Co-design Theory Adoptability

How Organizational Change is Co-created by Design Theorists and Theory Adopters

Pries-Heje, Jan; Baskervillle, Richard

Published in:
Co-creating for Context in the Transfer and Diffusion of IT - IFIP WG 8.6 International Working Conference on Transfer and Diffusion of IT, TDIT 2022, Proceedings

DOI:
10.1007/978-3-031-17968-6_2

Publication date:
2022

Document Version
Peer reviewed version

Citation for published version (APA):
Pries-Heje, J., & Baskervillle, R. (2022). Co-design Theory Adoptability: How Organizational Change is Co-created by Design Theorists and Theory Adopters. In A. Elbanna, S. McLoughlin, Y. K. Dwivedi, B. Donnellan, & D. Wastell (Eds.), *Co-creating for Context in the Transfer and Diffusion of IT - IFIP WG 8.6 International Working Conference on Transfer and Diffusion of IT, TDIT 2022, Proceedings* (pp. 24-39). Springer. https://doi.org/10.1007/978-3-031-17968-6_2

Co-Design Theory Adoptability: How Organizational Change Is Co-Created by Design Theorists and Theory Adopters

Jan Pries-Heje[1][0000-0003-2219-9332] and Richard Baskervillle [0000-0003-2860-5215]

[1] Roskilde University, Denmark
[2] Georgia State University, USA
janph@ruc.dk

Abstract. Descriptive theories tell us how to understand the world better and prescriptive theories tell us how to make the world better. Design theories are of the latter kind but usually involves a product or process artifact to create changes as well. Despite a focus on utility, there is little evidence about how design theories become adopted after their development. We know too little about how design theories are applied, adopted, adapted, and emerge and change over time. We present a study of an organizational change design theory that was adopted in 108 different organizations. The results indicate that the theory was adopted As-Is in 25% of the cases. But it was adapted in various ways for use in the remaining organizations. Our analysis of these cases provides a typology of eight categories of adoptability. Among these, organizations most commonly adapted the recommended organizational strategy. As a result of these and other findings we show how design theory adoption is a continuous co-creation process between design theorists and design adopters.

Keywords: Design Theory, Design Logic, Organizational Change, IT Adoption.

1 Introduction

In this paper we build forward from concepts found in design science research (DSR) [1]. DSR is a relatively new interventionalist research paradigm with much in common with action research [2], clinical field work [3], etc. Such research approaches seek to learn by practicing and develop knowledge by solving practical problems. In DSR, interventions usually center designing, building and evaluating an artifact situated in a problem setting. Like other interventionalist research, DSR interventions are driven by theory. Theory for DSR often takes specific forms: *design theory* entails the relationships between generalized requirements and generalized artifact designs [4]; *design principles* express heuristics for designing a class of artifacts [5]. Joan van Aken proposed the use of technological rules as a means of expressing the prescriptive outcomes of management theory, i.e., collectively representing management design theories. Technological rules were first described in Mario Bunge's [6, 7] works on philosophy of science; they are a means of contrasting scientific knowledge from technological knowledge. Technological rules entail three elements: (1) prescribed actions based on (2) desired outcomes and (3) contingencies. We can think of these rules as technological propositions that proceed from a scientific theory. Technological rules can guide practitioners (such as engineers, accountants, physicians, and project managers) who may not know the theory but do know common practices by which outcomes are known to follow actions when taken under known contingencies.

Adoptability is an ability to start-to-use (i.e., adopt) some new artifact, be it a method, a technology, a framework, etc. Innovation lies not only in the new artifact, but also in the act of starting-to-use. Hence adoptability is created by adaptability. The ability of a design theory to apply persistently across a class of problems and a class of solutions (i.e., artifacts) depends on its accommodation for adoptability of these artifacts. The artifacts must adapt to a range of similar problems in a range of different contexts. Indeed, the change involved in such artifacts may quickly amount to a transformation. The contemporary demand for rapid and transformational change in technical artifacts means that adoptability not only applies to the artifact, but also to the design theory itself. The notion of *design theory adoptability* regards a characteristic liability or tendency for the design theory itself to change in order for it to guide the design of artifacts that adapt to a transformational change in the range of problems and contexts.

With technological rules as a frame, we see there are eight kinds of ways that practitioners adapt technical knowledge. The purpose of this paper is to illustrate how this frame can reveal patterns in adaptation of technical knowledge in practice.

In this paper we report our research into the adoptability of design theory in practical settings. Our choice of design theory to study was an organizational change nexus [8] and we obtained access to 108 documented cases that applied a design theory for choosing organizational change strategy. The organizational change nexus design theory was based on technological rules [9]. We analyzed the achievement of adoptability through the use of a *heuristic operator*. This heuristic operator encodes adoptability into the design theory by adapting the logic present in the design principles.

The paper is organized as follows. First, we discuss design logic and technological rules. In section 3 we present a framework we derived to analyze adoptability. Then in section 4 we present our multi-case study research method, and then in section 5 we present our analysis of adoptability and answer the research question. Finally, we discuss and conclude the paper in section 6 and 7.

2 Design Logic: Analytical versus heuristic technological rules

Van Aken sought to improve the practical usefulness of the corpus of management research: “academic management research has a serious utilization problem” [9, p. 219]. Organizational research is too often descriptive and historical. There is little direct usefulness of this kind of reflective study, especially for managers confronting newer, more current problems. For management research to have a greater impact on management, it must become more prescriptive and more design-oriented.

A design theory (or a set of design principles) is a generalized or abstract form of design as a design for a class of artifacts [4, 5, 10, 11]. While, some of the seminal work in science of design considered design logic [e.g., 12], the IS research community has instead focused on design theory. The works in design logic often concern electronic circuit logic design, or uses the term less formally to regard the general thinking behind designs for communication [e.g., rhetorical design logic or community design, 13, 14]. As information systems researchers, we use the term design logic for a set of formal principles of reasoning employed by designers for creating a design. Such logic sits in the realm of meta-design because it represents a formal set of principles that may be used across a general class of designs.

Simon's [12] influential definition of a science of design required an imperative logic, one involving not statements of the way things *are*, but statements of the way things *should become*. Imperative logics have serious flaws. For example, systems of imperative logic often have a different purpose: more often associated with ethics than design [15]. As a substitute, Simon introduced declarative logic within a search process. Fundamentally, the designer uses a declarative framework as a means of expression to represent the design solution. Then the design process has only to search through various values for the elements of the framework until a satisfactory design is discovered (satisficing). "Problem-solving systems and design procedures in the real world do not merely assemble problem solutions from components but must search for appropriate assemblies" [12, p. 124].

Consistently with Simon's design logic, van Aken [16] proposed management theory was a type of design theory that consists of "field-tested and grounded technological rules" [9]. This proposal includes two possible outputs: artifacts and interventions. A professional episode includes three kinds of designs. The object-design defines the artifact or intervention. The realization-design is the plan for implementing the artifact or intervention. The process-design is a plan for the design process itself. Designing is a means of developing prescriptive knowledge.

For van Aken, "A technological rule follows the logic of 'if you want to achieve **Y** in situation **Z**, then perform action **X**'. The central element of the rule is action **X**, a general solution concept for a type of field problem" [17, p. 23]. A formal expression of this rule would be,

$$(Z,Y) \rightarrow X$$

In this rule, **X** is the imperative "Do **X**". Imperative logic prescribes the behaviour of human actors. An example of this logic would be the technological rule, "If you want to achieve user acceptance of a new technology in a situation of user alienation, then adopt a co-creative design approach." This technological rule might be stated,

((USER ALIENATION), (USER ACCEPTANCE OF A NEW TECHNOLOGY)) → ADOPT A CO-CREATIVE DESIGN APPROACH

This rule may seem too tidy. Perhaps co-creation works best in collaborative organizational cultures. If so, **Z** would become "user alienation in a collaborative culture". Exactness in expressing all contingencies that affect the choice of action **X** can be done with multiple rules, each defining different contingencies and/or outcomes. (Such was Simon's [12] notion of searching through a forest of declarative rules.) The alternative to creating a forest of rules with compounded terms is to keep the rules simple but define the process of applying them as a process of adaptation [9].

A process of adapting the rules to specific situations involves establishing a generative function. Pawson and Tilley [18] relate adaptability to *generative causality*. Generative causality means that the exact meaning of the terms in the technological rule is a result of translations by managers, not any natural cause. The meaning is anchored to intention: an outcome sought by such interventions. Technological rules are a generative form of causal relationship.

If adoption of technological rules is a practice involving translation and interpretation, practice must be informed by the underlying theory. In our example above, the underlying theory is a fundamental socio-technical theory. This theory explains how co-creation approaches build acceptance through involvement and commitment among participating users. Such involvement and commitment emerge because participation and co-creation in design decisions empowers users. This participation gives them shared control over their futures. For this reason, technological rules (Bunge's technical knowledge) must be grounded in theory (Bunge's scientific knowledge) [17]. Grounding prevents technological rules from degenerating into an instrumental form of rules-of-thumb. "In engineering and in medicine, grounding of technological rules can be done with the laws of nature and other insights from the natural and the life sciences (as well as from insights developed by these design sciences themselves). In management, grounding can be done with insights from the social sciences" (p. 25). Regardless of how useful technological rules may be in practice, they are not theories in design science until they are grounded in a way that is acceptable to social science.

3 Heuristic operators and design logic

"However, many prescriptions in a design science are of a heuristic nature. They can rather be described as 'if you want to achieve Y in situation Z, then something like action X will help'. 'Something like action X', means that the prescription is to be used as a *design exemplar*. A design exemplar is a general prescription which has to be translated to the specific problem at hand; in solving that problem, one has to design a specific variant of that design exemplar." [9, p. 227]

Distinguishing between algorithmic and heuristic prescriptions opens the design process not only to analytic productions, but also generative productions. Generative productions add creativity and innovation to the process of implementation of technological rules. They incorporate problem-situated knowledge that allow the rules to succeed. Generative productions require the evaluation of "something like" to incorporate creative invention in adapting the prescriptive action X to the exact context at hand. Generative productions are required because, in practice, organizational change cannot always be reduced to an analytical X, Y, Z formula because every organization has its own unique characteristics and contingencies.

We represent this notion of "something like" action X using a heuristic operator (~) as part of the logical representation of the technological rule. The revised technological rule now represents "if you want to achieve Y in situation Z, then something like action X will help." Such a rule would be represented by,

$$(Z,Y) \rightarrow \sim X$$

This revised rule represents a design production that contains both analytical and generative elements. The analytical element arises from the core rule,

$$(Z,Y) \rightarrow X$$

The generative element arises in the open adoptability around the action to be taken. This rule allows the designer to invent an adaptation of action X that best fits the designer's situation. This adaptable action, ~X, allows the design to be partly generative. It now requires a different form of reasoning on the part of the designer in deciding how exactly the special form of action X should emerge. But it still retains the core logic of the creator of the design rule: the design theorist. So adoption of the design theory is a co-creation of both design theorist and design theory adopter.

Human organizations are deeply diverse. This diversity in the settings means that action ~X may be more often the norm than action X itself. For many technological rules, it is possible that the ideal setting for action X arises only rarely, making action ~X a necessity for most cases. An example of this logic is, "If you want to achieve user acceptance of a new technology in a situation of user alienation, then adopt something like a participative design approach". This technological rule might be stated,

((USER ALIENATION), (USER ACCEPTANCE OF A NEW TECHNOLOGY)) → ADOPT SOMETHING LIKE A CO-CREATIVE DESIGN APPROACH

It follows that these diverse human settings make it likely that situation Z is also an idealization that may occur rarely in its exact form. Again, the heuristic operator can be used to represent a adaptable technological rule. "If you want to achieve Y in a situation something like Z, then something like action X will help."

$$(\sim Z,Y) \rightarrow \sim X$$

In the absence of specific deductive logic to help disambiguate **~Z ,** the determination as to whether the situation at hand is in fact "something like" Z will itself be a generative production. The designer must imagine the relevant ways in which the two situations relate. Indeed, the technological rule can be made fully adaptable. "If you want to achieve something like Y in a situation something like Z, then something like action X will help."

$$(\sim Z,\sim Y) \rightarrow \sim X$$

Without specific deductive logic to help disambiguate **~Y ,** the determination as to whether the goals at hand are in fact "something like" Y is a generative production. The designer must imagine the relevant ways in which the two sets of goals are similar and the ways that they are different. The heuristic operators offer the designer the latitude for generative productions that are flexible enough to adapt the design logic to the particular setting. In addition to permitting generative productions in a design setting, heuristic operators introduce generality into design logic. For example, the core technological rule, will often arise as a kind of empirical *point solution*:

$$(Z,Y) \rightarrow X$$

A rule such as, "If you want to achieve user acceptance of a new technology in a situation of user alienation, then adopt a co-creative design approach" results from a field experience in which participative design provided a way to overcome user alienation. At that time of its origination, the design logic embodied in the rule provided a quite pointed, specific, practical solution to a quite pointed, specific, practical problem. This point solution is advanced as the general rule above. In this generalization, we translated the point logic (as used in the original setting) into the technological rule (the design logic). However, there are differences. In the point logic, the co-creative design was quite specific. For example, the point logic might have involved assigning users as well as developers to co-creative design teams, user specification review sessions, or prototyping together. In the design rule, these point solutions are expressed more generally as "co-creative design".

The heuristic operator further opens up the generality, suggesting that "something like co-creative design" should operate successfully. This enhanced adoptability enables the designer to generate alternatives to co-creative design that work better in the setting-at-hand. It effectively makes co-creative design an element of some unstated general class of solutions that will need to be re-conceptualized in the future.

Heuristic operators are found in genetic algorithms to incorporate problem-dependent knowledge in order to transform any infeasible solution into a feasible one [19]. Our heuristic operator can be described as a *modal operator*. Modal words and phrases are sentential adverbs. Sentential adverbs modify whole phrases or sentences. Examples of modal words include *necessarily* and *possibly*. In our case, the heuristic operator modifies a phrase like "want to achieve user acceptance of a new technology" to mean "want to *something like* achieve user acceptance of a new technology."

The logic around such modal operators is largely a linguistic topic. Modal logic provides the systematic expression of concepts that are represented in natural language by modal words and phrases. Modal logic operates as a meta-language that characterizes the logical, syntactical, and semantical properties of an object language [20].

Normally the modal expression in the object language is distinct from the associated operator in the meta-language. For example, we might specify a heuristic sign (say, ~) as a symbol in our object language that stands for heuristic. For our metalanguage, we would require a heuristic operator (say, □) as an expression of our metalanguage that stands for the operation that concatenates the heuristic sign with any other expression [20]. However, the scope of this paper is limited to explaining how this particular modal (heuristic) operates within a quite specific object language that is not ours to define [9]. This object language is limited to certain existing expressions of technological rules because of our focus on DSR. It is beyond the scope of this paper to specify a general object language for the universal application of the adoptability modal. Similarly, there is no necessity to develop a universal metalanguage. Simplicity commands that we conflate the notion of a heuristic sign in an object language with its related concatenation operation in a metalanguage, using the tilde as our heuristic operator (~) to imply both the notion *and* the attachment of that notion to a phrase or sentence.

In terms of modal scope, the heuristic operator is broader in scope and one of many possible parents of the narrower-scope modal *exactly*. The expression *something like participative design* will include the possibility of *exactly participative design*. The narrower scope of the modal *exactly* will have other parents of broader scope, such as *at least*, as in *at least a design conference* [21]. Similarly, expressions can be *modal free* (i.e., adoptability free technological rules). Modal free technological rules are those in which the heuristic operator (our modal sign) does not occur at all. Some expressions are *modally closed* (i.e., adoptability closed technological rules). Modally closed technological rules are those in which every occurrence of a phrase within the technological rules is an occurrence within the scope of the heuristic operator (our modal sign) [20].

3.1 Applying the Heuristic operator

We applied the heuristic operator initially to express our working theory about our expectations for common patterns of adoptability in organizational settings. This working theory involved two working propositions:

Working proposition 1: If the situation (Z) is unadaptable, designers are driven to more analytical mental productions, and this analysis will resolve the solution to the action (X) and the goal (Y) exactly as specified in the unadapted rule.

Working proposition 2: If the situation (~Z) is adaptable, designers will be driven to more generative mental productions and these will tend to adapt the action (~X) and/or the goal (~Y).

If these working propositions hold, common patterns of adoptability should cluster around four of the eight possible rule patterns:

$$(Z,Y) \rightarrow X$$

$$(\sim Z,\sim Y) \rightarrow \sim X$$

$$(\sim Z,\sim Y) \rightarrow X$$

$$(\sim Z,Y) \rightarrow \sim X$$

If the working propositions hold, the other patterns should be rare:

$$(Z,\sim Y) \rightarrow \sim X$$

$$(Z,\sim Y) \rightarrow X$$

$$(Z,Y) \rightarrow \sim X$$

$$(\sim Z,Y) \rightarrow X$$

4 Research Methodology

We examined the validity of the heuristic operator using a qualitative, multiple case study. Our approach involved a "clinical perspective in fieldwork" originally described by Edgar Schein [3]. The fieldwork involves interventions and the findings are validated by the resulting improvements. Our purpose in this fieldwork was to validate the use of adoptability logic to express and test theoretical propositions. Whether the outcome of the fieldwork confirms or disconfirms the propositions is less important than the validity and clarity of the logic used to express the propositions and the results (for our purposes). In the cases at hand, as the reader will see, the fieldwork disconfirms these propositions. But it is the logical clarity with which this result finds expression that satisfies our basic research goals. The ability to reformulate the propositions for further examination may indeed provide better evidence for the strength of the adoptability operation than an alternative result that simply confirmed the theory under test.

In this fieldwork, practicing information systems project managers explored the adoptability in their technological rule settings. The purpose of this fieldwork was to see whether the use of a heuristic operator as a logical modifier for technological rules would lead to insights into the use of technological rules in real project settings.

4.1 The organizational change nexus

The study of organizational change may be the most important management discipline. Academic and practitioner contributions to organizational change have been built on empirical work in a wide variety of organizations and from such different perspectives as psychology, sociology, and business. Examples of this work include descriptive accounts of change, normative models to guide the change process, theoretical models for understanding and analyzing change, typologies of different approaches to organizational change, and empirical studies of the success or failure of change. Most recently IT-related change has come into focus in studies of digital transformation [cf. 22]

The process by which an organization selects an approach to organizational change is often ad hoc or driven by habit, i.e. 'we did this the last time, let us do it again'. Different change approaches each has their advocates and adherents, and there is little comparative research for choosing among such approaches. To deal with this wicked problem Pries-Heje and Baskerville [8] derived an organizational change nexus which can be used to recommend one or more appropriate change approaches among ten prominent organizational change strategies.

The 10 change approaches can be said to follow technological rules in that a specific approach – action X - is recommended, based on a description of the current situation (Z) and the organizational change (Y) wished-for. Hence, the 10 change approaches from Pries-Heje and Baskerville [8] can be expressed as technological rules.

One of the 10 change strategies, for example, is called "commanding" which is well suited for situations (Z) where change is required fast and/or where organizational structures needs to be changed. Another is called "production oriented" which focusses on the flow of value in the organization, identification of bottlenecks, and removal of these bottlenecks (that is the change). Yet another is "business process reengineering" which aims for a total redesign of the organization. The use of this last change strategy requires a solid crisis like continuing major economic losses. This strategy involves the carte blanche designing of the optimal organization that centers on providing value to the customer. The change regards implementing this optimal organization. Two examples of the technological rules are shown in Figure 1.

If you want to initiate organizational change **in a situation** where you:

Believe that formal structures needs change

Where change is needed fast

Then choose a *Commanding* approach where change is driven and dictated by (top) management; one where management takes on the roles as owner, sponsor and change agents.

If you want to initiate organizational change **in a situation** where:

You believe that target group is very diverse and has large individual differences

The target group are experts

Then choose an *Optionality* approach where change is driven by the motivation and need of the individual; it is to a large degree optionality.

Fig. 1. Example technological rules from the organizational change nexus.

The organizational change nexus was implemented as:

- An IT artifact that is using design theories to score "fit" for each change strategy
- Questionnaire about the contingencies of the situation in the organization and the wished-for change
- Guidelines for engaging with practice

Together these parts can guide managers in evaluating and choosing which of the ten change strategies that would be most appropriate in an actual organizational setting for a defined organizational change. The IT artifact calculates a *fit* on a scale going up to 100% fit based on contingencies of the situation today and the desired change.

The organizational change nexus was articulated as follows: "… to improve the ability for organizational change managers to rationally select the most appropriate change strategies" [23] and they conclude "We developed a framework that binds together ten well-known organizational change strategies into a prescriptive recommendation for a cohesive and suitable change strategy for a particular organization's unique situation. The change strategies to be prescribed develop from a list-of-fit that indicates the relative suitability of each of the ten strategies to the organization's vision and setting" [23]. While the originators only evaluated the nexus in four companies in the financial sector, they predicted that it would be useful in other IT-organizations and –projects.

That prediction was confirmed in November 2014 when the organizational change nexus was incorporated as an prominent part of the ISO/IEC standard 33014 on software process improvement [24]. This standard has three levels: strategic, tactical and operational. The ten change strategies are included at the strategic level – and mentioned by name – to be used once an organization has identified its business goals and the scope of the organizational change. At this point the nexus can be used for identifying the *overall change strategy*. Hence the inclusion of this meta-level design theory as part of a recognized international standard indicates a very successful use of the nexus.

4.2 Multi Case Study fieldwork

We have collected data from 108 managers using the organizational change nexus up until March 2022 where 16 mainly project managers applied the nexus. An example is a case from 2022 where the overall change desired is less food-waste in a supermarket chain, and the concrete design artifact is a digital tool called How-Less-Waste that a major part of supermarket chain employees are to adopt. Now the question is how? The owner of the change, a program manager responsible for implementing the change in the 100 of supermarket stores, a change agent, and a representative from the target group of 1000s of employees meet and fill out the nexus questionnaire [8, Figure 3, p. 741]. The manager using the nexus then facilitates a discussion until the group agrees on what the situation is for this organizational change. The agreement is fed into a spreadsheet that uses the 10 design theories to calculate the fit for each of 10 change strategies. In the concrete example the three highest scoring change approaches are exploration (50% fit), production-driven and lean (43.75% fit), and a business process reengineering (BPR) change strategy (40%). Based on these fits the manager now decides to implement the organizational change aiming for less food-waste by first exploring ways to achieve less waste, and then implementing in a big production-oriented scale. However, the manager decides to ignore the third high scoring change strategy BPR. Thus, our perception of this change is that Y is clear and described well by the nexus. Z is clear and the participants can reach agreement on all 32 contingency factors that together describes the situation. But X is adapted in that the recommendation of combining the three change approaches with the best fit is not followed.

The way the 108 managers learned to use the nexus in their home organizations in a 4-semester executive master in project management and organizational change. The second semester focused on organizational change and the organizational change nexus was a mandatory part of the curriculum. Over the years we have collected data from more than 300 applications by the executive master students. For this paper we chose 108 change projects emphasizing IT-related changes such as the example above. All 108 participants chosen were managers at project level or higher in their own organizations. The project managers were asked to do the following:

1. Choose and describe an organizational change in the context of their own organization
2. Indicate agree or disagree to nexus statements. Preferably in a process where all relevant managers (relevant to the change that is) participates
3. Use agreement and disagreement scoring to calculate fit for each of the 10 strategies
4. Plan the organizational change based on the best fitting change strategies
5. Write a report documenting the plan, to what extent the plan was implemented, and write a critical review of the use of the nexus

On average the reports from the 108 change projects chosen were 5 pages long plus an appendix with the scoring (agreements and disagreements).

5 Analysis of the 108 cases

The reports from the 108 organizational change applications were analyzed using a framework looking as shown in Table 1 with an example of the coding of three organizations (see also the example case on less food-waste in a supermarket chain above).

Table 1. Example coding of 3 out of 108 cases

Case	2. You want to achieve Y	3. … in situation Z,	4… then perform action X	5. Re-articulation	6. Perception after use – we could use
A.	Y clear; Better performance of Revenue Information by more effective IT use	Z clear;	X clear; Combining 2-3 strategies. No discussion on how to combine	The organization was changed. Unfortunately too fast (hinting at the commanding strategy). It would have been smart to use the specialist-driven strategy more. The organization was filled with specialists	It makes sense to combine these 2-3 strategies.
B.	Y ambiguous; Unclear. Not described more precisely than "development" which can be anything	Z clear; Teachers at High School teaching different subjects – now also using virtual meetings	X clear; Combining specialist, learning and optionality	"It is also interesting to take a closer look at what strategies does NOT suit the organization, especially if these strategies have been tried without luck"	"The model operationalizes behavior and strategy in relation to each other which makes it really applicable" "A very little investment in time gives management a targeted way to reflect on change"
C.	Y clear; Reorganization causes by digital transformation leading to fewer	Z ambiguous; Top management still has not told us why	X ambiguous; Optionality (scoring highest) cannot be used for this type	"The tool does not take into account feelings and individual preferences"	"The model can give an indication … but the model does not take into account local contingencies"

people in the organization	the reorganization and what values it is expected to give	of change (reorganization)	

The coding is based on Joan van Aken [17, 25] and his notion of "technological rules" presented earlier. The organizational nexus [8] can be seen as 10 design theories that each aim for a desired change "Y" with some contingent characteristics, i.e. whether the change is tangible or not; The situation today "Z" also having some characteristics i.e. how successful we have been in the past wit hour change projects; And the ten overall change strategies "X".

In Table 1 the design theory elements "Y", "Z" and "X" are coded in column 2-4. At the top of each field it is stated whether any adaptation or translation was taking place. In Case A, for example, no adaptation of the nexus took place. Whereas in case B the scope of the change wished-for was unclear; it was just a direction "development" using virtual meeting apps. And in case C the situation was unclear as management had not communicated why the reorganization (= the change) was necessary, as well as the recommended change strategy – Optionality – that could not be used for a directed reorganization. Thus cases A, B and C are respectively examples of no adaptation at all, very little adaptation, and quite a lot of adaptation.

In Table 1, Column 5 called "Re-articulation" the eventual considerations when making a tactical plan are given, and in column 6 called "Perception after use" we have coded any critical reflection the project manager may have had when using the nexus in their own practice.

There are eight potential codings of the 108 cases:

1. (Z,Y) → X (everything follows nexus)
2. (~Z,Y)→ X (the change wished-for adapted to fit nexus)
3. (Z,~Y) → X (description of situation adapted in relation to nexus)
4. (Z,Y) → ~X (the change strategy adapted e.g. by combining several or taking #2)
5. (~Z,~Y) → X (change AND situation description adapted)
6. (Z,~Y) → ~X (situation AND change strategy adapted)
7. (~Z,Y) → ~X (change wished-for and change strategy adapted)
8. (~Z,~Y) → ~X (everything adapted to local conditions)

The waste-less-food example above is here a type 4 in that neither Z or Y changes, but one of the recommended strategies are left out – hence adapting X.

In Table 2 you can see how many of each type we found in our coding

Table 2. Instances of each of 8 types of adoptability found in coding of 108 cases

TYPE	No. of Instances
Type 1: (Z,Y) → X (everything follows nexus)	27
Type 2: (~Z,Y)→ X (the change desired adapted to fit nexus)	14

Type 3: (Z,~Y) → X (description of situation adapted in relation to nexus)	12
Type 4: (Z,Y) → ~X (the change strategy adapted e.g. by combining several or taking #2)	24
Type 5: (~Z,~Y) → X (change AND situation description adapted)	6
Type 6: (Z,~Y) → ~X (situation AND change strategy adapted)	12
Type 7: (~Z,Y) → ~X (change desired and change strategy adapted)	5
Type 8: (~Z,~Y) → ~X (everything adapted to local conditions)	7

The most common coding in table 2 was no. 1, which is using the nexus exactly as intended. The least common coding was no. 5, which adapted the change AND the situation, and no. 7, which adapted the change and the strategy recommended. There were respectively 6 and 5 instances in those. Further, adaptations of a single element – types 2. to 4. – were more common than changing two elements (types 5.-7.). In general it was much more common – about 75% – to adapt some element or elements (2.-8.) than not (no. 1.).

6 Discussion

In Figure 2 we have summarized our work presented until now. At the top of the figure you find the Organizational Change Nexus consisting of three parts. The three parts are the 10 design theories formulated as technological rules (X), a questionnaire posing statements about the context (Y and Z) and some guidelines for how to engage the case practice; an IT artifact that based on the identified context calculates the "fit" for each of the ten design theories. In the middle you have the 108 cases in our fieldwork; in each case a number of top-, middle- and project managers were involved in scoring the fit for the ten change strategies. Then at the bottom, we have the results of our analysis and coding; 25% did no adaption at all, whereas 69% adapted something; and 6% adapted everything – organizational context, change context and change strategy.

As a whole the fieldwork offers results at two levels. The first level regards the outcome in terms of the propositions that were disconfirmed by the data. The second level regards the validity and usefulness of the heuristic operator when used to modify the predicate logic of technological design rules.

6.1 Discussion of the fieldwork

While the confirmation/disconfirmation was not our primary objective, we must not overlook a brief discussion of these results in terms of their implications for future research. Our fieldwork does not support the working propositions and consequently does not support the working theory. Actually, the most common patterns appeared to be those that we expected to be most rare. The most common pattern involved no adaptability (Type 1 in Table 2). This situation is a common pattern in our data, similar to one in which the action to be taken is adaptable, (Z,Y) → ~X.

However, the data suggest that this logic may need to be expanded. The other most-common pattern involved adoptability in the situation (Z). We speculate that the presence of adoptability in Z drives designers to make an analytical treatment of Y and X, (~Z,Y) → X.

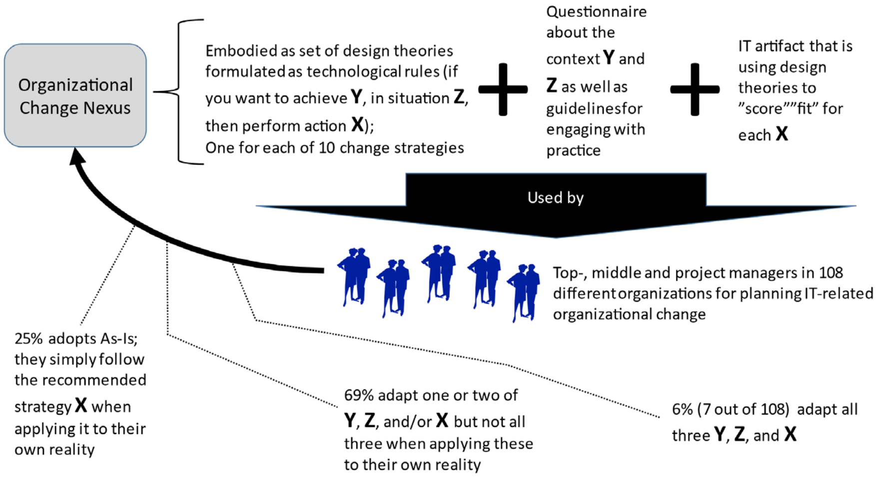

Fig. 2. Overview of our fieldwork

6.2 Discussion of the heuristic operator

While our qualitative fieldwork is not an ideal way to prove or disprove theories, it does strongly validate the usefulness of the heuristic operator in design logic. Whether the outcome of the fieldwork confirms or disconfirms the propositions is less important than the validity and clarity of the logic used to express the propositions and the results.

The fieldwork indicates that a heuristic operator permits a clearer understanding and representation of the expected and unexpected variation between the elements of a design logic expression and an instance of the application of this design logic. Our work offers an original contribution in its development of a heuristic operator. Simon's [12] original formulation of the Science of Design proposed the design logic as an expression of a scientific design. He mentions design theory only in passing. The availability of the heuristic operator is an improvement in expressions of design logic that make such logic both more valid and more practical for use in empirical DSR fieldwork.

Our work also extends van Aken's [9] application of Bunge's [6, 7] technological rules as an expression of management design logic. By adding heuristic operators to van Aken's logic, we believe these technological rules provide a more realistic and practical expression of their prescriptive value.

We offer an original contribution by providing means for expressing points within the design process at which a designer must work creatively to make prescriptive design theories and prescriptive design logic fit a design setting at hand. The heuristic operator presents a declaration of the need for such generative productions on the part of those applying a design theory. Heretofore, explicit points of such generative opportunities have had no formal means of expression in design logic or design theories.

The heuristic operator in our work is tightly tied to design logic in the form of technological rules. Future research is needed to explore how adoptability can be more explicitly expressed in other forms of prescriptive design theories in DSR. Our work was successful in critically validating the heuristic operator. It also provided a means for considering design theory propositions. However, we need more research in how adoptability properties, whether of artifacts, theories, or logic, can be evaluated as an outcome of DSR.

Furthermore, in relation to diffusion and adoption of theories in practice the adoptability and the heuristic operator can be seen as a kind of co-creation of adoption between design theorists and practitioners. We know that 3 out of 4 adoptions a given theory – in our case a design theory on how to plan your organizational change – more or less. We haven't seen other papers being that specific about what is being changed – adopted – in theories when "translated" to practice.

7 Conclusion

Our clinical fieldwork explains how the adoption of a design theory on organizational change occurred for 108 different organizations. It is quite unique to have data on how a theory is applied or translated in practice. Through the lens of technological rules and heuristic operators, different kinds of adaptability appear. In our illustrative cases, only in 25% of the more than 100 cases reported here the nexus theory was applied as-is. In 75% of the cases the nexus theory was adapted in one of seven different ways, the most popular adaptation was to accept the description of the situation today (Z), the desired change (Y) but change the action, what one should be doing (X).

This analysis of our fieldwork data represents a considerable improvement in expressions of adaptability in design logic that will also make it more practical for use in future empirical DSR fieldwork. Future DSR research can use this logic to examine explicit adoptability rules for new design theories.

References

1. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science In Information Systems Research. MIS Quarterly 28, 75-105 (2004)
2. Baskerville, R., Wood-Harper, A.T.: Diversity in Information Systems Action Research Methods. European Journal of Information Systems 7, 90-107 (1998)
3. Schein, E.: The Clinical Perspective in Fieldwork. Sage, Newbury Park, Calf (1987)
4. Baskerville, R., Pries-Heje, J.: Explanatory Design Theory. Business & Information Systems Engineering 2, 271-282 (2010)
5. Markus, M.L., Majchrzak, A., Gasser, L.: A design theory for systems that support emergent knowledge processes. MIS Quarterly 26, 179-212 (2002)
6. Bunge, M.: Scientific Research I: The Search for System. Springer-Verlag, New York (1967)

7. Bunge, M.: Scientific Research II: The Search for Truth. Springer-Verlag, New York (1967)
8. Pries-Heje, J., Baskerville, R.: The design theory nexus. MIS Quarterly 32, 731-755 (2008)
9. van Aken, J.E.: Management Research Based on the Paradigm of the Design Sciences: The Quest for Field-Tested and Grounded Technological Rules. The Journal of Management Studies 41, 219-246 (2004)
10. Gregor, S., Jones, D.: The Anatomy of a Design Theory. Journal of the Association for Information Systems 8, 312-335 (2007)
11. Walls, J.G., Widmeyer, G.R., El Sawy, O.A.: Building an information system design theory for vigilant EIS. Information Systems Research 3, 36-59 (1992)
12. Simon, H.A.: The Sciences of the Artificial. MIT Press, Cambridge, Mass. (1996)
13. Bryan, S.P.: Cognitive complexity, transformational leadership, and organizational outcomes. Louisiana State University and Agricultural & Mechanical College, United States -- Louisiana (2002)
14. Fritzen, S.A.: Can the Design of Community-Driven Development Reduce the Risk of Elite Capture? Evidence from Indonesia. World Development 35, 1359-1375 (2007)
15. Gensler, H.J.: Formal ethics. Cambridge Univ Press (1996)
16. van Aken, J.E.: Valid knowledge for the professional design of large and complex design processes. Design Studies 26, 379-404 (2005)
17. van Aken, J.E.: Management Research as a Design Science: Articulating the Research Products of Mode 2 Knowledge Production in Management. British Journal of Management 16, 19-36 (2005)
18. Pawson, R., Tilley, N.: Realistic Evaluation. Sage, London (1997)
19. Chu, P.C., Beasley, J.E.: A Genetic Algorithm for the Multidimensional Knapsack Problem. Journal of Heuristics 4, 63-86 (1998)
20. Cocchiarella, N.B., Freund, M.A.: Modal Logic : An Introduction to its Syntax and Semantics. Oxford University Press, New York (2008)
21. Breakstone, M.Y., Cremers, A., Fox, D., Hackl, M.: On the analysis of scope ambiguities in comparative constructions: Converging evidence from real-time sentence processing and offline data. Semantics and Linguistic Theory, vol. 21, pp. 712-731 (2011)
22. Vial, G.: Understanding digital transformation: A review and a research agenda. Managing Digital Transformation 13-66 (2021)
23. Pries-Heje, J., Vinter, O.: A framework for selecting change strategies in IT organizations. In: International Conference on Product Focused Software Process Improvement, pp. 408-414. Springer, (Year)
24. ISO/IEC_33001: Information technology -- Process assessment -- Concepts and terminology. Geneva, Switzerland. (the first of the ISO/IEC 330xx serie of standards). (2015)
25. van Aken, J.E.: The Nature of Organizing Design: both like and unlike material object design. Working Paper 06.13, Eindhoven Centre for Innovation Studies, Department of Technology Management, Technische Universiteit Eindhoven (2006)