**Association for Information Systems**
**AIS Electronic Library (AISeL)**

Wirtschaftsinformatik Proceedings 2009 Wirtschaftsinformatik

2009

# TOWARDS A RESEARCH METHOD FOR THEORYDRIVEN DESIGN RESEARCH

Andreas Gehlert
*Software Systems Engineering, Institute for Computer Science and Business Information Systems, Universität Duisburg- Essen*

Michael Schermann
*Santa Clara University*

Klaus Pohl
*Software Systems Engineering, Institute for Computer Science and Business Information Systems, Universität Duisburg- Essen*

Helmut Krcmar
*Chair for Information Systems, Technische Universität München*

Follow this and additional works at: http://aisel.aisnet.org/wi2009

Recommended Citation
Gehlert, Andreas; Schermann, Michael; Pohl, Klaus; and Krcmar, Helmut, "TOWARDS A RESEARCH METHOD FOR THEORYDRIVEN DESIGN RESEARCH" (2009). *Wirtschaftsinformatik Proceedings 2009*. 42.
http://aisel.aisnet.org/wi2009/42

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

# TOWARDS A RESEARCH METHOD FOR THEORY-DRIVEN DESIGN RESEARCH

Andreas Gehlert[1], Michael Schermann[2], Klaus Pohl[3], Helmut Krcmar[4]

***Abstract***
*In this paper we outline a new methodical approach for integrating theories into the design research process. Incorporating theories in design projects allows design researchers to reason on the effects of the IT artifact prior to its realization. We argue that design decisions should be transparent claims of utility based on theory-grounded arguments. Documenting design decisions requires the design researcher to integrate appropriate theories and document the rationale behind a particular design decision. Overall, we demonstrate on the example of constructing a new modeling grammar how to integrate theories in the design research process and discuss conflicts which occur when applying these theories.*

## 1. Introduction

*"But how does such work contribute to the progress of the IS discipline?"*

*[27, p. 8]*

The concept of theory is a cornerstone of improving the inter-subjectivity of design artifacts [25]. The relation between theory and design is twofold [15, cf. Figure 1]: First, theories can be used to derive design artifacts such as airplanes, information systems, computers etc. (theory before design). This relation is based on the assumption that any design artifact cannot work if it does not comply with the laws embodied in a theory. In addition, the artifact may be significantly improved if the underlying theory is explicit. For instance, "… one can design an airplane wing on the basis of tested, technological (black-box) rules, but such wings can be designed much more efficiently on the basis of tested and grounded technological rules, grounded on the laws and insights of aerodynamic and mechanics." [23, p. 228] Second, theories can be extracted from design artifacts to initiate theory building [17, theory after design]. The main reason for this activity is not only to extend the IS theory base but also to generalize design knowledge and to facilitate the comparison of design artifacts [27].

[1] Software Systems Engineering, Institute for Computer Science and Business Information Systems, Universität Duisburg-Essen, Schützenbahn 70, 45117 Essen, Germany
[2] Chair for Information Systems, Technische Universität München, Boltzmannstr. 3, 85748 Garching, Germany
[3] Software Systems Engineering, Institute for Computer Science and Business Information Systems, Universität Duisburg-Essen, Schützenbahn 70, 45117 Essen, Germany
[4] Chair for Information Systems, Technische Universität München, Boltzmannstr. 3, 85748 Garching, Germany

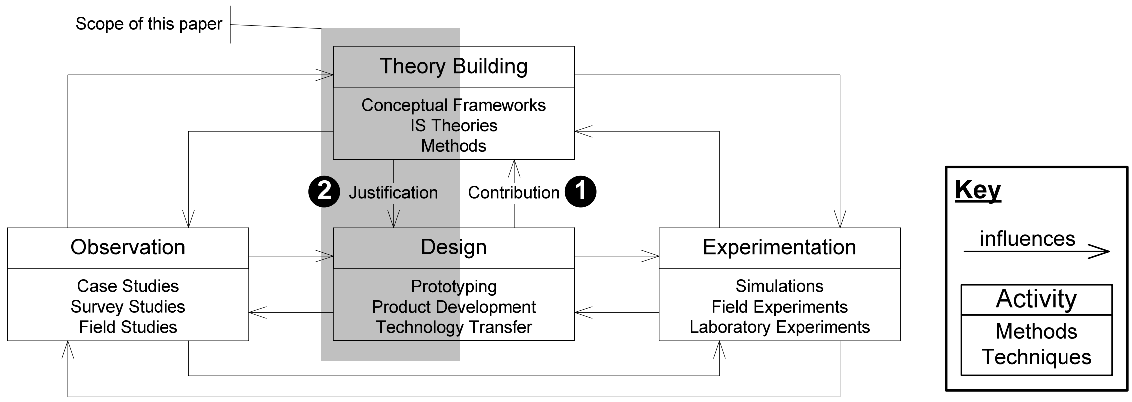


**Figure 1: Elements of theorizing in design research based on Nunamaker and Chen [15]**

In this paper, we focus on the contribution of theories for building design artifacts in the information systems discipline, e. g. we assume that already existing IS theories inform the design researcher when creating IT artifacts. Although many IS researchers argue that the design research process should be informed by existing theories [e. g. 8; 13; 22; 25], we argue that methodological guidance is needed to systematically integrate theories in the design research process.

We base our argumentation on the notion of design rationales and show how theories can be used to support the decision making process in design research. Design rationales serve as a documentation of the design process and enable the design researcher to trace his or her design decisions [11; 16]. In current design rationale approaches, however, the decision making process is guided by the experience of the design researcher only. This experience may include best practices and/or theoretical knowledge. Since the link between a particular design decision and the theoretical knowledge leading to this decision is implicit, design decisions cannot be clearly traced back to theoretical knowledge. In this paper we show how existing design rationale approaches can be combined with theoretical knowledge to overcome this problem.

The remainder of the paper is organized as follows: In section 2, we review existing design research approaches. We show that research has primarily focused on the structure of theories in design research. We describe our approach in the third section and discuss its benefits as well as its limitations. The paper closes with a summary of the results and an outlook of our further research plans.

## 2. Related Research on Theorizing in Design Research

Many authors addressed the methodological aspect of theorizing activities in design research [8; 13; 19; 24; 25]. In their seminal paper, Hevner et al. [9] for instance propose seven guidelines to determine the contribution of design research results to the IS knowledge base. For instance, their guideline 3 requires that the *"… utility, quality, and efficacy of [the] design artifact must be rigorously demonstrated."* To meet human requirements, the resulting artifact is usually complex and the result of an iterative process [22]. Consequently, Hevner et. al's guideline 6 characterizes the design process as a search process.

However, following both guidelines constitute a problem for the design researcher. They need to document and trace their decision-making process and the underlying information, e. g., requirements, design proposals, and development records. For instance, evaluating the quality of an artifact requires documenting the relationship of initial requirements, their evolution during the design process and the corresponding aspects of the artifact that addresses these requirements [24].

Therefore, design researchers have focused on providing a structure for systematizing and justifying decisions during the design process. Table 1 shows that all approaches provide categories for documenting design decisions: requirements, a corresponding design proposal and means for justifying the design proposal. Furthermore, all approaches require design researcher to specify ways to evaluate the design proposal. Additionally, Gregor and Jones (2007) and Schermann et al. (2007) provide a category to reflect the iterative character of design processes and the evolution of design artifacts.

**Table 1: Existing approaches to design theorizing in IS research**

| | Authors / Characteristics | Walls et al. (1992) | Markus et al. (2002) | Verschuren and Hartog (2005) | Gregor and Jones (2007) | Schermann et al. (2007) |
|---|---|---|---|---|---|---|
| Elements of design theorizing | 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 |
| | 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 |
| Guidance on how to... | … trace preparation of decisions | - | - | - | - | - |
| | … trace design decisions | - | - | - | - | - |
| | … trace justification of decisions | - | - | - | - | - |
| | … trace design propositions | - | - | - | - | - |

However, Table 1 clearly shows that all approaches fail to provide methodical guidance on the *actual* process of theorizing in design research. No guidance is given on how to relate design decisions to suitable theoretical justification. Overall, we argue that the process of developing the artifact requires integrated theorizing activities. In the remainder of this paper we show how to document design decisions by developing theory-based design arguments.

## 3. Justifying Design decisions with Theories

The structure of our approach, which is explained in this section, is depicted in Figure 2. To further describe this structure we use the construction of a new modeling grammar as running example. The description is split into three parts: First, the classical design research perspective is introduced. This perspective also provides the goals and requirements of the running example. Second, theories, which are relevant for our example, are elicited and the structure of theories is described. Lastly, design rationale – the integrating element between this theoretical perspective and the design research perspective is introduced.

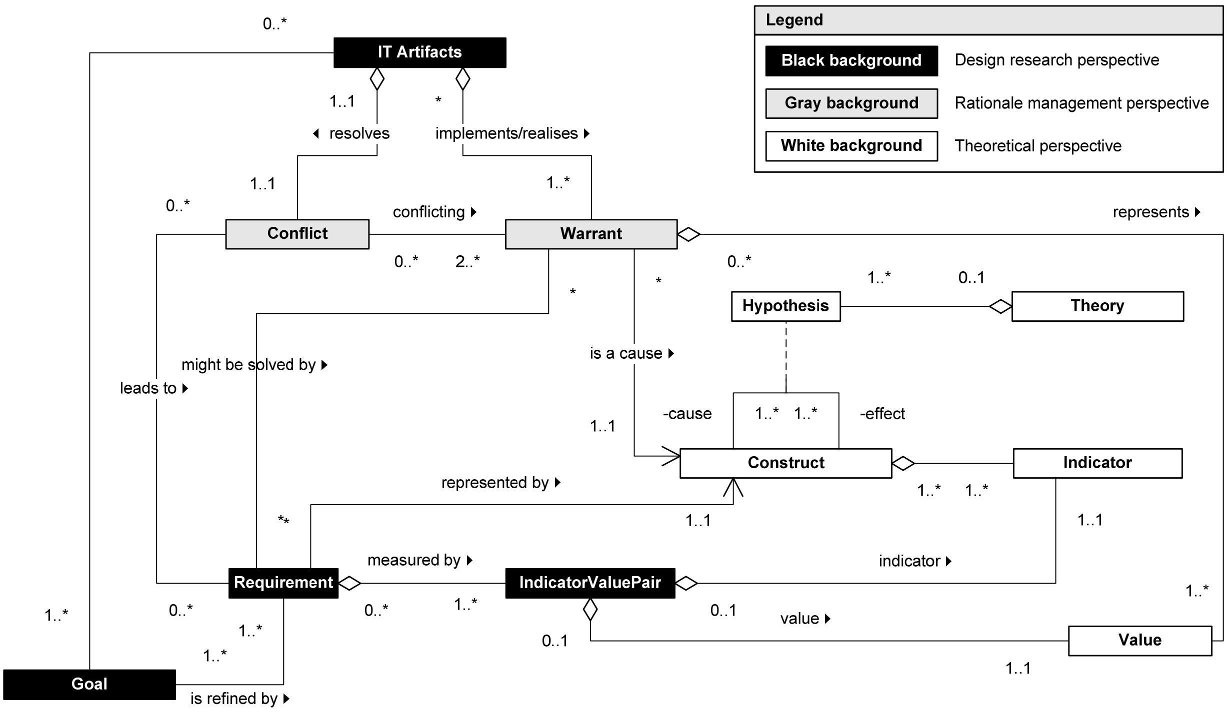


**Figure 2: Constructs of our approach**

## 3.1. Design Research Perspective

Each design process starts with goals [20]. Generally, goals give a coarse-grained picture of the envisioned IT artifact. The overall design goal in our example is the *"construction of a conceptual modeling grammar for inexperienced modelers"* [e. g. 28]. Goals are further decomposed into requirements. This decomposition process serves two purposes: First, the design process can be traced more easily by specifying what makes up this goal. Second, the decomposition of a goal into requirements allows evaluating the IT artifact. To achieve this measurability of all requirements, each requirement is attributed with indicators. The desired outcome of this measurement is described as value (cf. black elements in Figure 2). The decomposition of the goal into measurable indicators is described in the following exhibit:

> *Exhibit 1*: According to Wand and Weber [26, p. 364] a modeling grammar is a set of modeling constructs representing some proportion of the real world together with rules describing the lawful combinations of these constructs. The authors argue that such a modeling grammar should enable an efficient interpretation of its models [26, p. 366]. To interpret a model it should be easy to understand. To use a model it should aid the model interpreter when solving problems with this model. Hence, we decompose the general goal into two requirements "models should be easy to understand" (Rf-1) and "problems should be easy to solve with the models" (Rf-2; cf. Figure 3). The requirement Rf-2 should be measured by the property "problem solving accuracy" and the design researcher expects at least 75% problem solving accuracy during the evaluation of the artifact. Consequently, the design researcher assigns the value 75% to the property "problem solving accuracy". Rf-1 is decomposed accordingly.

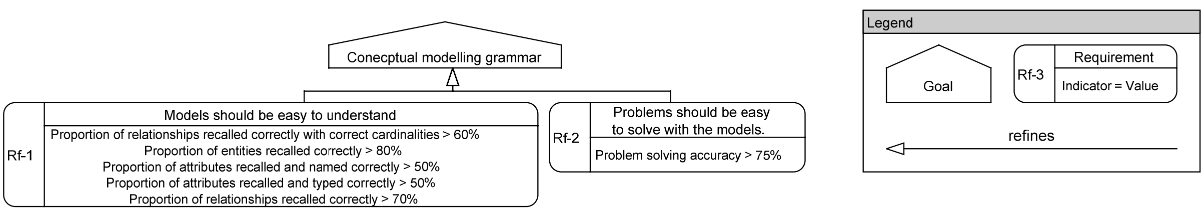


**Figure 3: Goal Decomposition**

In the following, we assume that requirements with different operationalizations, e. g., different sets of indicators are different and requirements with equivalent sets of indicators are identical.

*Consistency Rule 1*: Requirements with different sets of indicators are different; requirements with equivalent sets of indicators are identical.

To understand consistency rule 1 we need to express what we understand by different and identical indicators. Two indicators are said to be equivalent if they describe identical real world proportions. The indicators are different if they describe different real world proportions. Furthermore, two sets of indicators A and B are equivalent if for each indicator in set A there is an equivalent indicator in set B and vice versa. If this condition does not hold both sets are different.

So far we have followed traditional design science approaches. The next question is how to derive solutions, which fulfill the requirements. Since, solving problems is generally characterized as creative activities, design researchers implicitly draw upon theoretical knowledge [8; 14; 20; 22]. Achieving traceability in this creative task, the design researcher needs to link his or her existing theoretical knowledge to the specific characteristics of the IT artifact.

### 3.2. Theories in IS Research

We need to further investigate the structure of theories to prepare their integration with a rationale management system in the next section. A theory can be described as set of hypotheses ([2]; see [7] for other definitions of theory in IS research). A hypothesis is a cause-effect relationship based on at least two constructs [5]. One construct represents the hypothesis' cause and the other construct its proposed effect. Each construct may be represented by a set of indicators (reflexive construct) or is causing these indicators (formative construct) [6]. Overall, the indicators operationalize the construct and are, therefore, the way the construct is measured during theory testing. This notion of theory is represented by the white elements in Figure 2.

Given that the construct is well described by its indicators (construct validity) we can conclude:

*Consistency Rule 2*: Constructs with different sets of indicators are different; constructs with equivalent sets of indicators are identical.

For our running example we use theoretical hypotheses from the modeling research domain, which closely fit our goals of understandable models and models, which facilitate problem solving. These hypotheses are described in the next exhibit:

Exhibit 2: We apply the work of Burton Jones & Weber [4] and Bodart et al. [1] for our design work. Both papers show how to measure comprehension of models and their problem solving capability. Furthermore, they suggest hypotheses, which are relevant for our design problem. Bodart et al. [1] describe the construct surface level comprehension which is 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" [1, pp. 391]. Bodart et al. failed to falsify the hypothesis that the existence of optional properties has a positive impact on surface level comprehension using a free recall experiment with 52 participants. Therefore, their hypothesis is still valid (not falsified). Burton-Jones & Weber [4] measure the "comprehension" with the indicator "recall accuracy". The authors also failed to falsify the hypothesis that the existence of properties of relations negatively influences the comprehension of the model in a controlled experiment with 76 participants. Since the hypothesis was not falsified, it is still valid. Please note that model comprehension is measured with different indicators in the studies by Burton-Jones and Weber and Bodart et al. respectively. According to our

consistency rule 2 we cannot integrate the constructs but have to represent them separately (constructs C-3 and C-4 in Figure 4). Both papers measure “problem solving” by the indicator “accuracy”. Further on, both studies show that optional properties and properties of relations both have negative impacts on problem solving and provided empirical evidence for their claims by experimentation with 76 participants in the Burton-Jones and 96 participants in the Bodart et al. studies. The resulting set of the hypotheses is presented in Figure 4.

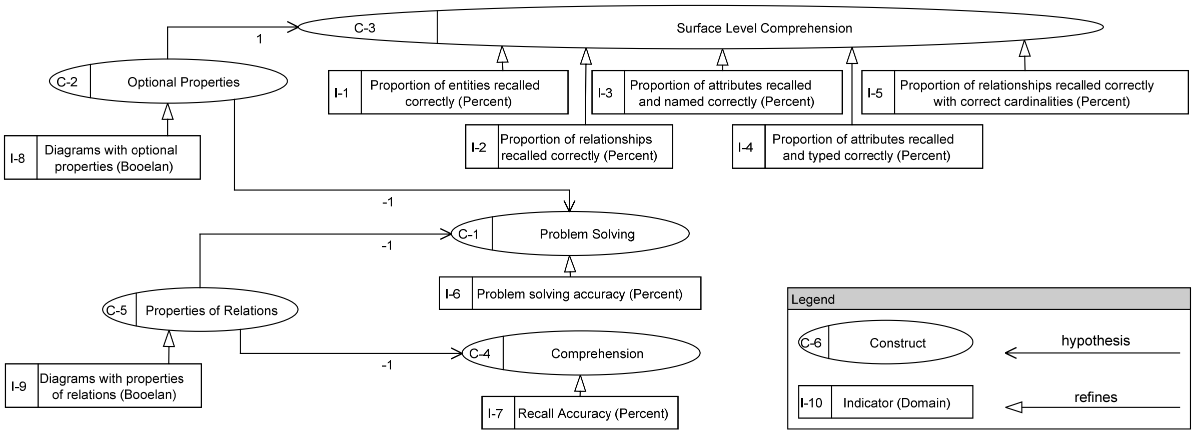

**Figure 4: Theory of a Conceptual Modeling Grammar**

Currently, we have explained the design research and the theoretical perspectives. In the next section we combine both perspectives by introducing the notion of design rationale.

### 3.3. Rationale Management Perspective

Design Rationales (DR) are founded in argumentation theory, which states that an argument always contains four elements [21]: claims (C), grounds (D), warrants (W) and rebuttals (R). *Claims* are always made on the basis of the presented *grounds*, e. g. data or facts. The *warrant* links the claim with grounds. A *rebuttal* represents known exceptions of this argument (see Figure 5).

When applying DR approaches to IT artifacts, the claim of an IT artifact is to have some utility, which is caused by its specific characteristics. While this claim is substantiated by experience in traditional design science approaches, we argue to use theoretical hypotheses. These cause-effect relationships are implemented as goal-means relations in IT artifacts and are, therefore, used as warrant for the design argument. The rebuttal in this setting is represented by hypotheses, which may contradict the warrant’s hypothesis (see Figure 5).

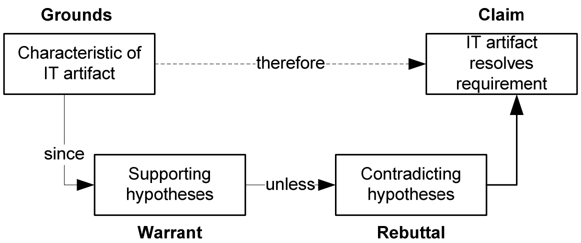

**Figure 5: Structure of design arguments [based on 3, pp. 44]**

Although the basic argumentation structure was implemented in many DR approaches [cf. 11 for an extensive review], these approaches extend and/or modify the Issue Based Information Systems (IBIS) approach, which we also apply here [10]. In IBIS design problems are represented by *issues*. As issues in IBIS represent agreed problems to be resolved they are best represented as grounds in Toulmin’s argumentation principle. Positions are assigned to issues and, thereby, claim to resolve this issue (claims). In the subsequent discussion process, *arguments* can be brought forward to support or to rebut a position (warrant and rebuttal). According to our arguments, these positions should be linked to hypotheses to ensure their traceability to theoretical knowledge.

When applying theories to design, the design researcher has to map requirements to "suitable" hypotheses. We define two consistency rules for this mapping:

*Consistency Rule 3*: Only requirements and constructs with identical sets of indicators can be matched.

*Consistency Rule 4*: Requirements can only be mapped to constructs which participate as effects in at least one hypothesis.

Consistency rule 3 ensures that the requirement exactly matches the theoretical construct. We thereby assume that the indicators define the requirement and the theoretical constructs fully (consistency rules 1 and 2). Consequently, a requirement, which is matched to a theoretical construct with diverging sets of indicators, implies that the requirement and the theoretical constructs have different meanings. In other words, consistency rule 3 prevents that requirements are mapped to theoretical constructs with different meanings. Consistency rule 4 ensures that requirements are matched with the "correct side" of the hypothesis. This mapping is described in the following exhibit:

Exhibit *3:* In exhibit 1 we introduced the requirement "models should be easy to understand" (Rf-1) and "problems should be easy to solve with the models" (Rf-2). To map these requirements to the hypotheses of exhibit 2, we must find appropriate theoretical constructs, which share the same set of indicators. According to consistency rule 3, we can only map Rf-1 to the construct "surface level comprehension" and Rf-2 to the "problem solving" construct. Since both constructs are effects in at least one hypothesis, this mapping is valid (consistency rule 4).

As argued above the warrant of an argument is represented by a hypothesis. More precisely the warrant is expressed by the hypothesis' construct, which represents its cause. In addition, the warrant assigns values to all the construct's indicators. These vales represent the design researcher's decision on how to implement the construct in the latter IT artifact. Due to conflicting requirements, it may be possible that two different warrants exist for the same requirement. Warrants conflict with each other if and only if they assign different values to at least one indicator. This is expressed by the following consistency rule:

*Consistency Rule 5*: A conflict is raised if and only if two warrants assign different values to at least indicator.

Conflicts can be theoretically traced back to conflicting causes of different hypotheses. It means that the respective warrants cannot be implemented together in the final IT artifact. Additionally, such a conflict cannot be resolved by theory since it originates from conflicting requirements. The design researcher has to manually choose between the conflicting positions by prioritizing the requirements. The resolution of the conflict is embodied in the final IT artifact. This decision is described in the following exhibit:

*Exhibit 4*: After the mapping of the requirements to the theoretical constructs, warrants can be formulated based on the hypotheses from exhibit 2. For instance, the research of Bodart et al. [1] indicates that avoiding optional properties in models enhances problem solving. Consequently, one way to achieve problem solving is to avoid optional properties in models. This fact is expressed by position P-1 (cf. Figure 6). Hence, the warrant assigns the value "false" to the indicator "diagrams with optional properties" to illustrate how the position will be considered in the design. However, we also want to achieve "surface level comprehension" and we know from theory that diagrams with optional properties help to attain this goal (P-3). Consequently, we assign the value

"true" to the indicator "diagrams with optional properties". Now we can analyze the resulting conflict. Obviously, we cannot have diagrams with and without optional properties at the same time. So, we must decide whether to include the optional properties construct in the modeling grammar. To visualize this, we raise a conflict between the conflicting positions (I-1 in Figure 6). This conflict must be solved by the design researcher. As discussed above this decision cannot be guided by theory because it reflects contradicting requirements. According to our theory it is not possible to achieve surface level comprehension and problem solving with the means "optional properties" at the same time. In our example, the design researcher decides to avoid optional properties in the modeling grammar. This decision is documented by the IT artifact S-1.

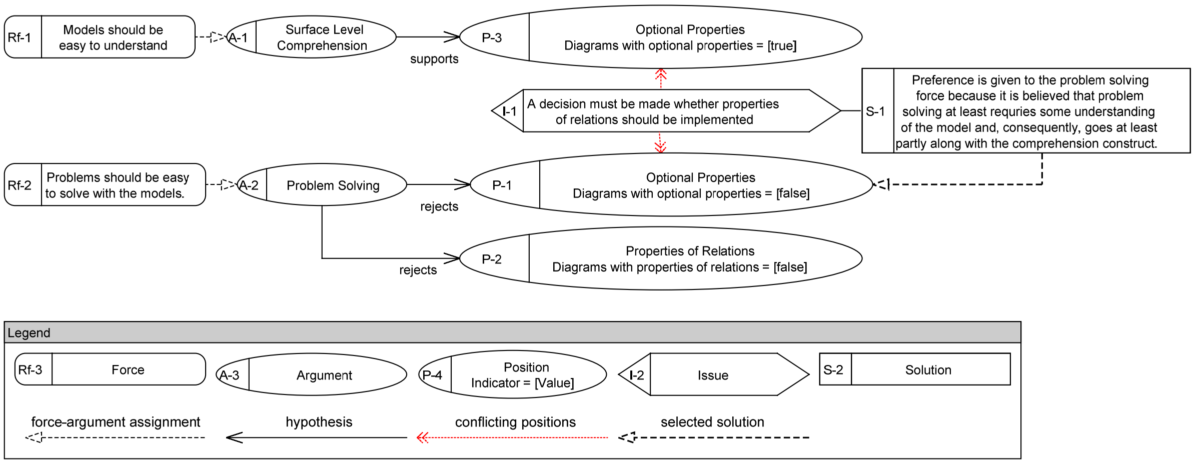


**Figure 6: Rationale Management**

## 3.4. Discussion

In line with Walls et al. [25] we have argued that hypotheses should be used when designing IT artifacts. Furthermore, we have shown that the systematic usage of hypotheses implies that the hypotheses' constructs and the requirements must have exactly the same granularity and scope. This granularity and scope is expressed via the set of indicators of theoretical constructs and requirements.

A mismatch of requirements and theory can be traced back to the following four situations:

1. There is no theoretical construct that shares at least one indicator for one requirement. In this situation there is no applicable theory for the design in focus.
2. The theoretical constructs have more indicators than the respective requirement, so that the theory is more fine grained than the requirement.
3. In case the theoretical construct has fewer indicators than the requirement the theory is coarser grained than the requirement.
4. The requirement and the theoretical construct share some indicators and differ in other indicators so that the theory and the requirements are incompatible.

In the situations two, three and four the design researcher has the choice to modify the requirement's indicators to achieve the same granularity and scope than the theoretical construct. If the theoretical construct is more fine grained than the requirement, the design researcher may decide to add additional indicators to the respective requirement. In this way, theoretical knowledge informs design research and helps the design researcher to decide how to measure requirements. The value of such adapted IT artifact's requirements has recently been demonstrated by Maiden [12].
With regard to the first situation, the design researcher may decide to proceed without a modification of the requirement's indicators and, therefore, without theoretical support [2]. In this case the design researcher can always construct the IT artifact based on experience [18]. However, our

framework requires documenting implicit hypotheses used during the design process as well. In this case, hypotheses are a "side-product" of our proposed methodology. As requirements are specialized by indicators to allow for their measurement (see above) they can be directly mapped to a theoretical construct (effect, see consistency rule 3). The same can be said about the warrant, which the design researcher extracts from the IT artifact (cause). Consequently, the derived hypotheses result directly from the design work. After a successful evaluation of the IT artifact, the designer may hand the derived hypotheses over to an empirical researcher who tests these hypotheses. This two-way-exchange between theoretical and design knowledge facilitates the intertwining of design research and theoretical research.

## 4. Conclusion and Outlook

In this paper we presented a methodical approach for substantiating and documenting design decisions with theories. The integration of references to theories in DR approaches enable the design researcher to anticipate the characteristics of the design before the artifact is instantiated. Even in the case of missing theories, our approach requires the design researcher to explicate his or her background knowledge as hypotheses.

The evaluation of the results of theory-based artifacts may stimulate theory development and testing and provides a link between design research and theoretical research. Furthermore, our approach helps design researchers to trace, revise and enhance design decisions during design research processes.

Our approach also clarifies the task of evaluating the design artifact by integrating theories into design and we can identify three types of evaluation [29]:

1. In out*put*-oriented evaluation the design researcher tests whether the artifact has the required *quality, i. e. whether the artifact is appropriately implemented.* Thus, the objective is to determine the gap between the required and implemented characteristics of the artifact. Overall, output-oriented evaluation stimulates the improvement of the artifact itself.
2. In out*come*-oriented evaluation the design researcher has to substantiate the claim of *utility* of the artifact, i. e. to show whether the IT artifact attains its goals. Failing to attain the goals can then be traced back to an inappropriate set of design arguments. Hence, outcome oriented evaluation shows how to improve the desired characteristics of the artifact.
3. However, if outcome-oriented evaluation shows that the chosen set of design arguments is inappropriate for attaining the goal, *"… an anomaly has arisen and the [applied] theory is suspect"* [27, p. 9]. Consequently, the theory used in design should be (re)evaluated to find out whether the theory is actually correct or whether the deviation from the design goal has another cause. In any case, the evaluation of theory-based artifacts creates data-sets that stimulate theory development and refinement.

However, our approach also shows an important limitation of theorizing in design research: Theory-based design research requires theories with a similar granularity level to link requirements with constructs of theories. Although, we could find appropriate theories in the case of our example, suitable theories are not available for every design issue. Therefore, we enable the researcher to explicate his or her own hypotheses which may then be subject to empirical research. Furthermore, different construct definitions in different versions of theories or different replication scenarios complicate the use of such theories in design work. Here, methodological guidance is required to resolve such conflicts. However, these limitations do not corrode our approach – as they do not conflict with its underlying assumptions – but provide an argument for a closer connection between the design research and the empirical research domains.

## 5. References


[1] BODART, F. et al., Should Optional Properties Be Used in Conceptual Modelling? A Theory and Three Empirical Tests, in: Information Systems Research 12 (2001) 384-405.
[2] BRIGGS, R. O., On theory-driven design and deployment of collaboration systems, in: International Journal of Human-Computer Studies 64 (2006) 573-582.
[3] BROCKRIEDE, W.; EHNINGER, D., Toulmin on Argument: An Interpretation and Application, in: Quarterly Journal of Speech 46 (1960) 44-53.
[4] BURTON-JONES, A.; WEBER, R., Understanding Relationships with Attributes in Entity-Relationship Diagrams, Proceedings of the 20th International Conference on Information Systems (ICIS 1999), Charlotte, NC, USA, 1999.
[5] CHMIELEWICZ, K., Forschungskonzeptionen der Wirtschaftswissenschaft, C. E. Poeschel Verlag, Stuttgart, Germany, 1979.
[6] FREEZE, R. D.; RASCHKE, R. L., An Assessment of Formative and Reflective Constructs in IS Research, in: Österle, H., Schelp, J.,Winter, R., (Eds.), Proceedings of the 15th European Conference on Information Systems (ECIS2007), June 7-9 2007, St. Gallen, Switzerland, University of St. Gallen, St. Gallen, 2007, pp. 1481-1492.
[7] GREGOR, S., The Nature of Theory in Information Systems, in: MIS Quarterly 30 (2006) 611-642.
[8] GREGOR, S.; JONES, D., The Anatomy of a Design Theory, in: Journal of the Association for Information Systems 8 (2007) 313-335.
[9] HEVNER, A. R. et al., Design Science in Information Systems Research, in: MIS Quarterly 28 (2004) 75-105.
[10] KUNZ, W.; RITTEL, H. W. J., Issues as Elements of Information Systems, Institute of Urban and Regional Development, University of California, Berkeley, CA, USA, 1970.
[11] LOURIDAS, P.; LOUCOPOULOS, P., A Generic Model for Reflective Design, in: ACM Transactions on Software Engineering Methodology 9 (2000) 199-237.
[12] MAIDEN, N., Servicing Your Requirements, in: IEEE Software 23 (2006) 14-16.
[13] MARKUS, M. L.; MAJCHRZAK, A.; GASSER, L., A Design Theory for Systems that support emergent Knowledge Processes, in: MIS Quarterly 26 (2002) 179-212.
[14] NEWELL, A.; SIMON, H. A., Human Problem Solving, Prentice-Hall, Englewood Cliffs, NJ, USA, 1972.
[15] NUNAMAKER, J. F.; CHEN, M., Systems Development in Information Systems Research, in: Journal of Management Information Systems 7 (1991) 89-106.
[16] REGLI, W. C. et al., A Survey of Design Rationale Systems: Approaches, Representation, Capture and Retrieval, in: Engineering with Computers 3/4 (2000) 209-235.
[17] ROMME, A. G.; ENDENBURG, G., Construction Principles and Design Rules in the Case of Circular Design, in: Organization Science 17 (2006) 287-297.
[18] SARKER, S.; LEE, A. S., Using a Positivist Case Research Methodology to Test Three Competing Theories-In-Use of Business Process Reengineering, in: Journal of the Association for Information Systems 2 (2002) 1-74.
[19] SCHERMANN, M.; BÖHMANN, T.; KRCMAR, H., A Pattern-based Approach for Constructing Design Theories with Conceptual Models, in: Österle, H., Schelp, J.,Winter, R., (Eds.), 15th European Conference on Information Systems (ECIS 07), University of St. Gallen, St. Gallen, Switzerland, 2007, pp. 1368-1379.
[20] SIMON, H. A., The Sciences of the Artificial, The MIT Press, Cambridge, MA, USA, 1969.
[21] TOULMIN, S., The Uses of Argument, Cambridge University Press, Cambridge, MA, USA, 1958.
[22] VAISHNAVI, V.; KUECHLER, W., Design Research in Information Systems, 2004.
[23] VAN AKEN, J. E., Management Research Based on the Paradigm of the Design Sciences: The Quest for Field-Tested and Grounded Technological Rules, in: Journal of Management Studies 41 (2004) 219-246.
[24] VERSCHUREN, P.; HARTOG, R., Evaluation in Design-Oriented Research, in: Quality & Quantity 6 (2005) 733-762.
[25] WALLS, J. G.; WIDMEYER, G. R.; EL SAWY, O. A., Building an Information System Design Theory for Vigilant EIS, in: Information Systems Research 3 (1992) 36-59.
[26] WAND, Y.; WEBER, R., Research Commentary: Information Systems and Conceptual Modeling -- A Research Agenda, in: Information Systems Research 13 (2002) 363-377.
[27] WEBER, R., Toward A Theory of Artifacts: A Paradigmatic Base for Information Systems Research, in: Journal of Information Systems 1 (1987) 3-19.
[28] WEBER, R., Conceptual Modelling and Ontology: Possibilities and Pitfalls, in: Journal of Database Management 14 (2003) 1-20.
[29] WEISS, C. H., Evaluierungsforschung: Methoden zur Einschätzung von sozialen Reformprogrammen, Westdeutscher Verlag, Opladen, 1974.