43% for Participant T). This contrast indicates the “effortful” manner participants engage in when they use design principles.

Second, we observe the difference between simple application and selective application. Participant P shows the greatest difference (simple application 60% compared to selective application 2%), whereas Participants T (simple application 45%, selective application 43%) and S (simple application 35%, selective application 33%) show the least. Finally, we see a pronounced difference among participants in their implicit application of design principles. Participant Q shows the highest fraction of segments with implicit application of design principles (53%) compared to the other participants. These observations again indicate how design professionals do not simply follow available design principles. We explored this further by examining the differences across the design principles as summarized in Figure 6.

Figure 6 shows Design Principle 4 to be the most frequently applied principle, whereas Design Principles 8 and 9 were applied the least. *Simple application* remained the most frequent category, with *selective application* a close second, and *implicit application* the least frequent (with some minor violations).

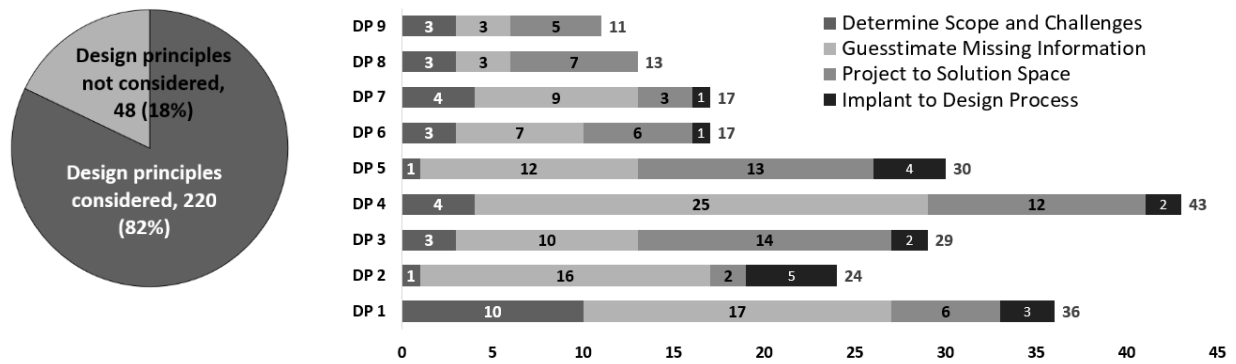
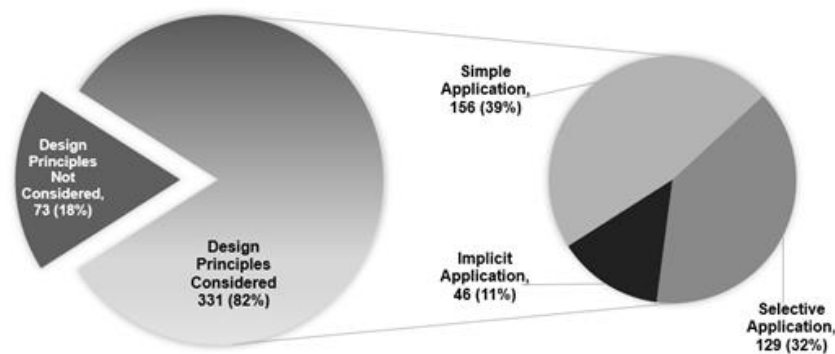


Figure 4. Relative Frequency of Design Behaviors (for Each Design Principle)

Table 9. Application Modes (How Designers Apply Design Principles)

Application mode	Description
<b>Category 1: Simple application</b>	
Focus	Designer focuses on a single design principle.
Orchestration	Designer draws explicit connections between two or more design principles.
<b>Category 2: Selective application</b>	
Contextualization	Designer instantiates a design principle in a particular context.
Extension	Designer expands/adds details to a design principle during its application.
Refinement	Designer refines part of a design principle during its application.
<b>Category 3: Implicit application</b>	
Internalization	Designer uses a design principle but not does not mention it explicitly.
Ignoring	Designer considers a potentially applicable design principle but does not use it.
Rationalization	Designer looks to a design principle to support something done earlier.
Willful disregard	Designer acknowledges a design principle but does something different.



Note: The difference in total number of occurrences (Figure 4 vs. Figure 5) is the consequence of the coding approach (see Table 7).

Figure 5. Relative Frequency of Application Modes—Aggregated

Table 10. Relative Frequency of Modes of Application—Across Participants

Aggregate	Design principles	P	Q	R	S	T	U	V
18%	Not considered	25%	9%	46%	23%	7%	13%	29%
37%	Simple application	60%	22%	18%	35%	45%	37%	24%
28%	Selective application	2%	16%	9%	33%	43%	17%	28%
17%	Implicit application	13%	53%	27%	10%	5%	33%	19%

Note: Rounded to the nearest percent

### 4.2.1 Simple Application

This category describes how a design professional applies one (focus) or multiple (orchestration) design principles. Table 11 shows the frequency of each application mode.

Out of 156 total occurrences in the *simple application* category, 116 (74%) focused on the application of a single design principle, whereas the remaining 40 (26%) orchestrated the application of multiple principles. The principles most often applied with others (orchestration mode) were Design Principle 3 (along with Design Principles 1, 2, 4, and 6), Design Principle 4 (along with Design Principles 1, 3, and 5), and Design Principle 5 (along with Design Principles 1, 3, and 4). Consider the following examples:

*So principle 5 ... aha, that's important now for principle 3. I have to change back because I have a new insight.* (Participant T; application mode: orchestration; Design Principles 3 and 5)

*maybe with principle 4, which says, as individual/group, so it is possible that even if it is an individual, then he is not directly involved but someone from outside and then he doesn't see the information, but he can look at the case. He must look at the case beforehand.* (Participant T; application mode: orchestration; Design Principles 4 and 5)

To explore why Design Principles 3, 4, and 5 were applied more frequently with an orchestration mode, we examined these design principles more closely. Unfortunately, we could not discern what made these different from other design principles. We can nevertheless offer the following interpretation. When design principles are formulated in a DSR project, the researchers analytically separate design lessons from other lessons in the project. When reusing them, design professionals must integrate the set of principles into their design efforts by reassembling some of its components. A better understanding of this application mode would inform the formulation of more useful design principles.

### 4.2.2 Selective Application

This category describes how the participants contextualized, extended, or refined the design principles. Table 12 shows the frequency of these application modes.

The 129 occurrences in the *selective application* category consisted of 45 occurrences of

contextualization (35%), 47 occurrences of extension (36%), and 37 occurrences of refinement (29%). Consider the following examples:

*The modes of presentation, it's important because every doctor has somewhat different expectations probably [about] what's the best way to, um, deal with the information that comes from the software.* (Participant U; application mode: contextualization; Design Principle 4)

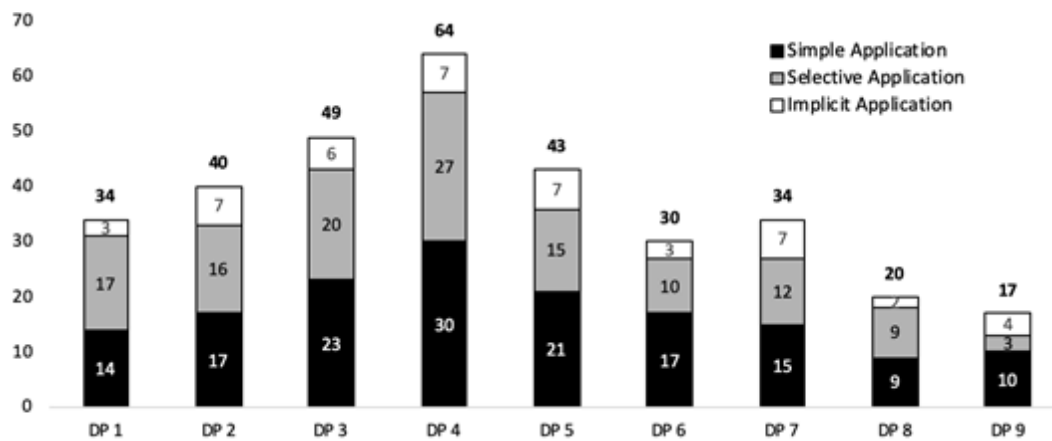
*That you can trace results in methods of treatment through the experts and make a note of it in the software, to give points of reference, um, for the doctors too, and also for future research, simply what the doctors do, um, to give points of reference.* (Participant U; application mode: contextualization; Design Principle 4)

*So that's basically about the information that the system administrates and provides, that all relevant information are there, that they probably are also easily accessible, that exactly.* (Participant V; application mode: refinement; Design Principle 3)

Here as well, some design principles (particularly Design Principles 1-4) were more amenable than others to the selective application mode. Design Principle 4 (*allowing different presentation modalities*) was applied with the contextualization mode 13 times (29% of occurrences of this mode) and with the extension mode 11 times (23% of occurrences of this mode). Design Principle 3 (*making knowledge visible on demand*) was refined (12 times), extended (12 times), and contextualized (7 times). The two other principles applied selectively were Design Principle 1 (extended 10 times) and Design Principle 2 (refined 8 times). Again, a closer reading of the design principles failed to reveal any intrinsic properties to explain why some were used more often with the selective application mode. Once the reasons for selective application are grasped, we can also understand how to move from a nomothetic formulation of design principles to an idiosyncratic one. This understanding will, in turn, help in crafting more useful design principles.

### 4.2.3 Implicit Application

This category captures some rather unexpected application modes, such as ignoring or willfully disregarding a design principle. Table 13 summarizes the frequencies. Although few occurrences were observed, they pointed to interesting possibilities.



Note: The different total frequencies (see Figure 4) are the result of our coding approach (see Table 7).

**Figure 6. Frequency of Application Modes Across Design Principles**

**Table 11. Simple Application: Focus vs. Orchestration**

Design principle	Simple application		Comments
	Focus	Orchestration	
Design Principle 1	11	3	-
Design Principle 2	16	1	-
Design Principle 3	15	8	Used with Principles 1, 2, 4, 5, and 6
Design Principle 4	23	7	Used with Principles 1, 3, and 5
Design Principle 5	11	10	Used with Principles 1, 3, and 4
Design Principle 6	15	2	-
Design Principle 7	13	2	-
Design Principle 8	6	3	-
Design Principle 9	6	4	-
Aggregated	116	40	

**Table 12. Selective Application: Contextualization vs. Extension vs. Refinement**

Design principle	Selective application			Comments
	Contextualization	Extension	Refinement	
Design Principle 1	4	10	3	Extended
Design Principle 2	5	3	8	Refined
Design Principle 3	7	11	12	Refined, extended, contextualized
Design Principle 4	13	11	3	Contextualized, extended
Design Principle 5	4	5	6	-
Design Principle 6	3	1	6	-
Design Principle 7	6	2	4	-
Design Principle 8	1	4	4	-
Design Principle 9	2	-	1	-
Aggregated	45	47	37	



**Table 13. Implicit Application: Internalization vs. Ignoring vs. Rationalizing vs. Willful Disregard**

Design principle	Implicit application				Comments
	Internalization	Ignoring	Rationalizing	Willful disregard	
Design Principle 1	1	2	-	-	-
Design Principle 2	3	3	1	-	<i>Internalization, ignoring</i>
Design Principle 3	2	3	-	1	-
Design Principle 4	2	3	-	2	<i>Multiple modes</i>
Design Principle 5	-	4	-	3	<i>Ignoring, disregard</i>
Design Principle 6	-	3	-	-	<i>Ignoring</i>
Design Principle 7	-	4	1	3	<i>Ignoring, disregard</i>
Design Principle 8	-	2	-	-	-
Design Principle 9	1	2	-	1	-
<b>Aggregated</b>	<b>9</b>	<b>26</b>	<b>2</b>	<b>10</b>	

We identified only 47 occurrences in the *implicit application* category. Of these, 26 (55%) captured ignoring the design principle (distributed fairly evenly across the design principles), 10 occurrences (21%) showed willful disregard (select design principles), and 9 occurrences (19%) showed internalization (select design principles). Consider the following examples:

*and ... for me now it would be like ... the case or the patient ... each of them ... yeah, every single patient file. ... it's the central object besides a list of ... the cases to be handled. They are, of course, some kinds of ... patient files that belong to them and the notes that are written, so that you, well, ... the treatment that's ... the decision that's been made ... so that you can also retrieve them again ... It comes up to a kind of archive and access function.* (Participant Q; application mode: ignoring; Design Principle 3)

*Well, that also leads to the question: can you do more with a system like that than just simply make the existing work sequences more efficient? Can you also make them more efficient so you will you get to a better result? So is there a way to do that?* (Participant V; application mode: willful disregard; Design Principle 5)

*Then somewhere you would also have a GUI, administration GUI, um, ideally available through, um, internet. Then you have some software on it that represents the whole logic. And, um, then the experts also have access to the GUI via user ID and password, for example.* (Participant S; application mode: internalization; Design Principle 4)

Participants sometimes ignored a design principle, and at other times willfully disregarded a design principle (particularly Design Principles 5 and 7). A closer reading of Design Principle 5 (*facilitate meaningful participation*) and Design Principle 7 (*facilitate integration of expert and evidence-based knowledge*) suggested one possibility for this disregard. Perhaps they could have been better described and elaborated, especially when it comes to the notion of “expert,” “meaningful participation,” and “convergent and divergent discussions.” The few occurrences do, however, suggest the need to ensure that useful design principles are crafted.

Finally, we noted a few occurrences of two other application modes in this category—internalization (9 occurrences) and rationalizing (2 occurrences). Although it is tempting to overlook these as merely outliers, we acknowledge that these application modes may appear more often in other contexts.

### 4.3 Mapping across the Dimensions: Pathways for the Use of Design Principles

In this section, we identify and visually display pathways for the use of design principles. Consider, for example, *Scenario A*: Faced with a design situation, a design professional embarks on a certain design behavior (e.g., determine scope and challenges). To do this, she considers the design principles available as a source of knowledge and identifies a potentially helpful principle or some of them (e.g., Design Principle 3). Once identified, she applies the design principle in a certain way (e.g., simple application) to proceed toward the desired goal.

In contrast, consider *Scenario B*: Faced with a design situation, a design professional has access to potentially useful design principles. She explores the design principles and decides to focus on one (e.g.,

Design Principle 4). Then, she explores various ways in which it may be applied (e.g., selective application) and uses it to engage in various design behaviors (e.g., guesstimating missing information).

Scenario A is driven by a goal, where the design professional works backward to find and then apply the design principles as needed. It resembles *backward chaining*, a term used to describe an inference method used in the design of inference engines and artificial intelligence (AI) applications (Russell & Norvig, 2002). We find analogs to this pathway elsewhere, e.g., retrieval and adaptation (Purao et al., 2003), demand-side reuse (Sen, 1997), studies of task-oriented information-seeking (Kim, 2007), models of top-down human information-processing (Lindsay & Norman, 2013), and rational models of decision-making (Hastie & Dawes, 2010; see also the critiques in Simon, 1979). The following quote illustrates backward chaining:

*So, for example, or, um, alignment ... with other data sources, so it is definitely the case that you have different systems for different, um, user data, so basically patient data and then for cases. So, if there are such, such departments, then you have to make sure that they are aligned automatically and that not everything has to be typed in once.* (Participant T; application: simple; behavior: guesstimating missing information; Design Principle 9)

In contrast, Scenario B is driven by the availability of design principles. The design professional begins with an exploration of design principles and works forward to consider their usefulness in the design situation. It resembles *forward chaining*, a term used to describe an exploratory strategy used in AI applications (Russell & Norvig, 2002). We find analogs to this pathway elsewhere, such as evaluating and debugging a usage context (Rosson & Carroll, 1996), exploratory models of information-seeking (White & Roth, 2009), bottom-up human information-processing (Lindsay & Norman, 2013), and intuitive decision-making (Gigerenzer & Goldstein, 1996). The following quote illustrates forward chaining:

*Simplify, support the integration of expert knowledge and knowledge based on evidence. Allow convergent and divergent discussions. So, the blog that you have for discussions, um, you have to think about, if there are other prior possibilities to communicate.... Otherwise, there definitely has to be a moderator during the meeting who solves this organizationally and moderates.* (Participant S; application: simple; behavior: projecting to Solution Space; Design Principle 7)

Figures 7 and 8 show the two pathways, emphasizing the progression from different starting points: the goals (design behaviors) for backward chaining, and the design principles for forward chaining. The figures include pathways with high frequency (indicated by a number on the arrow). Complete data for the pathways appears in Appendix B.

## 5 Discussion and Implications

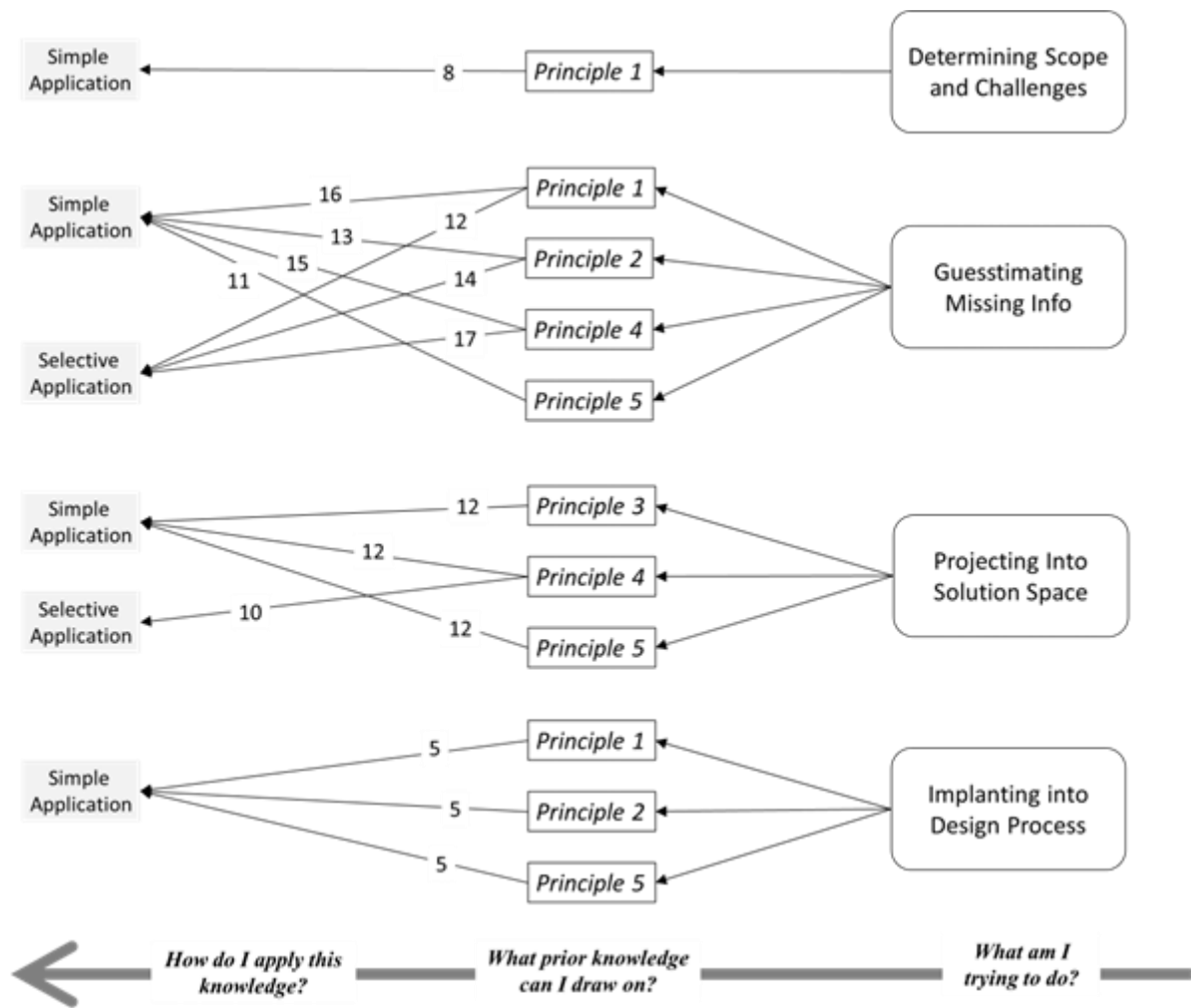
### 5.1 Design Behaviors and Application Modes: Novelty or Remix?

Our findings related to design behaviors have similarities with prior scholarship (e.g., Atman & Bursic, 1998; Cardella, et al., 2006; Newell & Simon, 1972; Purao et al., 2002; Ullman, 2010). The behavior categories we found map well against the descriptions of design professionals' efforts without access to design principles. This mapping indicates that the availability of design principles did not produce entirely novel design behaviors. Table 14 depicts the comparison.

The findings in Category 2 (*guesstimating missing information*) contribute to the contemporary understanding of design behaviors. The design behaviors in this category (drawing inferences, making assumptions, and posing rhetorical questions) mostly occurred when there was at least one design principle being considered. However, there was a possible downside to the availability of design principles. We observed how the participants did not look for additional pathways beyond what the design principles suggested. Our analysis revealed almost no instance where the participants explored new design possibilities after considering the design principles. The access to design principles appeared to hasten the move to design fixation (Jansson & Smith, 1991).

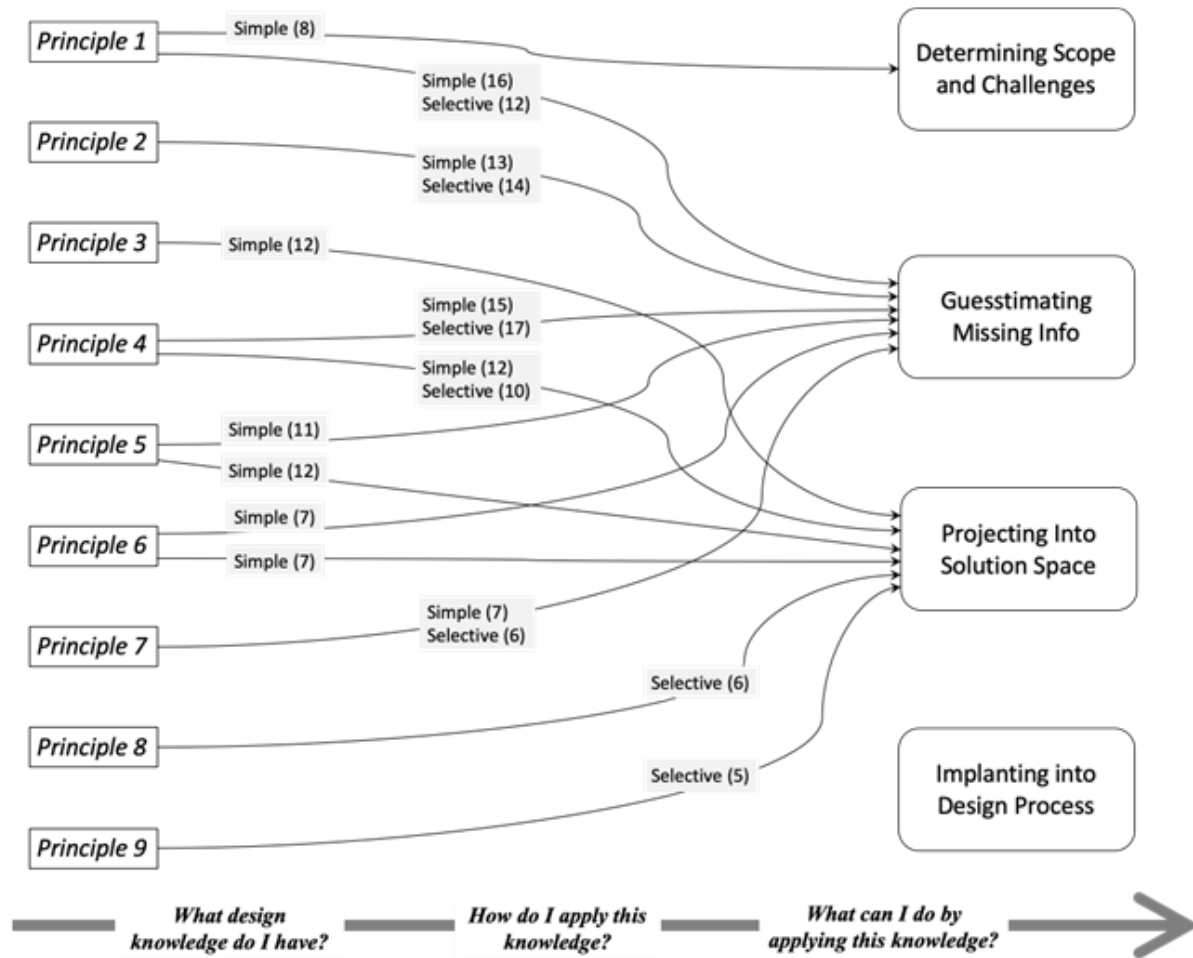
Our findings also show similarities to and extend prior empirical work in design knowledge reuse (Rosson & Carroll, 1996; Purao et al., 2003; Sen, 1997). These studies do not distinguish between design behaviors and application modes. In contrast, our work develops application modes with descriptive labels. Table 15 shows how our findings extend and refine findings from this stream.

More specifically, these findings suggest analytical refinements to the application modes implied in prior work, such as "retrieval and assessing applicability" (Rosson & Carroll 1996, Purao et al. 2003). The selective application category (specifically, contextualization, extension, and refinement) can be viewed as possible refinements of another activity noted in prior work, adaptation and integration (Purao et al., 2003; Sen, 1997). The implicit application category adds more nuance to prior work (e.g., Melton 2006), specifically ignoring and willful disregard. We contribute to this relatively underexplored area by offering a more granular understanding of how designers reuse design principles.



Note: Three of the four design behaviors led to consideration of multiple principles. The design professionals frequently traverse the same subset of design principles. The mode of application was often simple, along with some selective and little implicit applications.

**Figure 7. Use of Design Principles: A Backward Chaining Model**



Note: The design professionals select several design principles. The application mode is mostly simple or selective. Some principles contribute to multiple design behaviors.

Figure 8. Use of Design Principles: A Forward Chaining Model

Table 14. Design Behaviors: Comparing against Prior Work

Observed	Mapped to prior work (see Table 2)
<b>Category 1:</b> Determining scope and challenge	Understanding the problem and scope (e.g., Ullman, 2010) Retrieving prior knowledge (e.g., Purao et al., 2002)
<b>Category 2:</b> Guesstimating missing information	Retrieving prior knowledge (e.g., Purao et al., 2002) Looking for alternatives (e.g., Cardella et al., 2006)
<b>Category 3:</b> Projecting into solution space	Looking for alternatives (e.g., Cardella et al., 2006) Proposing a solution (e.g., Purao et al., 2002)
<b>Category 4:</b> Implanting into design procedure	Implementation and communication (e.g., Ullman, 2010)

**Table 15. Modes of Application: Comparison with Prior Work**

Observed	Refinements to prior work in design knowledge reuse
<i>Category 1:</i> simple application	See “Finding a usage context” in Rosson and Carroll (1996), “Retrieval” in Sen (1997), and “Retrieval” in Puroo et al. (2003).
<i>Category 2:</i> selective application	See “Debugging a usage context” in Rosson and Carroll (1996), “Adaptation” and “Evaluation” in Sen (1997), and “Adaptation” and “Integration” in Puroo et al. (2003).
<i>Category 3:</i> implicit application	See “Finding a usage context” and “Evaluating a usage context” in Rosson and Carroll (1996) and “Understanding Retrieved Artifacts” and “Selecting the Best Artifact” in Sen (1997).

According to our analysis, some design professionals tend to modify design principles when applying them. Whether such modification is essential is a question about the foundation of design principles (i.e., how prescriptive they should be) and their roles in design as a creative act (i.e., problems inherent in reuse—Favaro, 1991; Judicibus, 1996; Rech et al., 2007). This discussion can be applied to other domains (e.g., the laws of user interaction, Yablonski, 2020; cognitive design principles, Johnson, 2014; and the universal design principles, Lidwell et al., 2010). The DSR community may choose to emphasize that the design principles they generate are mesolevel (Kuechler & Vaishnavi, 2012) and therefore open to modification during application.

## 5.2 Pathways to Use: Two Assumptions about Design Principles

The two pathways we identify, *backward-chaining* and *forward-chaining*, provide alternative characterizations of how designers use design principles. The *backward-chaining* pathway describes how design principles can help designers reduce the search space (Simon, 1996). Researchers appear to rely on this logic when designing tools and approaches to facilitate reuse-based design in IS (Puroo et al., 2003; Umopathy & Puroo, 2008), engineering (e.g., Ahmed, 2005; Baxter et al., 2007; Demian & Fruchter, 2006; Fu et al., 2016), and computer science (e.g., Parnas & Clements, 1986; Tiwana & Ramesh, 2001). However, the application modes of design principles vary. This variance may explain why design knowledge reuse remains problematic in spite of such tools and approaches.

In contrast, acknowledging the *forward-chaining* pathway may open new directions to help designers during a design session. Classic definitions of forward-chaining (Feigenbaum et al., 1988, p. 318) describe it as starting with facts, using these to draw some conclusions, which then act as new facts in drawing more conclusions until the problem is solved. A straightforward application of this strategy can be difficult because designers often lack a clear understanding of what the phrase “until the problem is solved” means in a design context (see Rittel

& Weber 1973). Implicit application modes (e.g., ignoring and willful disregard) can be seen as efforts to prune some of the paths in a large solution space. To the best of our knowledge, no tools or approaches have been designed to support such a strategy.

By supporting designers to make effective design decisions in an unfamiliar domain, both pathways promote a move to a strong (instead of weak) approach to design<sup>7</sup> (Vessey & Glass, 1998) whose methods are designed to address a specific (instead of generic) problem type (Newell, 1969). The backward-chaining pathway describes the recognition of problems in the unfamiliar domain as the first step before moving to an exploration of specific lessons contained in the design principles. On the other hand, the forward-chaining pathway describes an exploration of the design principles as the first step before assessing their applicability to the problem. Table 16 summarizes the observations.

## 5.3 Drawing Lessons: Crafting More Useful Design Principles

We found a variety of ways that designers apply design principles, which map well against perspectives suggested in knowledge use (e.g., Forsgren et al., 2018; van den Hooff & de Ridder, 2004), knowledge transfer (e.g., Inkpen & Tsang, 2005; Mowery et al., 1996), and knowledge translation (e.g., Graham et al., 2006). Acknowledging these variations can inform DSR scholarship on how to respond more effectively to the dual goals of knowledge accumulation and usefulness to design practitioners.

More specifically, the findings offer new possibilities for crafting more useful design principles. One possibility is *specialization*. Some design principles may explain meta-requirements (see Walls et al., 1992), others may emphasize desirable capability clusters (see operational principles from Gregor & Jones, 2007), while others still may provide lessons about the process (see Gregor & Jones, 2007). The differences in application modes may provide clues about why some design principles are useful at different stages of design and how they may be crafted to support design work (Gregor et al., 2020).

like a monkey wrench, is designed to adjust to a multiplicity of problems, but solve none of them optimally.”

<sup>7</sup> Vessey and Glass (1998, p. 99) describe it thus: “A strong method, like a specific size of wrench, is designed to fit and do an optimal job on one kind of problem; a weak method,

**Table 16. How the Use Pathways Point to Possibilities for Future Research**

	Use pathways	
	Backward-chaining	Forward-chaining
<b>Assumptions about design principles</b>	Helping designers narrow the design space	Helping designers explore the design domain
<b>Possibilities for future research</b>	New approaches to support design knowledge reuse	New approaches to support pruning of search space
<b>Prior work</b>	Reuse-based design approaches (Umapathy & Puroo 2008) Weak and strong design approaches (Vessey & Glass, 1998)	

A second possibility is *recursive elaboration*. DSR scholars often articulate design principles as a flat structure with scant attention to differences in abstraction levels or interconnection across principles (e.g., Gaudioso, 2016). Possibilities such as the orchestration application mode suggest the need to acknowledge such interconnections and point to new possibilities for specifying design principles, much like a network of assertions.

A third possibility for crafting more useful design principles involves *intentionally limiting* the content (for example, by focusing on “what” as opposed to “how”). This move provides both novice and expert designers with guidance to overcome design fixation (Jansson & Smith, 1991), promote creativity with constraints (Baskerville et al., 2016; Costello & Keane, 2000), and acknowledge designer expertise (Cross, 2004). Our findings about implicit application modes point to this possibility. Table 17 summarizes these lessons.

#### 5.4 Limitations of Our Work

Our study has limitations, which also present opportunities for future work. First, the design principles we used are from prior research (Gaudioso et al., 2016) and not uniform in structure and content. Future studies could address this issue if the DSR community norms evolve to more structured formalization of design principles (e.g., Gregor et al., 2020).

Second, our work identifies individual designer behaviors and application modes with verbal protocols. Future work could explore how teams, including users, may collaborate and coordinate (e.g., Dong, 2005) when reusing design principles. Future researchers could also consider additional dimensions for data analysis (e.g., reactivity, where verbalization itself may alter internal processes—Leow, 2002).

Third, relying on a small number of design professionals may be seen as a limitation. However, as noted earlier, the participants had significant and relevant experience, a consideration that is more important compared to the number of participants (see,

e.g., Khatri & Vessey, 2016; Price & Shanks, 2011; Puroo et al., 2002; Wang, 1996). They are representative of the population of designers likely to reuse the design principles. Further, as design activities are messy, the insights provided by studies like ours can be more meaningful than quantitative evidence obtained through controlled experiments, particularly when the design task is realistically complex. In spite of these arguments, future research may be able to recruit more participants with the right characteristics and consider different design situations and design principles to derive more generalized patterns of use

Finally, some readers may consider the lack of domain expertise for the participants as a limitation of our study. This is likely to be representative of design situations, for instance, when software designers consider an unfamiliar problem space. Such design situations arguably provide the appropriate context to study how designers reuse design principles. However, future studies could vary these parameters to generate different combinations of designers and design situations to add to our findings.

## 6 Concluding Remarks

Design principles are important because they help DSR scholars encapsulate and disseminate research outcomes and build a cumulative body of knowledge (Gregor et al., 2020). They are also of interest to design practitioners because they can provide actionable knowledge (Romme & Endenburg, 2006) by communicating “knowledge about creating other instances of artifacts that belong to the same class” (Sein et al., 2011, p. 39). These dual goals may sometimes be synergistic and at other times may hinder progress. One approach to examining their interplay is to develop a greater understanding of how design principles capture design knowledge that can be reused. Capturing design knowledge in design principles has been the focus of much contemporary research (Gregor et al., 2020; Iivari, et al., 2020). Their actual reuse in design practice has not received much attention.

**Table 17. Crafting Useful Design Principles**

Lesson	Description	Rationale
Specialization	<ul style="list-style-type: none"> <li>• Distinguish between meta-requirements and desirable capability clusters.</li> <li>• Distinguish between a process and artifact focus.</li> </ul>	<ul style="list-style-type: none"> <li>• Improve transparency and clarity of design principles.</li> <li>• Support reuse and different design stages and application modes.</li> </ul>
Recursive elaboration	<ul style="list-style-type: none"> <li>• Specify design principles at different levels of abstraction and interconnect them.</li> </ul>	<ul style="list-style-type: none"> <li>• Acknowledge possibilities for the orchestration application mode.</li> </ul>
Intentionally limit the content	<ul style="list-style-type: none"> <li>• Focus on “what” as opposed to “how” when formulating design principles</li> </ul>	<ul style="list-style-type: none"> <li>• Acknowledge the designer’s expertise to support the implicit application mode.</li> </ul>

Motivated by these observations, we explored how design professionals use design principles in a new design situation. Several elements ensured the robustness of our investigation. We relied on design principles that were published at a premier IS workshop (Gaudioso, 2016), we recruited experienced designers as study participants, and we employed a research method that surfaces the designer’s thinking (the think-aloud method, see van Someren et al., 1994) while allowing multiple theoretical perspectives to influence our analyses.

We make five key contributions to DSR scholarship. First, we propose new vocabularies (application modes) to help us better understand how design principles are used. Second, we describe two pathways for the use of design principles. Third, we derive three possibilities for crafting more useful design principles. Fourth, we highlight how the think-aloud method can provide insights into the microprocesses of designer behaviors with design principles. Finally, our analyses bring

together perspectives from two communities: the DSR community (from the IS discipline) and the design studies community (from architecture and engineering).

We hope that our findings will enhance the appreciation of design principles as a knowledge form produced by DSR scholars, provide fresh insights to DSR scholars to craft more useful design principles, and inspire more empirical investigations to complement the conceptual discourse about design principles.

## Acknowledgments

We remain grateful to the expert design professionals who participated in our think-aloud sessions. We also thank Dr. Dorothy Leidner and the anonymous reviewers for their constructive comments. This research was supported by the Research Funds of the University of Liechtenstein under grant FFF\_wi\_20\_1. The work of the second author was supported by a grant from Bentley University.

## References

- Ahmed, S. (2005). Encouraging reuse of design knowledge: a method to index knowledge. *Design Studies*, 26(6), 565-592.
- Akin, Ö. (2001). Variants in design cognition. In C. Eastman, M. McCracken & W. Newstetter (Eds.), *Design knowing and learning: Cognition in design education* (pp. 105-124). Elsevier Science.
- Amabile, T. M. (1988). A model of creativity and innovation in organizations. *Research in Organizational Behavior*, 10(1), 123-167.
- Argyris, C., Schön, D.A. (1974). *Theory in practice: Increasing professional effectiveness*. Jossey-Bass.
- Atman, C. J., & Bursic, K.M. (1998). Verbal protocol analysis as a method to document engineering student design processes. *Journal of Engineering Education*, 87(2), 121-132.
- Baskerville, R. Pries-Heje, J. (2010). Explanatory Design Theory. *Business and Information Systems Engineering*, 5, 271-282.
- 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., Kaul, M., Pries-Heje, J., Storey, V., & Kristiansen, E. (2016). Bounded creativity in design science research. *Proceedings of the 37th International Conference on Information Systems*.
- Baxter, D., Gao, J., Case, K. et al. (2007) An engineering design knowledge reuse methodology using process modelling. *Res Eng Design* 18, 37-48.
- Borchers, J. O. (2001). A pattern approach to interaction design. *AI & Society*, 15(4), 359-376.
- Cardella, M. E., Atman, C.J., & Adams, R.S. (2006). Mapping between design activities and external representations for engineering student designers. *Design Studies*, 27(1), 5-24.
- Cash, P. Hicks, B., & Culley, S. (2015). Activity Theory as a means for multi-scale analysis of the engineering design process: A protocol study of design in practice. *Design Studies*, 38, 1-32.
- Chaturvedi, A. R., Dolk, D. R., & Drnevich, P. L. (2011). Design principles for virtual worlds. *MIS Quarterly*, 35(3) 673-684.
- Costello, F. J., & Keane, M. T. (2000). Efficient creativity: Constraint-guided conceptual combination. *Cognitive Science*, 24(2), 299-349.
- Cramer-Petersen, C.L., Christensen, B.T., & Ahmed-Kristensen, S. (2019). Empirically analysing design reasoning patterns: Abductive-deductive reasoning patterns dominate design idea generation. *Design Studies*, 60, 39-70.
- Cross, N. (2001a). Design cognition: Results from protocol and other empirical studies of design activity. In: Eastman, C.; Newstetter, W. and McCracken, M. (eds.). *Design knowing and learning: cognition in design education* (pp. 79-103). Elsevier.
- Cross, N. (2001b). Designerly ways of knowing: Design discipline versus design science. *Design Issues*, 17(3), 49-55.
- Cross, N. (2004). Expertise in design: an overview. *Design Studies*, 25(5), 427-441.
- Cross, N., Christiaans, H., & Dorst, K. (1994). Design expertise amongst student designers. *Journal of Art & Design Education*, 13(1), 39-56.
- Demian, P., Fruchter, R. (2006). An ethnographic study of design knowledge reuse in the architecture, engineering, and construction industry. *Research in Engineering Design*, 16, 184-195
- Denning, P., & Dargan, P. (1996). Action-centered design. In T. Winograd (Ed. ), *Bringing design to software* (pp. 105-119). ACM Press.
- Dong, A. (2005). The latent semantic approach to studying design team communication. *Design Studies*, 26(5), 445-461.
- Dorst, K. (2006). Design problems and design paradoxes. *Design Issues*, 22(3), 4-17.
- Dorst, K., & Cross, N. (2001). Creativity in the design process: co-evolution of problem-solution. *Design studies*, 22(5), 425-437.
- Ericsson, K. A., & Crutcher, R. J. (1991). Introspection and verbal reports on cognitive processes: Two approaches to the study of thinking: A response to Howe. *New Ideas in Psychology*, 9(1), 57-71.
- Ericsson, K. A., & Simon, H. A. (1993). *Protocol analysis: Verbal reports as data*. The MIT Press.
- Ericsson, K. A., & Simon, H. A. (1998). How to study thinking in everyday life: Contrasting think-aloud protocols with descriptions and explanations of thinking. *Mind, Culture, and Activity*, 5(3), 178-186.



- Favaro, J. (1991). What price reusability? A case study. *Proceedings of the First International Symposium on Environments and Tools for ADA*.
- Feigenbaum, E. A., McCorduck, P., & Nii, H. P. (1988). *The rise of the expert company: How visionary companies are using artificial intelligence to achieve higher productivity and profits*. Times Books.
- Fischer, C., Winter, R. & Wortmann, F. (2010). Design Theory. *Business Information Systems and Engineering*, 2, 387-390.
- Forsgren, N., Sabherwal, R., & Durcikova, A. (2018). Knowledge exchange roles and EKR performance impact: Extending the theory of knowledge reuse. *European Journal of Information Systems*, 27(1), 3-21.
- Fowler, M. (1996). *Analysis patterns: Reusable object models*. Addison-Wesley Longman.
- Fu, K.K., Yang, M.C., Wood, K.L. (2016). Design principles: Literature review, analysis, and future directions. *Journal of Mechanical Design* 138, 1-13.
- Gamma, E. (1995). *Design patterns: Elements of reusable object-oriented software*: Pearson Education India.
- Garud, R. (1997). On the distinction between know-how, know-why, and know-what. *Advances in strategic management*, 14, 81-101.
- Gaudio, C., Jain, H., & Puro, S. (2016). *A patient-centered CDSS for cancer treatment decisions*. Paper presented at the Pre-ICIS Workshop on Information Technologies and Systems, Dublin, Ireland.
- Gero, J. S. (1990). Design prototypes: A knowledge representation schema for design. *AI Magazine*, 11(4), 26-36.
- Gero, J. S., & Kannengiesser, U. (2014). The function-behaviour-structure ontology of design. In A. Chakrabarti, L. Blessing (Eds.), *An anthology of theories and models of design* (pp. 263-283). Springer.
- Gero, J. S., & Mc Neill, T. (1998). An approach to the analysis of design protocols. *Design Studies*, 19(1), 21-61.
- Gigerenzer, G., Goldstein, D. G. (1996). Reasoning the fast and frugal way: Models of bounded rationality. *Psychological Review*, 103 (4): 34-59.
- Glaser, B. G. (1978). *Theoretical sensitivity: Advances in the methodology of grounded theory*. The Sociology Press.
- Graham, I. D., Logan, J., Harrison, M. B., Straus, S. E., Tetroe, J., Caswell, W., & Robinson, N. (2006). Lost in knowledge translation: Time for a map? *Journal of Continuing Education in the Health Professions*, 26(1), 13-24.
- Gregor, S., & Jones, D. (2007). The anatomy of a design theory. *Journal of the Association for Information Systems*, 8(5), 312-335.
- Gregor, S., Chandra Kruse, S., & Seidel, S. (2020). The anatomy of a design principle. *Journal of the Association for Information Systems*, 21(6), 1622-1652.
- 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.
- Hastie, R., & Dawes, R. M. (2010). *Rational choice in an uncertain world: The psychology of judgment and decision making*. SAGE.
- Hatchuel, A., & Weil, B. (2002). CK theory. *Proceedings of the Herbert Simon International Conference on Design Sciences*.
- Hevner, A., & Chatterjee, S. (2010). Design science research in information systems. In *Design research in information systems* (pp. 9-22). Springer.
- Iivari, J. (2016). Information system artefact or information system application: That is the question. *Information Systems Journal*, 27(6), 753-774.
- Iivari, J., Hirschheim, R., & Klein, H. K. (2004). Towards a distinctive body of knowledge for Information Systems experts: coding ISD process knowledge in two IS journals. *Information Systems Journal*, 14(4), 313-342.
- Iivari, J., Hansein, M.R.P., & Haj-Bolouri, A. (2020) A proposal for minimum reusability evaluation of design principles, *European Journal of Information Systems*, 30(3), 286-303.
- Inkpen, A. C., & Tsang, E. W. (2005). Social capital, networks, and knowledge transfer. *Academy of management review*, 30(1), 146-165.
- Jansson, D. G., & Smith, S.M. (1991). Design fixation. *Design Studies*, 12(1), 3-11.
- Johnson, J. (2014). *Designing with the mind in mind: Simple guide to understanding user interface design guidelines*. Elsevier.
- Judicibus, D. D. (1996). Reuse: A cultural change. *Proceedings of the International Workshop on Systematic Reuse: Issues in Initiating and Improving a Reuse Program*.

- Katz, R., & Allen, T. J. (1982). Investigating the Not Invented Here (NIH) syndrome: A look at the performance, tenure, and communication patterns of 50 R & D Project Groups. *R&D Management*, 12(1), 7-20.
- Khatri, V., & Vessey, I. (2016). Understanding the role of is and application domain knowledge on conceptual schema problem solving: A verbal protocol study. *Journal of the Association for Information Systems*, 17(12), 759-803.
- Kim, J. (2007). Modeling task-based information seeking on the web: Application of information seeking strategy schema. *Proceedings of the American Society for Information Science and Technology*, 44(1), 1-13.
- Krueger, C.W. (1992). Software reuse. *ACM Computing Survey*, 24(2), 131-183.
- 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.
- Laing, S., Apperley, M., & Masoodian Masood. (2017). Investigating the effects of client imagery on the ideation process of graphic design. *Design Studies*, 53, 78-98.
- Latour, B. (2008). A cautious Prometheus? A few steps toward a philosophy of design (with special attention to Peter Sloterdijk). *Proceedings of the 2008 Annual International Conference of the Design History Society*.
- Le Masson, P., Dorst, K., & Subrahmanian, E. (2013). *Special issue on design theory: history, state of the arts and advancements*. Springer.
- Leow, R. P. (2002) Models, attention, and awareness in SLA: A response to Simard and Wong's "Alertness, orientation, and detection: The conceptualization of attentional functions." *Studies in Second Language Acquisition*, 24, 113-119.
- Lidwell, W., Holden, K., & Butler, J. (2010). *Universal principles of design, revised and updated: 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design*. Rockport.
- Lindsay, P. H., & Norman, D. A. (2013). *Human information processing: An introduction to psychology*. Academic Press.
- Lukyanenko, R., & Parsons, J. (2020). 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*, 21(5), 1115-1190.
- March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, 15(4), 251-266.
- Markus, M. L. (2001). Toward a theory of knowledge reuse: Types of knowledge reuse situations and factors in reuse success. *Journal of Management Information Systems*, 18(1), 57-93.
- Mathiassen, L. (1998). Reflective systems development. *Scandinavian Journal of Information Systems*, 10(1), 12.
- Mathiassen, L., & Purao, S. (2002). Educating reflective systems developers. *Information Systems Journal*, 12(2), 81-102.
- Melton, H. (2006). *On the usage and usefulness of OO design principles*. Paper presented at the Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and application, Portland, OR.
- Mowery, D. C., Oxley, J. E., & Silverman, B. S. (1996). Strategic alliances and interfirm knowledge transfer. *Strategic Management Journal*, 17(S2), 77-91.
- Newell, A. (1969). Heuristic programming: Ill-structured problems. In J. Aronofsky (Ed.), *Progress in operations research* (Vol. 3, pp. 360-414). Wiley.
- Newell, A., & Simon, H. (1972). *Human problem solving*. Prentice-Hall.
- Nonaka, I., & Konno, N. (1998). The concept of "ba": Building a foundation for knowledge creation. *California Management Review*, 40(3), 40-54.
- Nonaka, I., & Toyama, R. (2003). The knowledge-creating theory revisited: knowledge creation as a synthesizing process. *Knowledge Management Research & Practice*, 1(1), 2-10.
- Parnas, D. L., & Clements, P. C. (1986). A rational design process: How and why to fake it. *IEEE Transactions on Software Engineering*, SE-12(2), 251-257.
- Parsons, J. and Saunders, C., (2004). Cognitive heuristics in software engineering applying and extending anchoring and adjustment to artifact reuse. *IEEE Transactions on Software Engineering*, 30(12), 873-888.
- Price, R., & Shanks, G. (2011). The impact of data quality tags on decision-making outcomes and

- process. *Journal of the Association for Information Systems*, 12(4), 323-346.
- Pringle, A., & Sowden, P. T. (2017). Unearthing the creative thinking process: Fresh insights from a think-aloud study of garden design. *Psychology of aesthetics, creativity, and the arts*, 11(3), 344-358.
- Purao, S., Rossi, M., & Bush, A. (2002). Towards an understanding of the use of problem and design spaces during object-oriented system development. *Information and Organization*, 12(4), 249-281.
- Purao, S., Storey, V. C., & Han, T. (2003). Improving analysis pattern reuse in conceptual design: Augmenting automated processes with supervised learning. *Information Systems Research*, 14(3), 269-290.
- Rech, J., Decker, B., Ras, E., Jedlitschka, A., & Feldmann, R.L. (2007). The quality of knowledge: Knowledge patterns and knowledge refactorings. *International Journal of Knowledge Management*, 3(3), 74-103.
- Rittel, H., & Webber, M. M. (1973). 2.3 planning problems are wicked. *Polity*, 4, 155-169.
- Romme, A. G. L., & Endenburg, G. (2006). Construction principles and design rules in the case of circular design. *Organization Science*, 17(2), 287-297.
- Rosson, M.B. and Carroll, J.M. (1996). The reuse of uses in Smalltalk programming. *ACM Transactions on Computer-Human Interaction*, 3(3), 219-253.
- Russell, S., & Norvig, P. (2002). *Artificial intelligence: A modern approach*. Pearson Education, Inc.
- Saldaña, J. (2015). *The coding manual for qualitative researchers*. SAGE.
- Salovaara, A., & Tuunainen, V. K. (2013). Software developers' online chat as an intra-firm mechanism for sharing ephemeral knowledge. *Proceedings of the 34<sup>th</sup> International Conference on Information Systems*.
- Salovaara, A., & Tuunainen, V. (2015). Mediated sharing as software developers' strategy to manage ephemeral knowledge. *Proceedings of 23<sup>rd</sup> European Conference on Information Systems*.
- Schön, D. A. (1983). *The reflective practitioner*. Basic Books.
- Schön, D. A. (1987). *Educating the reflective practitioner: Toward a new design for teaching and learning in the professions*: Jossey-Bass.
- 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
- Seidel, S. & C. Urquhart. (2013). On emergence and forcing in information systems grounded theory studies: The case of Strauss and Corbin. *Journal of Information Technology* 28(3), 237-260.
- Sein, M. K., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action design research. *Management Information Systems Quarterly*, 35(1), 37-56.
- Sen, A., 1997. The role of opportunism in the software design reuse process. *IEEE Transactions on Software Engineering*, 23(7), pp.418-436.
- Simon, H. (1996). *The sciences of the artificial* (3rd ed.). MIT Press.
- Simon, H. A. (1979). Rational decision making in business organizations. *The American Economic Review*, 69(4), 493-513.
- Sio, U. N., Kotovsky, K., & Cagan, J. (2018). Silence is golden: The effect of verbalization on group performance. *Journal of Experimental Psychology: General*, 147(6), 939-944.
- Strauss, A. L., & Corbin, J. (1998). *Basics of qualitative research. Techniques and procedures for developing grounded theory* (2nd ed.). SAGE.
- Suwa, M., & Tversky, B. (1997). What do architects and students perceive in their design sketches? A protocol analysis. *Design Studies*, 18(4), 385-403.
- Suwa, M., Purcell, T., & Gero, J. (1998). Macroscopic analysis of design processes based on a scheme for coding designers' cognitive actions. *Design Studies*, 19(4), 455-483.
- Tiwana, A., & Ramesh, B. (2001). A design knowledge management system to support collaborative information product evolution. *Decision Support Systems*, 31(2), 241-262.
- Todd, P., & Benbasat, I. (1987). Process tracing methods in decision support systems research: exploring the black box. *MIS Quarterly*, 11(4), 493-512.
- Tsang, E. W., & Williams, J. N. (2012). Generalization and induction: Misconceptions, clarifications, and a classification of induction. *MIS Quarterly*, 36(3) 729-748.
- Tuunainen, T., & Peffers, K. (2018). Population targeted requirements acquisition. *European*

- Journal of Information Systems*, 27(6), 686-711.
- Ullman, D. G. (2010). *The mechanical design process*. McGraw-Hill.
- Umapathy, K. and Purao, S. (2008). Representing and accessing design knowledge for service integration. *Proceedings of the IEEE International Conference on Services Computing* (pp. 67-74).
- Urquhart, C., Lehmann, H., & Myers, M. D. (2010). Putting the “theory” back into grounded theory: Guidelines for grounded theory studies in information systems. *Information Systems Journal*, 20(4), 357-381.
- van Aken, J. E. (2001). *Improving the relevance of management research by developing tested and grounded technological rules*. Eindhoven Centre for Innovation Studies.
- van den Hooff, B., & de Ridder, J. A. (2004). Knowledge sharing in context: the influence of organizational commitment, communication climate and CMC use on knowledge sharing. *Journal of Knowledge Management*, 8(6), 117-130.
- van Someren, M. W., Barnard, Y. F., & Sandberg, J. A. (1994). *The think-aloud method: A practical guide to modelling cognitive processes* (Vol. 2): Academic Press.
- Vessey, I. Glass., R. (1998). Strong vs. weak approaches to systems development. *Communications of the ACM*, 41(4), 99-102.
- Visser, W. (2009). Design: One, but in different forms. *Design Studies*, 30(3), 187-223.
- Vitalari, N. P. (1985). Knowledge as a basis for expertise in systems analysis: An empirical study. *MIS Quarterly*, 9(3), 221-241.
- Vitalari, N. P., & Dickson, G. W. (1983). Problem solving for effective systems analysis: An experimental exploration. *Communications of the ACM*, 26(11), 948-956.
- Walls, J. G., G. W. Widmeyer and O. A. El Sawy. (1992). Building an information systems design theory for vigilant EIS. *Information Systems Research*, 3(1), 36-59.
- Wang, S. (1996). Toward formalized object-oriented management information systems analysis. *Journal of Management Information Systems*, 12(4), 117-141.
- White, R.W., & Roth, R.A. (2009). *Exploratory search: Beyond the query-response paradigm*. Morgan & Claypool.
- Wieringa, R. (2009). Design science as nested problem solving. *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*.
- Yablonski, J. (2020). *Laws of UX: Using psychology to design better products and services*. O'Reilly.

## Appendix A: How Design Principles are Used: Illustrative Quotes

Table A1. How Design Principles are Used: Illustrative Quotes

Design principle	Application mode	An illustrative quote	As a part of
<b>Principle 1:</b> Design the artifact as an integral part of the organizational and technological fabric/work practices	<i>Simple application: orchestration</i>	<i>If I go back to principle one, the workflow, and you have some meeting, then preliminary research takes places and add some point we also have the patient, the patient data. And there you must, then the two information go ... be matched. [short pause]"</i> (Participant S while orchestrating Design Principles 1 and 3)	<i>Project to Solution Space: Verify Solution</i>
<b>Principle 2:</b> Researcher embedding in the setting, high-fidelity prototypes	<i>Simple application: focus</i>	<i>A researcher has a special, a somewhat different focus, so he doesn't do just one treatment daily, but he um goes to conventions ... conducts experiments ... they are definitely two different types of users ... who play a role there.</i> (Participant T while identifying different stakeholders)	<i>Guesstimating missing information: making assumptions</i>
<b>Principle 3:</b> Make available and visible case-relevant knowledge demanded/requested	<i>Selective application: extension</i>	<i>and it's almost like... like... if we want to exaggerate... so... clients who have bought book X have also bought book Y... it's like... recommendations. it could be helpful... give recommendation, which cases would be suitable.</i> (Participant Q on which data to be collected)	<i>Guesstimating missing information: making assumptions</i>
<b>Principle 4:</b> Allow multiple presentation modalities with integrated summary, user-directed access to details	<i>Selective application: contextualization</i>	<i>Data protection is certainly ... but functional, functional not really. But the group, I mean, they all have the same roles. I mean if, if a doctor or a researcher is trusted with, so assigned the case, then he has to, he is bound to professional discretion, there is confidentiality.</i> (Participant T on data protection)	<i>Guesstimating missing information: making assumptions</i>
<b>Principle 5:</b> Facilitate, support, ensure meaningful participation in the case discussion from all experts	<i>Implicit Application: Willful disregard</i>	<i>well that also leads to the question, can you do more with a system like that than just simply make the existing work sequences more efficient? Can you also make them more efficient, so you will you get to a better result? So is there a way to do that?</i> (Participant V while considering changes in routines)	<i>Guesstimating missing information: Posing rhetorical questions</i>
<b>Principle 6:</b> Allow directed access to group memory and similar cases based on issues	<i>Simple application: focus</i>	<i>it means one needs only good search function for similar cases ... it happens online ... thus in these ten minutes...it can hardly take place before that ... except if the assistant who does the preparation, who does it here, who prepares each case ... then says ... there are already similar cases in the past</i> (Participant P on the relationship between components)	<i>Guesstimating missing information: making assumptions</i>
<b>Principle 7:</b> Facilitate/support integration of expert and evidence-based knowledge	<i>Implicit application: willful disregard</i>	<i>But it is still difficult. So, you would definitely have to, well basically you would have to talk to the same doctors regarding all these questions, how, how they... exactly, well, how exactly you can support that</i> (Participant V while considering changes in routines)	<i>Implanting into design Process: Recognizing stakeholders</i>
<b>Principle 8:</b> Preserve group memory	<i>Selective application: extension</i>	<i>Even though the software probably will have to be adapted to the technologies here and there, the data and what's behind it have to be very resistant and um long-living, to make sure, that you can also compare those um cases to each other in the future over 20, 30, 40 years from now.</i> (Participant U on database)	<i>Determining scope and challenges: establishing design scope</i>
<b>Principle 9:</b> Plan for and ensure accurate and current content	<i>Implicit application: ignoring</i>	<i>Contents that are accurate and actual... that's... based on my experience always difficult, to maintain the data. It's a big topic.</i> (Participant Q on which data to be collected)	<i>Guesstimating missing information: making assumptions</i>

## Appendix B. Use of Design Principles: Two Models

**Table B1. Use of Design Principles (DP): Backward Chaining Model**

Design principle	DP1			DP2			DP3			DP4			DP5			DP6			DP7			DP8			DP9					
Application mode	S	L	M	S	L	M	S	L	M	S	L	M	S	L	M	S	L	M	S	L	M	S	L	M	S	L	M	S	L	M
Determining scope and challenges	8	4	0	0	0	1	2	3	0	1	3	1	1	3	0	3	2	0	4	3	1	3	2	1	3	1	0	3	1	0
Guesstimating missing Info	16	12	2	13	14	0	8	8	0	15	17	2	11	7	2	7	4	1	7	6	2	4	1	0	2	1	1	2	1	1
Projecting into solution space	6	4	3	2	1	1	12	9	2	12	10	0	12	7	2	7	4	0	3	3	0	3	6	0	5	1	2	5	1	2
Implanting into design process	5	2	0	5	1	0	2	1	0	1	0	2	5	1	2	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0

Note: S = Simple application, L = Selective application, M = Implicit application. Please read across the rows (left to right) to emphasize the model depicted in Figure 7.

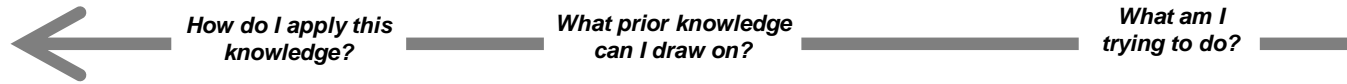
**Table B2. Use of Design Principles: Forward Chaining Model**

	Application mode	Determining scope and challenges	Guesstimating missing information	Projecting into solution space	Implanting into design process
<b>Design Principle 1</b>	Simple	8	16	6	4
	Selective	4	12	4	2
	Implicit	0	2	3	0
<b>Design Principle 2</b>	Simple	0	13	2	5
	Selective	0	14	1	4
	Implicit	1	0	1	0
<b>Design Principle 3</b>	Simple	2	8	12	2
	Selective	3	8	9	1
	Implicit	0	0	2	0
<b>Design Principle 4</b>	Simple	1	15	12	1
	Selective	3	17	10	0
	Implicit	1	2	0	2

<b>Design Principle 5</b>	Simple	1	11	12	5
	Selective	1	5	8	1
	Implicit	0	2	2	2
<b>Design Principle 6</b>	Simple	3	7	7	1
	Selective	2	4	4	0
	Implicit	0	1	0	0
<b>Design Principle 7</b>	Simple	4	7	3	0
	Selective	3	6	3	0
	Implicit	1	2	0	1
<b>Design Principle 8</b>	Simple	3	4	3	0
	Selective	2	1	6	0
	Implicit	1	0	0	0
<b>Design Principle 9</b>	Simple	3	2	5	0
	Selective	1	1	1	0
	Implicit	0	1	2	0

*Note:* Please read across the rows (left to right) to emphasize the model depicted in Figure 8. Reading across the group of three rows for each principle shows the different ways in which each principle was used.

**Table B3. Backward Chaining Model: Illustrative Quotes**



Application mode	Design principle (DP)	Design behavior	Illustrative quote
Simple application	DP 9	Guesstimating missing information	<ul style="list-style-type: none"> <li>• <i>That's basically um, what is that called, user input modification, right. If he enters a name now, then it should pop up rather than having to type everything. Um data plausibility check, right. [Long pause] for example um, treatment data, so if it is the same as last year, that can't be and things like that. Review age and things like that.</i></li> <li>• <i>So for example or um alignment ... with other data sources. So it is definitely the case that you have different systems for different um user data, so basically patient data and then for cases. So if there are such, such departments, then you have to make sure that they are aligned automatically and that not everything has to be typed in once.</i></li> <li>• <i>Um exactly, a current content ... , current doesn't only mean technical, but also organizational. So that it is current, then you would have to make sure that you can um update regularly. The application is supposed to be retrievable. (Participant T)</i></li> </ul> <p><i>Note: "current content" refers to "Plan for and ensure accurate and current content (Design Principle 9)."</i></p>
Selective application	DP 4	Implanting into design procedure	<ul style="list-style-type: none"> <li>• <i>And there also have to be cases of reference in the software, that you can, somehow create a profile of each patient, and this profile, which is always unique, so you can never have two cases which are the same, they have to have the points of reference to the other cases, that you can then um compare to each other.</i></li> <li>• <i>And this has to be um made very visible for the doctors, to compare and that they really can say the best, um, um best, with their best knowledge, with their experience, what the best method of treatment is for the patient.</i></li> <li>• <i>The modes of presentation...it's important because every doctor has somewhat different expectations probably, what's the best way to um deal with the information that comes from the software.</i></li> <li>• <i>There are people who can um deal with visual information very well, but there are also people who simply want to have it in text format. (Participant U)</i></li> </ul> <p><i>Note: "modes of presentation" refers to "Allow multiple presentation modalities with integrated summary and user-directed access to details (Design Principle 4)."</i></p>



**Table B4. Forward Chaining Model: Illustrative Quotes**



DP	Application mode	Design behavior	Illustrative quote
7	Simple application	Projecting into solution space	<ul style="list-style-type: none"> <li><i>Simplify, support the integration of expert knowledge and knowledge based on evidence. Allow convergent and divergent discussions. So, the blog that you have for discussions, um, you have to think about, if there are other prior possibilities to communicate ... Otherwise there definitely has to be a moderator during the meeting who solves this organizationally and moderates.</i></li> <li><i>You could also ... in the literature ... you could do a match of what else there is in the literature ... based on key term that the experts have to provide. Where you do a literature search, automatized, in the relevant databases, research databases, that different universities, libraries offer. (Participant S)</i></li> </ul> <p><i>Note: “Simplify, support the integration of...” refers to “Facilitate/support integration of expert and evidence-based knowledge (Design Principle 7)”.</i></p>
4	Selective application	Projecting into solution space	<ul style="list-style-type: none"> <li><i>Provide different modes of presentation with integrated summary and user-driven access. Yeah, obviously that’s about if ... in the work sequence of these specialists or whoever wants to use the system, different questions are in the ... focus, then.</i></li> <li><i>Then the information has to be ... presented differently, other things are maybe more important if you, for example, if you want to comprehend a decision afterwards, than if you want to catch it in the contribution associated with it, maybe. (Participant V)</i></li> </ul> <p><i>Note: “Provide different modes of presentation...” refers to “Allow multiple presentation modalities with integrated summary and user-directed access to details (Design Principle 4)”.</i></p>

## Appendix C. Design Session Script

*[Administer Informed Consent. Ice breaker. Hand over a notepad and pencils to the participant.]*

### **Preparatory Tasks:**

Before we begin with the design task, I'd like to practice thinking aloud with you.

For instance, I get this question: how much is  $25 \times 46$ ?

I start thinking aloud to answer the question, while using the notepad to help me.

*[Demonstrate thinking aloud.]*

Now it is your turn. Do you have any questions before we start?

First task: A bottle of wine costs 20 EUR. The wine costs 15 EUR more than the bottle. How much does the bottle cost?

*[Provide the participant with feedback on her/his thinking aloud.]*

Second task: How much is  $126 \times 78$ ?

*[Provide the participant with feedback on her/his thinking aloud.]*

### **Instructions:**

You will receive a design task and you should complete the task using the available information. The information consists of a problem description and a set of design principles. Everything is prepared in German and English. You may choose in which language you'd like to work. There is no right or wrong answer. I am only interested in how you use this information, given your professional background in software design and development. Therefore, I ask you to solve the following design task and while you do so, try to say everything that goes through your mind. Try not to structure or explain what you do. Imagine you are alone in the room and talk to yourself. I will start the recorder.

*[Hand over the scenario and the design principles.] [Start the recorder.]*

From now on you are Marco. The scenario *[point to the piece of paper.]* is about your new project. Use the design principles *[point to the piece of paper.]* and say everything that goes through your mind.

Now you can start by reading aloud the scenario and the design principles.

## Appendix D. Data Coding and Analysis

### Segmentation and Coding

Due to the exploratory nature of this study, we aimed to achieve the closest textual representation (“true verbatim,” including pauses, stutters, murmurs, and other unintelligible or incomplete words) of participants’ spoken thoughts. The debrief after each design session was also transcribed verbatim, but not included in the reported verbal protocols. Together with field notes, the transcription was used during analysis. Analysis was carried out with the NVivo™ software across multiple iterative steps.

- **Preparatory Stage 1 (Segmentation):** The protocols were divided into segments. Each captured a unit of an intelligible idea with a single sentence or multiple ones. This decision was made as a fixed unit (such as a sentence or minute of protocol), which can contain multiple ideas or none at all. The speed and frequency of spoken-aloud thoughts cannot be uniform. This stage led to a labeled set of segments, e.g., P67 was the 67th segment of Participant P.
- **Preparatory Stage 2 (Identification of Design Principles):** Next, the first author read through all segments to identify the design principle(s), if any, that were considered within each segment. During this stage, conducted within a few days after the design session, she consulted the notes and listened to the audio recording, wherever necessary.
- **Coding Stage 1 (Coding for Designer Behaviors):** The first stage proceeded in the spirit of open coding (Strauss and Corbin, 1998) to identify what the designers were doing, i.e., *designer behaviors*. Although we were sensitized by our background, we strove to remain open and allowed the codes to emerge from the data. The segments were coded independently by two researchers. The resulting codes were assessed by the research team, resolving any disagreements with discussion to arrive at nine behaviors across three categories.
- **Coding Stage 2 (Coding for Modes of Applying Design Principles):** In the second stage, the segments were examined again to identify how the designers were using the design principles, i.e., *application modes of design principles*. During this stage, some context-specific rules were devised to ensure consistency. For example, under the category-selective application, each segment could be coded with only one of the subcategories (contextualization or extension or refinement). However, a segment could be coded with subcategories from different application modes (e.g., as “orchestration” under the category simple use as well as “extension” under the category-selective use). During this stage, the researcher remained open to revising the designer behaviors (from Stage 1). *See a more detailed description of coding for application modes, in particular one application mode—implicit application—in Appendix D2.*
- **Coding Stage 3 (Developing Categories):** During the third stage, we examined the codes and developed clusters. This allowed us to further distinguish the two dimensions: “design behaviors” and “application modes,” and develop more evocative labels for the categories within each dimension. Another round of coding followed, first for two designers, and then, for all participants. In all coding rounds, the coding was done in sequence, one participant after the next.
- **Analysis Stage (Developing Multiple Representations and Analyses):** We created several representations of the coding results, such as frequency counts, intersection among subcategories and categories, and comparisons within and across the two dimensions. These representations were useful to examine the data from different perspectives. As the analysis progressed, we visited vocabularies from prior work to identify patterns and develop interpretations.

### Coding for Modes of Applying Design Principles

The discovery of application modes (Coding Stage 2) was both time and effort-intensive, with multiple rounds of analyses and negotiation among researchers to resolve disagreements. Consider the simple application mode (*focus and orchestrate*). Here, we coded segments that clearly referred to a particular design principle. Consider the following example (from Section 4):

*One has to consider that together with principle three ... In principle three it's required that relevant knowledge that's demanded or requested, be made available, and the old cases also belong to that. They can be, so to say, linked, and then directly retrieved.* (Participant: P, application mode: Orchestration; Design Principles 3 and 6)

In the above example, participant P clearly mentioned “principle three.” But consider the next example (from Appendix A):

*A researcher has a special, a somewhat different focus, so he doesn't do just one treatment daily, but he um goes to conventions ... conducts experiments ... they are definitely two different types of users ... who play a role there. (Participant T, application mode: Focus; Design Principle 2, Segment T30)*

Here we see no direct reference to Design Principle 2, but the segment before has a clear reference:

*[Short pause] then principle 2, integration of the researcher into the context, utilization of meaningful prototypes. [Long pause] um [short pause] what does the researcher do? [Long pause] so apparently there is the factor researcher and the specialist. (Participant T; Segment T29)*

Because segment T30 is a clear continuation of segment T29. There is neither semantic contradiction nor any syntactical indication that shows otherwise. Since segment T29 clearly refers to Design Principle 2, we assume segment T30 does, too.

Similar rationale was followed to code for the selective application mode as well. The last application mode—*implicit application*—required a unique and demanding effort. Unlike simple or selective application (where we can observe direct indicators), the implicit application mode requires the use of other indicators.

Implicit application describes the mode when designers ignore, internalize, rationalize, or even willfully disregard a readily available and relevant design principle. To infer these, we relied on the signals in the true verbatim transcription including paralanguage (e.g., stutter, murmurs, and pause), and listening to the recording to look for prosody (e.g., intonation, rhythm, and emphasis). Consider the following example (from Appendix A):

*But it is still difficult. So, you would definitely have to, well basically you would have to talk to the same doctors regarding all these questions, how, how they... exactly, well, how exactly you can support that. (Participant V; application mode: Willful disregard; Design Principle 2, Segment V23)*

When reading only the above quote, we cannot infer anything related to design principle 2. But in segment V19 participant V clearly tried to make sense of Design Principle 2:

*Facilitate, support the integration of expert knowledge and knowledge based on evidence. [short pause] What is knowledge based on evidence? [long pause] That's more research-, from the research. (Participant V; Segment V19)*

Fast forward to segment V22, we find participant V arguing against what is written in the design principle:

*The way I see it, there aren't [short pause] representative from the science are not necessarily there. Well, of course doctors are probably [short pause] more or less up to date when it comes to technology. (Participant V; Segment V22)*

By now we understand that segment V23 is a continuation of segment V22, where participant V started to willfully disregard design principle 2. Then we check what happens next:

*Create a per-, persistent group da-, data storage. That's about, yes, how is [short pause] how are the data that are collected stored there, how, how is the access to it administered, how do they make sure, well, now that's a little bit from groups, but of course this goes much further, but will be a very important question in the system. Right? (Participant V; Segment V24)*

Segment V24 clearly refers to Design Principle 3, marking the beginning of another chunk of thoughts (or design work). We also notice no semantic association between segment V23 and segment V24. Since segment V23 is more related to segment V22 than it is to segment V23, we can infer that it is about design principle 2.

To sum up, coding for application modes was challenging—in particular, for the implicit application mode. We needed to consider multiple segments (not a segment in isolation) and develop an interpretation of the segment in question in relation to other segments before or after it.

## About the Authors

**Leona Chandra Kruse** is an assistant professor of information systems and innovation at the University of Liechtenstein. She is an associate editor of *European Journal of Information Systems*. Leona studies the bright side and the dark side of digital transformation as well as the known and the unknown in designing for digital transformation. Her works have been published in the *Journal of the Association for Information Systems*, *European Journal of Information Systems*, and *Communications of the Association for Information Systems*.

**Sandeep Puro** is Trustee Professor and chair of the Information and Process Management Group, associate director of the Hoffman Center for Business Ethics at Bentley University, and visiting professor at Agder University. His research focuses on the design of technologies for social good and the sciences of design. His work has been published across disciplines in journals such as *MIS Quarterly*, *Information Systems Research*, *Journal of the Association for Information Systems*, *Journal of Management Information Systems*, *ACM Computing Surveys*, *ACM Transactions on Management Information Systems*, *Communications of the ACM*, multiple *IEEE Transactions*, *Computer-Supported Cooperative Work*, *Journal of Medical Internet Research*, *International Journal of Public Health*, *Journal of the American Medical Informatics Association*, and several others. He has served on the editorial boards of several leading IS journals. His research has been funded by the National Science Foundation, industry consortia, private foundations, and private industry.

**Stefan Seidel** is a professor and chair of Information Systems and Innovation at the Institute of Information Systems at the University of Liechtenstein and an honorary professor of business information systems at the National University of Ireland, Galway. His research focuses on digital innovation, digital transformation, and artificial intelligence in organizations and society. Stefan's work has been published in leading journals, including *MIS Quarterly*, *Information Systems Research*, *Journal of Management Information Systems*, *Journal of the Association for Information Systems*, *Journal of Information Technology*, *European Journal of Information Systems*, *Communications of the ACM*, *IEEE Computer*, and several others. He is an associate editor for *MIS Quarterly*.

Copyright © 2022 by the Association for Information Systems. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation on the first page. Copyright for components of this work owned by others than the Association for Information Systems must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or fee. Request permission to publish from: AIS Administrative Office, P.O. Box 2712 Atlanta, GA, 30301-2712 Attn: Reprints, or via email from [publications@aisnet.org](mailto:publications@aisnet.org).