Association for Information Systems
AIS Electronic Library (AISeL)

ECIS 2009 Proceedings

European Conference on Information Systems (ECIS)

2009

Justifying design decisions with theory-based design principles

Michael Schermann
Santa Clara University, michael.schermann@in.tum.de

Gehlert Andreas
University of Duisburg-Essen, andreas.gehlert@sse.uni-due.de

Helmut Krcmar
Technical University Munich, krcmar@in.tum.de

Klaus Pohl
University of Duisburg-Essen, klaus.pohl@sse.uni-due.de

Follow this and additional works at: http://aisel.aisnet.org/ecis2009

Recommended Citation
Schermann, Michael; Andreas, Gehlert; Krcmar, Helmut; and Pohl, Klaus, "Justifying design decisions with theory-based design principles" (2009). *ECIS 2009 Proceedings*. 198.
http://aisel.aisnet.org/ecis2009/198

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2009 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

JUSTIFYING DESIGN DECISIONS WITH THEORY-BASED DESIGN PRINCIPLES

Michael Schermann, Chair for Information Systems, Technische Universität München, Boltzmannstrasse 3, 85748 Garching, Germany, michael.schermann@in.tum.de

Andreas Gehlert, Software Systems Engineering, University of Duisburg-Essen, Schützenbahn 70, 45117 Essen, Germany, andreas.gehlert@sse.uni-due.de

Helmut Krcmar, Chair for Information Systems, Technische Universität München, Boltzmannstrasse 3, 85748 Garching, krcmar@in.tum.de

Klaus Pohl, Software Systems Engineering, University of Duisburg-Essen, Schützenbahn 70, 45117 Essen, Germany, klaus.pohl@sse.uni-due.de

Abstract

Although the role of theories in design research is recognized, we show that little attention has been paid on how to use theories when designing new artifacts. We introduce design principles as a new methodological approach to address this problem. Design principles extend the notion of design rationales that document how a design decision emerged. We extend the concept of design rationales by using theoretical hypotheses to support or object to design decisions. At the example of developing a new conceptual modeling grammar we demonstrate two main benefits of using design principles. First, the link between theory and design decision enables the design researcher to reason about the resulting behavior of the IT artifact prior to instantiation. Second, design principles allow deducing empirically testable hypotheses to foster the rigorous evaluation of IT artifacts.

Keywords: design principles, design rationale, design theories, theories, design research

1 INTRODUCTION

Often, design research results are criticized as being subjective and anecdotal in nature. Design researchers need to argue, how the IT artifact relates to hypotheses about the nature and characteristics of the artifact and its proposed benefits. Thus, theories are the cornerstone of improving the inter-subjectivity of design artifacts (Walls et al., 1992). Figure 1 shows a bilateral relation between theory and design (Nunamaker and Chen, 1991). According to relation ❶ "Contribution" in Figure 1, theory may follow design. Here, design research contributes to the IS knowledge base by initiating theory-building based on existing design artifacts. Walls et al. (1992) and others introduced the concept of design theories to specify the theoretical contribution of design research to the IS knowledge base (e.g. Markus et al., 2002; Walls et al., 1992).

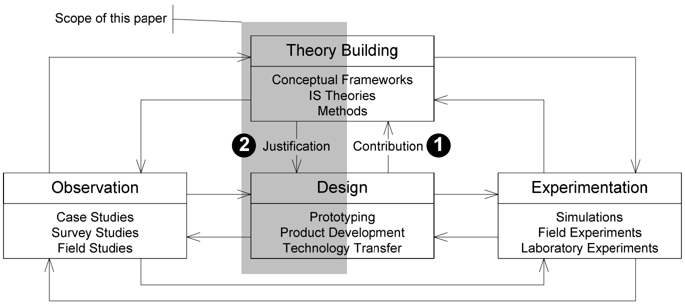

Figure 1: The elements of IS design theories based on Nunamaker and Chen (1991) and March and Smith (1995)

Theories can also be used to justify design decisions and subsequently anticipate the behavior of the IT artifact (relation ❷ "Justification" in Figure 1). This relation is based on the insight that any design artifact cannot work if it does not comply with the laws embodied in the underlying theory (Walls et al., 1992). In addition, the artifact may be significantly improved if the underlying theory is explicit (van Aken, 2004, p. 228).

Although this second relation is widely recognized on a general level (Gregor and Jones, 2007; Venable, 2006; Walls et al., 1992), we show that little attention has been paid to the methodological guidance on how to use theories when designing artifacts. This paper aims to fill this gap by answering the following research question: How can theoretical knowledge be used in the design process? To answer this question, we argue that the iterative nature of design research requires a particular focus on the specific decisions made during the design research process and the underlying arguments which finally led to the design artifact (Hevner et al., 2004). We apply and extend the concept of design rationale to structure, justify, and document the decisions made in the iterations of the design research process.

Design rationales can be used to justify design decisions during the construction of the artifact by documenting design alternatives and the reasons for selecting or rejecting them (Regli et al., 2000). Because of the transparency of the decision making process, design rationales are valuable sources to stimulate theory-building, i.e., they facilitate design researchers in specifying the first relation between design and theory (see ❶ in Figure 1). However, to support the theoretical justification of design decisions (see ❷ in Figure 1), we extend the concept of design rationale by introducing theory-based arguments. In doing so, the design decision can be traced back to empirically testable hypotheses, which enable the design researcher to reason about the effects of the IT artifact as well as to reason about the artifact's possible side-effects (Weber, 1987). We argue that such theory-based design decisions form a new type of IS design knowledge, which we call design principles. A *design principle* is a coherent claim of utility (Markus et al., 2002) and consists of a specific design problem, a solution to the problem and the trace path of the discussion leading from theory to the solution. The link between theory and design decision enables design researchers to reason about the resulting behavior of the IT artifact prior to instantiation. Furthermore, design principles allow deducing empirically testable hypotheses to foster the rigorous evaluation of IT artifacts.

The remainder of the paper is organized as follows: In the next section, we review existing work on the role of theories in design research. We show that little attention has been paid on how to incorporate theories when designing new artifacts. Next, we analyze the notion of theory in IS research and the concept of design rationales as the basic elements of our approach. Then, we show how to develop design principles by applying theoretical hypotheses as justificatory knowledge in design rationales. At the example of developing a new conceptual modeling grammar we demonstrate the utility of our approach. Overall, we contribute to the methodological foundations of IS design research by explaining how theories can be used to justify design decisions. The paper closes with a discussion of the implication of this approach and an outline for further research.

This paper is of exploratory and conceptual nature. Hence, we provide argumentative support when answering our research questions. However, we base our arguments upon available empirical and conceptual research results as well as on the results of a previous paper (Gehlert et al., 2009). This paper extends the results of the previous one by introducing the notion of design principles and by an in-depth analysis of the relation between these design principles and theories.

2 THEORIZING IN DESIGN RESEARCH

In this section, we summarize the current debate on the role of theories in IS design research. The important role of theories in design research has been addressed by a number of authors (Gregor and Jones, 2007; Markus et al., 2002; Schermann et al., 2007; Verschuren and Hartog, 2005; Walls et al., 1992). Gregor and Jones (2007) distinguish two roles of theories in IS design research: First, theories can be the output of design research: “The distinguishing attribute of theories for design and action is that they focus on ‘how to do something’. They give explicit prescriptions on how to design and develop an artifact, whether it is a technological product or a managerial intervention” (Gregor and Jones, 2007, p. 313). Second, theories provide the foundation of justifying decisions during the design process and the subsequent features of the artifact: “[..] we argue that these theories are a linking mechanism for a number, or all, of the other aspects of the design theory. […] Theories might come from natural science, social science […], other design theories, practitioner-in-use theories […], or evidence-based justification such as seen in medical research and action research.” (Gregor and Jones, 2007, p. 327)

Hevner et al. (2004) propose guidelines to determine the contribution of design research results to the IS knowledge base. Their guideline 3 requires that the “[…] utility, quality, and efficacy of [the] design artifact must be rigorously demonstrated” (Hevner et al., 2004, p. 83). In the line with this argument, Walls et al. (1992) argue, that design theories should propose empirically testable hypotheses to provide means for testing the utility of the design theory. Hence, the role of theories in IS design research is to facilitate the evaluation of design research results.

Furthermore, in their guideline 6, Hevner et al. (2004) characterize the design process as a search process: “The search process for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment” (Hevner et al., 2004). Hence, during the design process, the design researcher probably tries several means to solve the problem, receives unsatisfactory results, and consequently reshapes his or her approach. Hence, the design process can be characterized as an evolutionary process which should be governed by theories from the problem environment.

In sum, the role of theories in IS design research is threefold: theories are the output for the purpose of communicating and evaluating design knowledge, theories are the input for the construction of design artifacts, and theories shape the evolution of the IT artifact along the design process. Table 1distinguishes two groups of elements. The first group consists of goals and requirements and the subsequent design proposals that describe a solution, which matches the requirements and fulfils the goals. Overall, this group of elements describes “*how to do* something“ (Gregor, 2006, p. 628). The design proposal can be characterized as the claim of the design researcher to achieve utility. Hence, the second group consists of elements which provide information that substantiate this claim in an auditable way (Walls et al., 1992). The approaches presented in Table 1 provide the necessary structure to document the results of design research.

	Authors / Characteristics	Walls et al. (1992)	Markus et al. (2002)	Verschuren and Hartog (2005)	Gregor and Jones (2007)	Schermann et al. (2007)
Elements for describing the solution	Goals and requirements	Meta-requirements	User requirements	Contextual, functional, and user requirements	Purpose and scope	Context and problem
	Design proposal	Meta-design	Design principle	Structural specfication	Principles of form and function	Solution model
Elements for design theorizing	Design justification	Kernel theories	Kernel theories	Assumptions	Justificatory knowledge	Theory references
	Design evolution	-	-	-	Artifact mutability	Pattern references
	Design evaluation	Testable hypotheses	Testable hypotheses	Evaluation stages	Testable propositions	Consequences

Table 1: Approaches to design theories in IS design research

From a methodological point of view, the subsequent questions are how to proceed when incorporating theories in the design process and deducing testable hypotheses from the design results. Table 1 shows that the evolutionary aspect of design research has not been recognized until recently (Gregor and Jones, 2007; Schermann et al., 2007). The search process as described by Hevner et al. (2004) requires design researchers to document and trace their decisions and the corresponding justification across several iterations. Hence, design researchers need to track their decisions and the underlying linkages between informing theories, aspects of the design proposal, and resulting hypotheses (Gregor and Jones, 2007).

Surprisingly, little attention has been paid to developing methodological guidance on the use of theories in design research (Venable, 2006). We use the design research section on isworld.org (Vaishnavi and Kuechler, 2004) as example to clarify this position. In their extensive review of design research methodology, Vaishnavi and Kuechler (2004) characterize the role of theory in design research as informative: “[…] The design process, when interrupted and forced back […] contributes valuable constraint knowledge to the understanding of the always-incomplete-theories that abductively motivated the original design” (Vaishnavi and Kuechler, 2004). However, no further advice is given on how to proceed and document the learning process. How does the researcher document which design decision was grounded on which theoretical advice? Clearly, “more detailed and clear concepts would be helpful” to describe the use of theories in design research (Venable, 2006, p.1).

Overall, the general role of theory and its importance for rigorous design research has been widely recognized. However, existing approaches do not show how to proceed when making design decisions based on theory or deducing hypotheses from design proposals. In sum, our analysis shows that the required “linking mechanism” (Gregor and Jones, 2007) between goals, requirements, design proposals, underlying theories, and deduced hypotheses is largely unspecified.

3 FOUNDATIONS OF IS THEORIES AND DESIGN RATIONALES

In the previous section, we discussed that little methodological guidance exists on how to apply theoretical knowledge in design research. We suggest extending the concept of design rationales by enabling design researchers to develop a line of arguments leading from theory to an abstract design solution. In the following, we first clarify our notion of theories based on the typology provided by Gregor (2006). Second, we review the design rationale literature and select an appropriate approach for this paper. Both results are summarized as conceptual models, which will be then integrated in the next section.

3.1 The Notion of Theory in IS Research

The word theory is currently used in IS research to denote at least five different types of theoretical knowledge (Gregor, 2006). To integrate theories with design rationale approaches, we need to clarify the structure of such theories. A theory consists of a set of hypotheses (Popper, 2002). In general, a hypothesis is a correlation relationship based on two constructs (Balzer et al., 1987; March and Smith, 1995). The independent construct of the hypothesis is called *cause* while the dependent construct is called *effect*. According to the Structural Equation Modeling (SEM) approach, each construct is either represented by a set of indicators, which is called reflexive construct or is causing these indicators, which is then called a formative construct (Freeze and Raschke, 2007). The indicators operationalize the construct by providing the necessary means to measure the construct during theory testing. This notion of theory is summarized in the right area of Figure 2.

From our definition of theory we can draw three conclusions:

C1 Based on the definition of a construct, we conclude that two constructs with different sets of indicators are different; two constructs with the same sets of indicators are equivalent.

C2 Based on our definition of a hypothesis, it follows that two hypotheses are equivalent if they share the same constructs; otherwise the hypotheses are different.

C3 Based on our notion of theories, we conclude that theories with different sets of hypotheses are different and theories with the same sets of hypotheses are equivalent.

In sum, theories provide important information on cause-effect relations. However, we have shown that existing approaches to theorizing in IS research pay only little attention to the specific steps necessary to intertwine theorizing and design activities. We argue that the iterative nature of design research requires a particular focus on the specific decisions made during the design research process and the underlying arguments which led to the design artifact (Hevner et al., 2004). However, the challenge of documenting and justifying design decisions has been faced in the domains of software engineering and systems engineering, too. Thus, in the following section, we show that the notion of design rationales provide the foundation for developing theory-based design arguments, i.e., design principles.

3.2 Design Rationale Approaches

Research on design rationales is concerned with documenting and justifying particular decisions made during the design process (Rossi et al., 2004). The main goal of design rationale management is to document the relationship between the artifact, the underlying goals, the design proposal, and potential constraints (Louridas and Loucopoulos, 2000). This relationship can then be used to justify the design decisions which have led to the artifact. In doing so, the design rationale enables the design researcher to reason on his or her decisions at any given stage in the design process (Regli et al., 2000). For an extensive overview on existing approaches to rationale management see (Regli et al., 2000) and (Louridas and Loucopoulos, 2000).

Most of the approaches to design rationale management are based on the issue-based information systems (IBIS) approach (Kunz and Rittel, 1970). An issue in IBIS defines a certain problem that has occurred during the design phase. The design researcher can now establish various positions that may address the issue adequately. Then, different arguments can be made to either reject or substantiate a certain position. Finally, a position is being chosen and subsequently implemented in the design proposal (see the left area in Figure 2).

The following example explains IBIS: Let us assume we are in the process of constructing a new conceptual modeling grammar to resolve some issues with existing grammars, e. g. the Unified Modeling Language in a special problem domain (Object Management Group, 2005). The goal of the design project results in the requirement that the modeling grammar should enable its user to understand the resulting models easily. During the design process the design researcher has to decide how to represent the concepts of the modeling grammar. An *issue* in this discussion could be: “Should we assign graphical symbols to the grammatical concepts?”

The design researcher is now facing two *positions:* “yes” and “no”. A supporting *argument* for the “yes” position can be found in the experience, that graphical models are commonly easier to understand. An argument objecting the use of graphical symbols could arise from the costs of developing and introducing an appropriate graphical user interface when the modeling grammar is implemented. After the arguments for all the issue’s position have been discussed, the designer needs to select a position to resolve the issue. As the experience of the design researcher favors a graphical notation, the position “yes” is chosen. This results not only in the benefits of achieving better model comprehension by having a graphical representation (desired effect) but also in the potential constraint to the design process of implementing a costly graphical user interface (undesired side-effect).

4 DEVELOPING THEORY-BASED DESIGN PRINCIPLES

The goal of using theories in design research is to justify design decisions. The link between theory and design decision enables the design researcher to reason about the resulting behavior of the IT artifact prior to instantiation. Furthermore, this link allows deducing empirically testable hypotheses to foster the evaluation of the IT artifact. So far, we have discussed that design rationale approaches provide the means for establishing a traceable link between design decisions and experiences of the designer. Before that, we have analyzed the structure of theoretical arguments, i.e., cause-effect-relations. In the following we show how to develop design principles by grounding design rationales in theoretical arguments. The main idea of developing theory-based arguments is to reinterpret the cause-effect relation in a hypothesis as a goal-means relation (March and Smith, 1995; Weber, 1987).

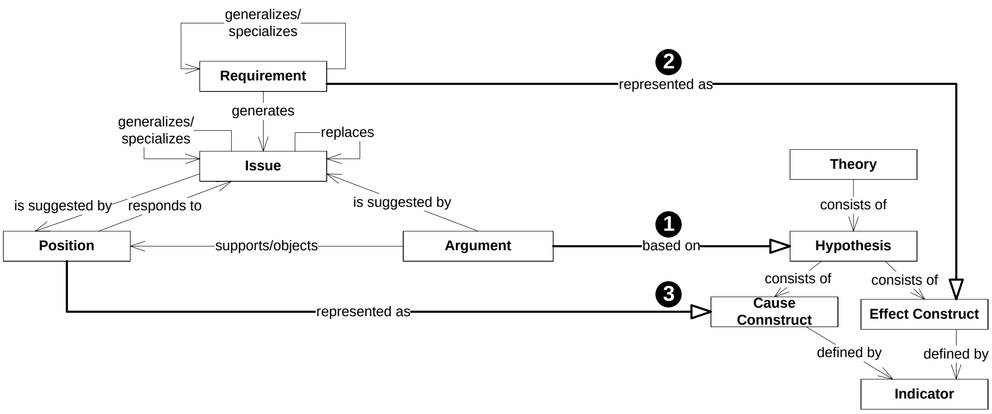

Figure 2: Integration of Theories into IBIS

To illustrate this reinterpretation, assume the following example: Larkin and Simon (1987) show that a diagrammatic notation (cause) results in a higher comprehension (effect) than textual notations. Inverting this relation and applying it to the construction of a modeling grammar allows the design researcher to formulate a theory-based argument: If the goal of a modeling grammar is to produce easily comprehensible models, *one way* to achieve it is to use a diagrammatic representation. However, this reinterpretation is possible only if the design researcher can actually influence the cause construct of the hypothesis, i.e., the design researcher is able to choose graphical symbols when developing the modeling grammar. In addition, there may be other mean–based on other hypotheses–which have the same effect.

In the context of the IBIS model, the starting point for developing theory-based arguments is the construct *Argument*. It provides the reasons for supporting a position or objecting to the position. The given set of arguments provides the foundation for making a decision with regard to an issue.

However, the argument itself does not explain *why* supporting or objecting a position resolves the issue, as long as the argument is not grounded in theoretical knowledge. In other words, if arguments are grounded in theoretical knowledge we do not only know, which arguments lead to the decisions made but also *why* these arguments contribute to the requirements of a design principle (➊ in Figure 2).

When assigning a hypothesis to an argument, we implicitly build upon two assumptions (➋ and ➌ in Figure 2):

A1 The cause construct of the hypothesis represents the position the argument refers to.

A2 The effect construct of the hypothesis represents the requirement the argument refers to via the position and issue.

These assumptions result from the reinterpretation of a hypothesis (cause-effect relations) as goal-means relations as described above. In this sense a requirement can be interpreted as (operational) goal to achieve. A position together with an issue represents possible solutions for this requirement (means). Consequently, the reinterpretation of a hypothesis must be represented by the position and requirements construct. Based on our notion of theory we know that the hypothesis' constructs are uniquely defined by its indicators (see Conclusions C1). In the light of A1 and A2, this means that the position and the requirement must be expressed and measured by the same indicators as the respective cause and effects constructs from the theory. Otherwise the mapping between the theoretical constructs (cause and effect) and the design constructs (requirement and position) is inconsistent.

Whenever the design researcher grounds an argument in a hypothesis, i.e., whenever the link between argument and hypothesis is established (➊ in Figure 2), the design researcher must map the hypothesis' cause construct (➋ in Figure 2) to the respective position and the hypothesis' effect construct to the requirement (➌ in Figure 2). The links ➋ and ➌ represent how the position and the requirement is interpreted, i.e., how the requirement and the position must be measured when evaluating the design principle.

Now we can deduce whether a theory-based argument supports a given position:

D1 The argument supports a position, if the selected position implements the cause construct and the hypothesis' correlation is positive.

D2 The argument supports a position, if this position is not based on the cause construct and the hypothesis' correlation is negative.

D3 In any other case, the argument objects to a position.

Let us return to the example of developing a modeling grammar and assume that the design researcher identified the following two hypotheses:

H1 If a graphical representation is used, the resulting models are easier to understand.

H2 If a modeling grammar has a graphical representation, then the automatic analysis of the models is more difficult.

Clearly, H1 has a positive correlation, whereas H2 has a negative correlation. As already explained above, the initial goal and thus the requirement of the modeling grammar is to enable its users to produce comprehensible models. The issue raised was whether to implement a graphical representation of the grammar. The available positions consist of "implementing a graphical representation" and "not implementing a graphical representation".

Both positions refer to the cause construct of H1 and H2, the graphical representation. The positive correlation between the graphical representation and comprehension of H1 supports the position of "implementing a graphical representation", because it advises the design researcher to use a graphical representation for the modeling grammar (D1). The negative correlation between graphical representation and automatic analysis of H2 supports the position of “not implementing a graphical representation”, because it advises against implementing a graphical representation (D2).

After all arguments were put forward the design researcher needs to select one of the positions to resolve the issue. This resulting decision manifests how the requirement should be interpreted, because the effect constructs of the hypotheses, are now related to the initial requirement (❸ in Figure 2). In our example following H1 means to refine the requirement “model comprehension” to “model comprehension by persons” while using H2 means to refine the requirement to “model comprehension by machines”.

Two important implications for the design researcher can be drawn from this analysis:

1) We discussed above that the constructs of a hypothesis are defined by a set of indicators. Consequently, selecting a position results in the decision of interpreting the requirement in the way the effect construct of the underlying hypotheses is being interpreted. By choosing the position of “implementing a graphical representation” we select to interpret and measure comprehension in accordance to hypothesis H1.
2) Selecting a position is rarely straightforward. In realistic cases more than one position is supported by theory-based arguments. Consequently, choosing a particular position means rejecting potentially important theoretical arguments. These arguments represent known side-effects of the particular design decision. In the example of the modeling grammar, the design researcher now knows that implementing a graphical representation will impede any automatic analysis.

After linking hypotheses to IBIS, we can now refine our notation of a design principle. A design principle is a set of requirements, issues, positions, selected positions, arguments and hypotheses. Grounding the design principle’s arguments in hypotheses provides not only the theoretical rationale for choosing among potentially useful positions but also explains how the requirements and the selected positions should be measured. In addition, the design principle shows possible drawbacks of the selected position as these drawbacks are represented by supporting arguments for all positions, which have not been chosen.

5 THEORY-BASED DESIGN PRINCIPLES FOR DESIGNING CONCEPTUAL MODELING GRAMMARS

To demonstrate the utility of our approach, we extend the example of constructing a new modeling grammar. The overall goal is to “construct a conceptual modeling grammar for inexperienced modelers”. In particular, the modeling grammar should not be a mere implementation of an existing framework such as the BWW model (Gehlert et al., 2005) but should be based on empirical evidence (Gehlert and Pfeiffer, 2007).

Before any issue can be raised, we need to decompose the goal into requirements. According to Wand and Weber (2003) a modeling grammar should enable users to interpret their models efficiently, i.e., to understand them easily. Consequently, our requirement discussed here is: “models should be easy to understand”. From a theoretical perspective, we seek theories that explain when and why models are easily comprehensible. To explain model comprehension we apply the work of Bodart et al. (2001). They distinguish between surface level comprehension and deep-level comprehension. Surface level comprehension is measured as proportion of the model to be recalled from memory after the model has been studied by a model user for a certain amount of time. Deep level comprehension is measured as the ability of a model user to answer questions regarding the model’s content (see the right area of Figure 3).

The authors studied the influence of optional properties on surface level comprehension and deep level comprehension where optional properties refer to attributes and relationships model elements may or may not possess. Bodart et al. (2001) found a significant positive effect between avoiding optional properties and deep level comprehension. In addition, they found a significant negative effect between avoiding optional properties and surface level comprehension. In sum, the authors showed that optional properties hinder deep level comprehension (negative correlation) but support surface level comprehension (positive correlation).

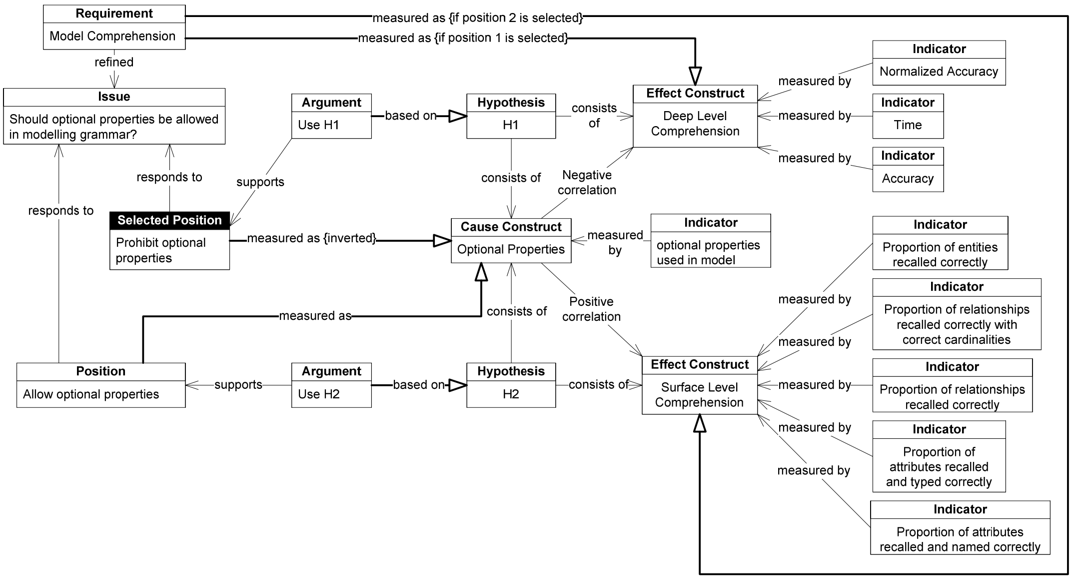

Figure 3: Discussion of Optional Properties

To describe the hypotheses used by Bodart et al. (2001) we need to know which indicators were used to measure them. The construct “surface level comprehension” was measured by the indicators “proportions of entities recalled correctly”, “proportion of relationships recalled correctly”, “proportion of attributes recalled and named correctly”, “proportion of attributes recalled and typed correctly” and “proportion of relationships recalled correctly with correct cardinalities” (Bodart et al., 2001, pp. 391). The construct “deep level comprehension” was measured by the indicators ”accuracy”, “time” and “normalized accuracy” (Bodart et al., 2001, pp. 397). The “optional properties” construct was measured binary – optional properties were either included in the model or were not included. It is important to note that the comprehension constructs “surface level comprehension” and “deep level comprehension” are different because they were measured with different sets of indicators (see Figure 3). As this theoretical setting explains the consequences of including optional properties, the central design issue for the new modeling grammar might be: Should the modeling grammar provide optional properties? Figure 3 summarizes the discussion about optional properties.

The issue whether to implement optional properties is related to the requirement “model comprehension”. There are two possible positions to resolve this issue, either to allow or to prohibit optional properties in the modeling grammar. Both positions are supported by the respective arguments and each argument is based on a hypothesis. For instance, the position “prohibit optional properties” is based on the hypothesis “using optional properties in models hinder deep level comprehension” (negative correlation). Since the position “prohibit optional properties” does not implement the hypothesis’ cause construct and the hypothesis’ correlation is negative, the hypothesis supports this position (see D2). This hypothesis, however, can only be applied in this argumentation process when its constructs are measured in the same way as the position and the requirement (assumptions A1 and A2). Committing to this assumption leads to specializing the meaning of the requirement “model comprehension” to “deep level comprehension”. In the same way, the assignment of the hypothesis to the arguments implies that the position is interpreted as “diagrams without optional properties”.

The same argumentation can be formulated for the position “allow optional properties”. It is assigned to the respective argument, which in turn is based on the hypothesis “allowing optional properties supports surface level comprehension”. By applying assumption A1 and A2, the meaning of the requirement “model comprehension” is specialized as “surface level comprehension”.

To resolve the issue, the design researcher needs to decide whether to allow optional properties in the modeling grammar. Figure 3 shows that the design researcher has decided to prohibit optional properties in the modeling grammar. Consequently, the design researcher chooses to interpret model comprehension as deep level comprehension rather than surface level comprehension. Based on this decision, we can derive possible constraints from our theory-supported arguments. Because of the choice to prohibit optional properties we know from the underlying theory, that the models of the modeling grammar will hinder surface level comprehension. In our case, was have inferred a side-effect from the design rational. This side-effect is expressed as constraint “the modeling grammar hinders surface level comprehension”.

Hence, the design principle can be summarized as follows: If a design researcher’s goal of a modeling grammar is to achieve comprehension, one way to realize it, is by prohibiting optional properties in the modeling grammar. This decision particularly enables deep-level comprehension of the models. At the same time it hinders the model’s surface-level comprehension.

Applying the hypothesis relating to optional properties in the design principle results in the following four effects:

- When using the design principle, the original requirement “model comprehension” is refined to the concept of “deep level comprehension” which is represented by the indicators “accuracy”, “time” and “normalized accuracy”.
- Since the argument is based on a generally applicable hypothesis, the design researcher now knows that the design principle supports deep-level comprehension whenever it is used. This claim is based on the logic of deductive reasoning. Whenever the design rationale is applied to a more concrete design situation, the hypothesis is also deductively applied in a more specific context. Because of the truth preserving nature of deductive reasoning (Balzer et al., 1987; Popper, 2002), the hypothesis will still hold in this specific context.
- Design principles enable the design researcher to keep track of the development of IS theories more easily. Assume that due to advances in theoretical IS research, the relation between optional properties and deep-level comprehension is falsified. Since the design researcher knows the link between the design principle and theory, the design decision of avoiding optional properties in the modeling grammar may be renegotiated.
- The proposed approach has also implications for theory testing. If the design principle developed above is used in different modeling grammars and the effect of deep-level comprehension cannot be achieved by this grammar, this indicates that the underlying hypothesis “avoiding optional properties supports deep level comprehension” may not be correct. Confronting theory experts with this situation may stipulate further theory testing (Weber, 1987).

6 CONCLUSION AND OUTLOOK

In line with Walls et al. (1992), we have argued that theories should be used when designing IT-artifacts. In particular, we discussed the justificatory role of theories in design research by integrating theoretical arguments in design rationales. We have introduced the concept of design principles to denote a set of requirements, issues, positions, selected positions, arguments and hypotheses. On the example of developing a modeling grammar we demonstrated the utility of design principles to capture theory-based design knowledge, which can be reused in similar design research projects.

Besides justifying design decisions, design principles have two additional benefits with regard to cumulative research activities in IS design research:

1) When progress in developing and testing IS theories falsifies a theory, the design researcher knows which design principle needs to be modified. If the design principle was not based on theory, this theoretical knowledge was incorporated into the experience of the design researcher and could not be easily identified in the design principle.

2) If a theory-based design principle fails to be useful when implementing an IT artifact, it indicates that the hypothesis used to back up the design principle's arguments may not be correct. This situation gives raise to extensive theory testing and may lead to a falsification of this theory.

Consequently, using design principles helps to bridge the gap between design research and theoretical research and intertwine both disciplines. However, we do not argue that every design principle should be backed up by theory. Since our approach is an extension to IBIS, it is still possible to derive design principles based on experience only. The main reason for deriving design principles based on experience is that IS theories are not yet available for every situation a design researcher may face. Furthermore, theories that are available may be not mature enough or too coarse or too fine grained to be used in design principles. For instance, the well accepted and tested Technology Acceptance Model (Davis, 1989) was too coarse grained to be usable for our example of designing a modeling grammar. Instead we needed to justify our design decision with the relatively new theory by Bodart et al. (2001), which was tested only once. The flexibility to develop design principles based on both theory and experience opens up two interesting avenues of future research to extend our concept of design principles:

1) Existing design principles may be complemented with theoretical knowledge, which explains the effects and side-effects of the design principle. Because a theory's hypothesis is always bound to its construct's indicators, applying the hypothesis means to explain how the positions and the requirements should be measured. Consequently, adding theoretical support to existing design principles leads to more precise understanding of the requirements and the solution of the deign principle.

2) If a designed artifact proves to fulfill its requirements, its design principles based on experience may be used to initiate theory building. Hypotheses can be derived from existing design principles by interpreting the decision's position as a cause construct and the requirement as an effect construct. Consequently, each design principle exhibits at least one hypothesis. These hypotheses can then be used to construct a theory for predicting (theory type IV in Gregor, 2006).

From a methodological perspective, our immediate subsequent research will focus on exploring the benefits of using theories of the types I, III, and V in design research and incorporate other argumentation theories, e.g. from Toulmin (1958).

7 REFERENCES

Balzer, W., Moulines, C. U. and Sneed, J. D. (1987) An Architectonic for Science: The Structuralist Program, D. Reidel Publishing Company, Dordrecht, Boston, Lancaster, Tokyo.

Bodart, F., Patel, A., Sim, M. and Weber, R. (2001) Should Optional Properties Be Used in Conceptual Modelling? A Theory and Three Empirical Tests, Information Systems Research, 12 (4), pp. 384-405.

Davis, F. D. (1989) Perceived usefulness, perceived ease of use, and user acceptance of information technology, MIS Quarterly, 13 (3), pp. 319-339.

Freeze, R. D. and Raschke, R. L. (2007) An Assessment of Formative and Reflective Constructs in IS Research, Proceedings of the 15th European Conference on Information Systems (ECIS2007), St. Gallen, Switzerland.

Gehlert, A., Buckmann, U. and Esswein, W. (2005) Ontology-Based Method Engineering, Proceedings of the 11th Americas Conference on Information Systems, Omaha, NE, USA.

Gehlert, A. and Pfeiffer, D. (2007) The BWW-Model as Method Engineering Theory, Proceedings of the 13th Americas Conference on Information Systems, Keystone, CO, USA.
Gehlert, A., Schermann, M., Pohl, K. and Krcmar, H. (2009) Towards a Research Method for Theory-Driven Design Research, Proceedings der 9. Internationalen Tagung Wirtschaftsinformatik, Business Services: Konzepte, Technologien, Anwendungen,Österreichische Computer Gesellschaft, Vienna, Austria, pp. 441-450.
Gregor, S. (2006) The Nature of Theory in Information Systems, MIS Quarterly, 30 (3), pp. 611-642.
Gregor, S. and Jones, D. (2007) The Anatomy of a Design Theory, Journal of the Association for Information Systems (JAIS), 8 (5), pp. Article 19.
Hevner, A. R., March, S. T., Park, J. and Ram, S. (2004) Design Science in Information Systems Research, MIS Quarterly, 28 (1), pp. 77-105.
Kunz, W. and Rittel, H. W. J. (1970) Issues as Elements of Information Systems, Institute of Urban and Regional Development, Univiversity of California, Berkeley, CA, USA.
Larkin, J. H. and Simon, H. A. (1987) Why a Diagram is (Sometimes) Worth Ten Thousand Words, Cognitive Science, 11 (1), pp. 65-99.
Louridas, P. and Loucopoulos, P. (2000) A Generic Model for Reflective Design, ACM Transactions on Software Engineering Methodology, 9 (2), pp. 199-237.
March, S. T. and Smith, G. F. (1995) Design and Natural Science Research on Information Technology, Decision Support Systems, 15 (4), pp. 251-266.
Markus, M. L., Majchrzak, A. and Gasser, L. (2002) A Design Theory for Systems that support emergent Knowledge Processes, MIS Quarterly, 26 (3), pp. 179-212.
Nunamaker, J. F. and Chen, M. (1991) Systems Development in Information Systems Research, Journal of Management Information Systems, 7 (3), pp. 89-106.
Object Management Group (2005) Unified Modeling Language (UML), version 2.0.
Popper, K. R. (2002) The logic of scientific discovery, Routledge, London.
Regli, W. C., Hu, X., Atwood, M. and Sun, W. (2000) A Survey of Design Rationale Systems: Approaches, Representation, Capture and Retrival, Engineering with Computers, 2000 (16), pp. 209-235.
Rossi, M., Balasubramaniam, R., Lyytinen, K. and Tolvanen, J.-P. (2004) Managing Evolutionary Method Engineering by Method Rationale, Journal of the Association for Information Systems, 5 (9), pp. 356-391.
Schermann, M., Böhmann, T. and Krcmar, H. (2007) A Pattern-based Approach for Constructing Design Theories with Conceptual Models, 15th European Conference on Information Systems (ECIS 07) (Eds, Österle, H., Schelp, J. and Winter, R.) University of St. Gallen, St. Gallen, Switzerland, pp. 1368-1379.
Toulmin, S. (1958) The Uses of Argument, Cambridge University Press, Cambridge, MA, USA.
Vaishnavi, V. and Kuechler, W. (2004) Design Research in Information Systems.
van Aken, J. E. (2004) Management Research Based on the Paradigm of the Design Sciences: The Quest for Field-Tested and Grounded Technological Rules, Journal of Management Studies, 41 (2), pp. 219-246.
Venable, J. R. (2006) The Role of Theory and Theorising in Design Science Research, Design Science Research in Information Systems and Technology (Eds, Chatterjee, S. and Hevner, A.), Claremont, CA, USA.
Verschuren, P. and Hartog, R. (2005) Evaluation in Design-Oriented Research, Quality & Quantity, 2005 (39), pp. 733-762.
Walls, J. G., Widmeyer, G. R. and El Sawy, O. A. (1992) Building an Information System Design Theory for Vigilant EIS, Information Systems Research, 3 (1), pp. 36-59.
Weber, R. (1987) Toward A Theory of Artifacts: A Paradigmatic Base for Information Systems Research, Journal of Information Systems, 1 (2), pp. 3-19.
Weber, R. (2003) Conceptual Modelling and Ontology: Possibilities and Pitfalls, Journal of Database Management, 14 (3), pp. 1-20.