Extending the Discussion of Model Quality: Why Clarity and Completeness may not always be enough

Jan Mendling[1] and Jan Recker[2]

[1] Vienna University of Economics and Business Administration
Augasse 2-6, 1090 Vienna, Austria
`jan.mendling@wu-wien.ac.at`
[2] Queensland University of Technology
126 Margaret Street, Brisbane QLD 4000, Australia
`j.recker@qut.edu.au`

Abstract. Quality of modeling for information systems analysis and design is an important field of research in which, however, a comprehensive and generally acknowledged understanding is still outstanding. Notions of “model” and “quality” often remain vague and focus on particular aspects such as “syntax” or “semantics” rather than a comprehensive perspective on model quality. In this paper we argue that it is foremost the question of modeling pragmatics that is of pertinence when trying to ascertain the quality of a modeling artefact. We illustrate how pragmatic concerns mediate traditional conceptions of model quality. We refer to the well-established Bunge-Wand-Weber representation model and discuss how pragmatic concerns affect the understanding of model quality in addition to the quality criteria provided by such ontology-based theories of modeling. We apply the formalism provided by Kühne to clarify the influence of pragmatic concerns on modeling as a mapping activity with choices.

1 Introduction

Modeling traditionally plays an important role in information systems analysis and design [1]. As modeling-based approaches and techniques have proliferated over the years, researchers and practitioners have attempted to specify reference frameworks based on which to compare, evaluate, and determine when to use these different techniques [2]. The fundamental principles underlying modeling have been discussed, among others, from the perspective of ontology (cf. e.g. [3]), metamodeling (cf. [4]), or epistemology (cf. e.g. [5]). These different disciplines have provided some answers to the questions on what, how, and why something has to be captured in a model. Unfortunately, however, a comprehensive and generally acknowledged understanding of these questions is still outstanding [6]. While prior research has addressed syntactic (e.g. [7]) and semantic aspects of modeling quality (e.g. [8]), especially pragmatic aspects of modeling have so far only scarcely been addressed in related research [6]. Lindland et al. discuss [9] in their framework the assertion of model quality on a pragmatic level, however, the notion of pragmatics is reduced to the correspondence between a model and its interpretation (whether a model can be understood by its audience). Only recently was the notion of pragmatics extended to transcend pure information delivery concerns to also address aspects of ’pragmatics of action’ (e.g. [10]).

In this paper we seek to examine the relationships between the *what* and the *how* question of modeling with a particular focus on the model creation process (cf. [11]). We refer to ontology-based theories for conceptual modeling as a theoretical foundation. In particular, we discuss how modeling as a mapping activity offers design choices to the modeler and what the consequences of these choices are. This way we contribute to a better foundation of the pragmatics of modeling. Correspondingly, the aim of our paper is to formalize the notion of modeling pragmatics and incorporate it into an existing framework for conceptual modeling. Against this background the paper is organized as follows. In Section 2 we introduce an ontology-based theory of representation that serves as a reference framework to our discussion. We selected the Bunge-Wand-Weber representation model motivated by its wide-spread adoption [12]. Our choice, however, has no immediate consequences for our approach and merely serves as an illustrating example. In Section 3 we discuss the principles of modeling. We mainly follow the argument of Kühne [4] who uses formal notations to clarify essential modeling concepts. According to his elaborations, modeling can be understood as a mapping activity that obeys certain principles. In Section 4 we discuss which design choices the modeler has to consider when he performs modeling as a mapping activity. Furthermore, we discuss the different alternatives and derive guidelines when a certain choice should be taken. Section 5 discusses related work before Section 6 concludes the paper.

2 Theoretical Foundations

From available approaches towards a foundation of conceptual modeling, e.g., based on action theory [14], semiotics [9, 10] or cognitive theory [15], we deemed ontology-based theories (e.g., [16]) a most suitable starting point based on the observation that, in their essence, computerized Information Systems are representations of real world systems. Real world systems, in turn, can be explained and described using ontology - the study of the nature of the world. Ontologies attempt to organize and describe what exists in reality, in terms of the properties of, the structure of, and the interactions between real-world things [17]. Wand and Weber [8, 18] suggest that ontology can be used to help define and build models of information systems that contain the necessary representations of real world constructs, including their properties and interactions. Hence, they developed and refined a set of models based on an ontology defined by Bunge [17] for the evaluation of modeling techniques and the scripts prepared using such techniques. The BWW representation model is one of three theoretical models defined by Wand and Weber [18] that make up the BWW models. The representation model serves as an ontological model that characterizes real-world phenomena in information systems domain that modelers may seek to have represented in their conceptual model. It articulates a set of necessary and sufficient constructs to represent any real-world phenomenon a user might choose [19]. Its key constructs can be grouped into four clusters: things including properties and types of things; states assumed by things; events and transformations occurring on things; and systems structured around things. Table 1 shows a summary of the BWW model constructs.

In order to illustrate the most fundamental notions, consider the example of a human. A human is a *thing* in this world, independent from the actual physical existence

Table 1. Constructs of the BWW model, arranged in clusters. Adapted from [19]

BWW Construct	Cluster	Description and Explanation
THING	Things including properties and types of things	A thing is the elementary unit in the BWW model. The real world is made up of things. Two or more things (composite or simple) can be **associated** into a **composite** thing.
PROPERTY in general in particular hereditary emergent intrinsic non-binding mutual binding mutual Attributes		Things possess properties. A property is modeled via a function that maps the thing into some value. For example, the attribute "weight" represents a property that all humans possess. In this regard, weight is an attribute standing for a property **in general**. If we focus on the weight of a specific individual, we would be concerned with a property **in particular**. A property of a composite thing that belongs to a component thing is called a **hereditary** property. Otherwise it is called an **emergent** property. Some properties are inherent properties of individual things. Such properties are called **intrinsic**. Other properties are properties of pairs or many things. Such properties are called **mutual**. **Non-binding mutual** properties are those properties shared by two or more things that do not "make a difference" to the things involved; e.g. order relations or equivalence relations. By contrast, **binding mutual** properties are those properties shared by two or more things that do "make a difference" to the things involved. **Attributes** are the names that we use to represent properties of things.
CLASS		A class is a set of things that can be defined via their possessing a single property.
KIND		A kind is a set of things that can be defined only via their possessing two or more common properties.
STATE	States assumed by things	The vector of values for all property functions of a thing is the state of the thing.
CONCEIVABLE STATE SPACE		The set of all states that the thing might ever assume is the conceivable state space of the thing.
LAWFUL STATE SPACE		The lawful state space is the set of states of a thing that comply with the state laws of the thing.
STATE LAW		A state law restricts the values of the properties of a thing to a subset that is deemed lawful because of natural laws or human laws.
STABLE STATE		A stable state is a state in which a thing, subsystem, or system will remain unless forced to change by virtue of the action of a thing in the environment (an external event).
UNSTABLE STATE		An unstable state is a state that will be changed into another state by virtue of the action of transformations in the system.
HISTORY		The chronologically-ordered states that a thing traverses in time are the history of the thing.
EVENT	Events and transformations occurring on things	A change in the state of a thing is an event.
CONCEIVABLE EVENT SPACE		The event space of a thing is the set of all possible events that can occur in the thing.
LAWFUL EVENT SPACE		The lawful event space is the set of all events in a thing that are lawful.
EXTERNAL EVENT		An external event is an event that arises in a thing, subsystem, or system by virtue of the action of some thing in the environment on the thing, subsystem, or system.
INTERNAL EVENT		An internal event is an event that arises in a thing, subsystem, or system by virtue of lawful transformations in the thing, subsystem, or system.
WELL-DEFINED EVENT		A well-defined event is an event in which the subsequent state can always be predicted given that the prior state is known.
POORLY DEFINED EVENT		A poorly-defined event is an event in which the subsequent state cannot be predicted given that the prior state is known.
TRANSFORMATION		A transformation is a mapping from one state to another state.
LAWFUL TRANSFORMATION stability condition corrective action		A lawful transformation defines which transformations occurring on a thing are lawful. The **stability condition** specifies the states that are allowable under the transformation law. The **corrective action** specifies how the values of the property functions must change to provide a state acceptable under the transformation law.
ACTS ON		A thing acts on another thing if its existence affects the history of the other thing.
COUPLING binding mutual property		Two things are said to be coupled (or interact) if one thing acts on the other. Furthermore, those two things are said to share a **binding mutual property** (or relation).
SYSTEM	Systems structured around things	A set of things is a system if, for any bi-partitioning of the set, couplings exist among things in the two subsets.
SYSTEM COMPOSITION		The things in the system are its composition.
SYSTEM ENVIRONMENT		Things that are not in the system but interact with things in the system are called the environment of the system.
SYSTEM STRUCTURE		The set of couplings that exist among things within the system, and among things in the environment of the system and things in the system is called the structure.
SUBSYSTEM		A subsystem is a system whose composition and structure are subsets of the composition and structure of another system.
SYSTEM DECOMPOSITION		A decomposition of a system is a set of subsystems such that every component in the system is either one of the subsystems in the decomposition or is included in the composition of one of the subsystems in the decomposition.
LEVEL STRUCTURE		A level structure defines a partial order over the subsystems in a decomposition to show which subsystems are components of other subsystems or the system itself.

(consider Bob the Builder, for instance). A human is endowed with *properties* that we can describe and perceive through *attributes* (e.g., the hair color of Bob the Builder). Another example could be the attribute IQ that could potentially serve as a representation of the human property ’intellect’ (although we should know that IQ is a rather obscure measure of intellect). Some of the properties we describe via attributes are properties *in general*, which can be ascribed to all humans (e.g., weight) and some are *in particular*, which can be ascribed to a specific human (e.g., Bob the Builder’s weight). Things can be grouped in *classes* (e.g., humans that are fictional characters) and which are characterized by *mutual* properties (all fictional characters have the property of not being physically existent).

Things are further assuming certain *states* during their lifecycle. A state is a vector of all attributes at a given point in time (the height and weight of Bob the Builder at a Sunday afternoon). A thing (like a human) may assume different states. There are certain *laws* that govern the traversal between states (e.g., a human may traverse from the state ’alive’ to the state ’dead’ but not vice versa). The collection of states that are lawful to a thing (i.e., which a thing may assume at some stage) is the *lawful state space* of the thing. The traversal of a thing from one state to another is called a *transformation* (e.g., Bob the Builder colors his hair from yellow to green). *Events* that may occur require things to change their state via a transformation. These events may be *external* or *internal* to the thing (e.g., the occurrence of lightning that changes Bob the Builder’s hair color to a dark black would be an external event).

Finally, things can be set into a *system* of things. Bob the Builder, for instance, is coupled to his mother and father, together forming a family system. Systems can be *decomposed* into subsystems or *composed* to a supersystem. Systems are also differentiated from their *environment* (e.g., Bob the Builder’s family has some neighbor families in its environment).

3 Modeling as a Mapping Activity

Ontology-based modeling theories such as the BWW representation model define *what* general entities can be observed in the real world. However, they usually do not provide an explicit answer to the question *how* these entities can be represented in a model to articulate a given real-world domain. More precisely, while ontological reference systems provide sets of representation concepts to faithfully articulate real-world phenomena, they usually fail to describe how the set of representation concepts should be arranged (or composed together) in order to arrive at meaningful and moreover purposeful articulations of real-world phenomena.

We turn to Stachowiak’s general model theory [20] to further elaborate on this point. Generally speaking, a model can be understood as an abstraction of a real or language-based system (cf. e.g. [4]). According to Stachowiak [20], a model possesses three features. First, it has a *mapping* feature. Since a model can be regarded as a language-based system there must be a relation between it and the “original” system. A consequence of this perception is that a model has characteristics of a role. As a second aspect, this mapping has a *reduction* feature, i.e., the model includes only a subset of properties of the original. Finally, the model is created with certain *pragmatics* in mind. Therefore, the model and the “original” system need to be consistent with respect to those characteristics that are relevant for the purpose of the model.

According to Kühne [4] modeling as a mapping activity can be characterized as an abstraction relation

$$\alpha = \tau \circ \alpha' \circ \pi$$

that consists of a projection π, a further abstraction α' that depends on the role of the model, and a translation τ that maps to a particular modeling language. Furthermore, he distinguishes a token model role and a type model role. *Token models* represent singular aspects of an original system such as “Bob the Builder has a yellow hat.” Such models do not provide a further abstraction (α' is an identity function) beyond projection and translation. In contrast to that, a *type model* involves a classification Λ of elements, i.e., $\alpha' = \Lambda$. Accordingly, a type model would deal with statements such as “Builders have yellow hats” and Bob would belong to the extension of such a classification. In essence, a classification relates to an equivalence relation. In the simple case, this equivalence relation such as “is builder” establishes a partition into two disjoint types “Builder” and a rest-type. As a special kind of classification, a generalization establishes a super-type like e.g. “Worker” that implies a union over several equivalence classes including “Builder”, “Plumber”, or “Electrician”. Since all these sub-functions of the abstraction function α can be isomorphisms in theory and are actually homomorphisms in practice, the characterization of α matches the reduction feature of modeling.

The pragmatics (i.e. the purpose) of the modeling task guides the choice of abstractions that are introduced in a model. Depending on the given purpose, the chosen form of abstraction can result in different models that have little or even no overlap at all. Consider Bob to be ill, then the medical file as a model of his well-being could record aspects such as results, medical attendance, or diet sheets. In another context, Bob’s payment information (such as bank account number or annual salary) might be relevant. In essence, the purpose determines the projections π to be made in the abstraction.

Evaluation of models by means of ontological reference systems [12] essentially deals with the translation sub-function τ and analyzes whether an aspect of an original system (conceptualized generically in the form of a representation model) can be represented appropriately. Two main evaluation criteria can be identified: If the translation sub-function τ involves a choice between alternative elements, there is a lack of *ontological clarity* [19]. Three forms of lacking ontological clarity are distinguished, viz. construct redundancy, construct overload and construct excess. If there is an appropriate translation target missing, there is a lack of *ontological completeness* [19].

While these criteria allow for conclusions to be drawn about the *representation fidelity* of a model or modeling language, they bear little explanatory power when considering modeling pragmatics. Obviously, a translation that is not complete might still be satisfactory if it turns a blind eye to those real-world aspects that do not serve the purpose. For instance, if a modeler does not have a need to graphically articulate system decompositions in a model then, certainly, she would not be concerned with whether the modeling language of choice actually provides representation forms for system decompositions (which would potentially result in an incomplete model) or whether there would be several representation forms available (which would potentially decrease the clarity of the model). Obviously, the opposite case also may hold true. In the area of process modeling, for instance, it was found that for some modeling purposes (such as devising executable workflow specifications), elements may be included in a model that do not represent any real-world concept per se (and hence would be classified as construct excess) but are nevertheless used to articulate implementation details [23].

Prior research informs us that the criteria of ontological clarity and completeness can easily be assured by considering, for instance, the BWW representation model as a metamodel of the translation target (e.g. [24–26]). In order to foster our discussion on modeling pragmatics, in the following section, we will discuss the role of projection and classification if such ontological clarity and completeness is guaranteed.

4 Modeling Choices

In this section, we investigate different mapping choices that have to be considered in the process of modeling, i.e., projection π and abstraction in terms of classification Λ. We refer to the elaborations of Kühne [4] as a formal foundation for these terms. Moreover, we consider the constructs of the BWW representation model both as the range and the domain of this mapping activity in order to guarantee ontological clarity and completeness of the translation relation. For an original system that is part of the range we also refer to as a source system, and for a model system we use alternatively the term target system. Furthermore, we discuss the consequences of certain mapping choices and under which preconditions they might be appropriate. For this discussion we focus on type models.

4.1 Analysis

Following our elaborations above we are now able to formally describe how pragmatic concerns determine the choices a modeler has to set in a modeling exercise. Accordingly, we show which combinations of BWW constructs (as measurements for ontological completeness and clarity of the translation relation) should be considered when the modeler is given the choice to reduce the set of originial aspects to be captured in the model. The aim of this section then is to specify the seemingly rather obvious notion of pragmatics in the modeling process in order to provide a formalized understanding on which subsequent works can be based.

Thing: (π) A thing T can be projected to a thing in the model or it can be mapped to the empty set. The latter case may be appropriate if there is no class relevant for the model that has T as an instance and if no properties, states, events, transformations, and subsystems related to T are relevant for other things that are included in the model. (Λ) A thing can be mapped to a class C if C includes all properties of T that are relevant for the modeling purpose. The choice for an adequate equivalence relation relates to the set of properties that is not skipped. In case of a classification singular properties, states, and events of the thing get lost.

Consider Bob's bank account. For a medical file this thing would not be relevant (at least not in a narrow sense) while in the context of payment data it would have to be included in the projection. Then, it could be classified to "bank account" as a type with the relevant properties account number, bank address, etc.

Property: (π) A property P of a thing T can be projected to a property in the model or it can be mapped to the empty set if it is not relevant for T and if there are no states, events, and transformations related to it that are relevant. (Λ) In case of a classification a property of a thing becomes a property of a type.

Consider the properties “date entering the company” and “date leaving the company.” Again, these properties might not be directly relevant for a medical file. In an accounting context these properties could be interesting since they relate to a transformation “delete employee from payroll.” Accordingly, they would have to be included in the projection.

Class: (π) A class C can be projected to a class in the model if it is relevant for the modeling purpose. It can be skipped if no property, state, event, transformation, or subsystem that is related to an instance is relevant for the modeling purpose. (Λ) A class C can be classified to another class if no instance of the original class, but only the class itself is relevant for the modeling purpose since singular instances of the original class are no longer captured.

Consider the class “Builder.” This class might be relevant for a payment model since this class could be related to a payment scheme. Therefore, we would have to include it in the projection. Furthermore, we could classify “Builder” as an instance of a class “Employee Category”. Bob would be an instance of “Builder,” but not of “Employee Category”.

State: (π) A state S can be projected either to a state in the model system or to the empty set. If it is included in the target the properties related to that state must also be represented. If it is excluded there must not be a transformation related to that state in the model. (Λ) A classification of a state relates to the thing that is associated with this state to become a type.

Consider Bob’s hair color again. If a transformation of black hair to green hair is not relevant for the modeling purpose, we would be allowed to skip the state that relates to hair color. If this transformation would be relevant, we would need to include the pre- and post-state of it, and properties that relate to these states.

Event: (π) An event can be projected to an event in the target system if it is relevant for the modeling purpose. Then, the property that is changed by the event must also be included in the model. The property related to the event can only be skipped if also the event is not relevant to the modeling purpose. (Λ) Similar to states and properties the classification of an event relates to the thing associated with the event to become a type. The same observations can also be made for **Transformations**.

Again, if the change of hair color would be relevant, we would also have to model the hair color of Bob.

Coupling: (π) A coupling can be skipped if at least one thing that participates in the coupling is not relevant for the purpose. A coupling can be projected to the model if it is relevant. This projection can be done in different ways. In the simple case the coupling can be projected to a coupling in the model. This is appropriate if it must be able to navigate to both ends of the coupling. If one thing might depend upon the existence of the other, the second might be mapped to a subsystem. Furthermore, if it is sufficient to record only whether a coupling to a second thing exists, it might be appropriate to map the second thing to become a property of the first. (Λ) For the classification the choice of an appropriate equivalence relation must be taken. Similar considerations have to be made as for classification of thing.

System: (π) If a system S is not relevant, i.e. none of its subsystems, things, properties, states, events, and transformations are relevant for the modeling purpose, it can be skipped in the projection. Otherwise, there are choices to make about the projection target of the system. In the simple case, it can be mapped to a system in the model. If the subsystems of S are not relevant, it can be mapped to a class that simply has properties. In this case, the properties, states, events, and transformations of the sub-parts are no more visible. The system might also be mapped to a property of a supersystem that is represented in the model if only the fact whether it exists matters and none of its structural and behavioral details. (Λ) For the classification the problem of an appropriate equivalence relation arises again.

4.2 Implications

The discussion above permits the following conclusions to be drawn:

1. Projection involves a yes/no choice whether some aspect of the original system should be included in the model. This choice can *only* be made in accordance to a given modeling purpose and as such is independent of the modeling language that is used for the translation.
2. Due to the reduction feature of a modeling activity there are several choices available for the mapping target of things, couplings, and systems. Again, the adequate extent of reduction can only be determined with respect to a given modeling purpose, which governs the adequate granularity and level of detail of the model.
3. Classification involves a choice for at least one or multiple equivalence relations that capture those aspects of an original system that are of relevance before the background of the given modeling purpose. In comparison to the yes/no choice of a projection this classification choice is often less obvious. It may even be possible that alternative classifications might serve one and the same purpose. Still, it is related to the projection of properties of a thing to be classified since only captured properties (i.e. those properties that are explicitly articulated via attributes) can be the base for an equivalence relation.
4. Several choices of modeling cannot be made in isolation but are instead interlinked. This may constitute a problem since only the positive choices (the “includes”) become apparent in the model but not the negative choices. In order to validate the complete set of modeling choices made by the modeler, one would need some sort of reference to the modeling aspects that were intentionally skipped.
5. Ontological clarity must be regarded as a necessary condition for deriving models that meet a given modeling purpose. For ontological completeness this implication holds in most but not all cases. There may be some purposes that can be catered for with a (theoretically) ontologically incomplete modeling language. In conclusion, translation is the only modeling activity that is directly affected by the extent of representational fidelity in a modeling language.

Following this line of argumentation, we can draw some conclusions as to a formalized understanding of the quality of modeling. We theorize that the quality of modeling

q_α can be defined as a function of three parameters. First, the representational fidelity f of the modeling language influences the quality of the translation. Second, the clarity of the modeling purpose p enables clear decisions about including and excluding aspects of the original system in the model and about which classifications are appropriate. Third, the competence of the modeler m enables an appropriate translation, abstraction, and projection. Accordingly, we write

$$q_\alpha(f, m, p) = q_\tau(f, m) \circ q_{\alpha'}(p, m) \circ q_\pi(p, m)$$

In addition to the fact that this equation stresses the importance of a clear modeling purpose, it also emphasizes a point that we have somewhat excluded from our discussion so far, namely that the importance of modeling competence of the modeler is another key to modeling quality. Most notably, however, the representational fidelity, which so far has obtained the most attention in related literature, appears to be of secondary importance only. In fact, set aside the extensive amount of related work on ontological foundations of modeling (see [12] for an overview), there is only little research that has focused, for instances, characteristics of the individual that carries out modeling tasks. Noticeable is the work by Agarwal et al. [27] who investigated in detail the notion of modeler experience. Furthermore, recently have researchers started to also investigate other attributes of modeler competency (such as method expertise or domain knowledge) and their effect on modeling activities (such as model understanding) [28].

Against the background of lack of empirical insights into conceptual modeling, we deem our theoretical analysis a basis for further empirical work in the future that should focus on investigating the consequences of modeler competencies and modeling pragmatics on model creation or model understanding tasks.

5 Related Work

Little research has comprehensively investigated the notion of model quality [6]. Interesting is the work of Lindland et al. [9] who developed an understanding of conceptual model quality based on semiotic theory, defining a syntactic, semantic and pragmatic level of model quality. Recently, this work was revised by Krogstie et al. [10] who, most notably, extended modeling pragmatics as ’pragmatics of understanding’ to also include ’pragmatics of action’. In line with their arguments we have argued that the notion of pragmatics should transcend pure information delivery concerns to also address aspects of ’fit to purpose’. Accordingly, our discussion in this paper sought to address purpose-related concerns of model quality.

Early attempts of establishing quality criteria for modeling include the guidelines of modeling [29]. This approach, however, lacks a sound theoretical methodology, and provides only limited empirical proof as to its feasibility as a quality framework [30]. Ontology-based theories of modeling quality are quite wide-spread in the IS community (e.g. [12]). The use of ontology for the purpose of asserting modeling quality, however, has been critiqued of recent years, for instance, with respect to the lack of pragmatic aspects in ontology-based studies [35]. We have focused in this paper on the relationships and potential inter-dependencies between evaluations of semantic quality aspects and the modeling pragmatics that govern modeling tasks and requirements.

6 Conclusions

In this paper we conducted a theoretical analysis of the choices imposed upon a modeler in the model creation tasks (the “how” question of modeling) before the background of the fundamental premises of ontology-based reference systems for modeling (the “what” question of modeling). We showed how traditional notions of model quality (such as completeness and clarity of the resulting model) have to be put in perspective in accordance to modeling pragmatics as well as the characteristics and competencies of the modeling individual. From a theoretical perspective, our study forms a basis on which a better understanding of modeling activities can be established. We identified a set of under-represented factors that appear to be important for theorizing modeling quality over and above pure syntactic and semantic concerns. By recapitulating modeling choices that any modeling individual has to face we aimed to contribute to a more thorough understanding of the notion of quality in modeling.

Our conceptual study suffers from an obvious limitation in being a form of theoretical research. As such, our study can only be a-priori given the absence of empirical testing. In our future research we thus aim at empirically investigating in more detail the theorized notions of modeler competencies and modeling purpose and their effects on building and understanding “better” models.

References

1. Karimi, J.: Strategic planning for information systems: Requirements and information engineering methods. Journal of Management Information Systems **4** (1988) 5–24
2. Gorla, N., Pu, H.C., Rom, W.O.: Evaluation of process tools in systems analysis. Information and Software Technology **37** (1995) 119–126
3. Wand, Y., Weber, R.: Mario bunge’s ontology as a formal foundation for information systems concepts. In Weingartner, P., Dorn, G.J.W., eds.: Studies on Mario Bunge’s Treatise. Rodopi, Amsterdam, The Netherlands (1990) 123–149
4. Kühne, T.: Matters of (meta-) modeling. Software and Systems Modeling **5** (2006) 369–385
5. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. MIS Quarterly **28** (2004) 75–105
6. Moody, D.L.: Theoretical and practical issues in evaluating the quality of conceptual models: Current state and future directions. Data & Knowledge Engineering **15** (2005) 243–276
7. ter Hofstede, A.H.M., van der Weide, T.P.: Formalisation of techniques: Chopping down the methodology jungle. Information and Software Technology **34** (1992) 57–65
8. Wand, Y., Weber, R.: On the ontological expressiveness of information systems analysis and design grammars. Journal of Information Systems **3** (1993) 217–237
9. Lindland, O.I., Sindre, G., Sølvberg, A.: Understanding quality in conceptual modeling. IEEE Software **11** (1994) 42–49
10. Krogstie, J., Sindre, G., Jørgensen, H.D.: Process models representing knowledge for action: a revised quality framework. European Journal of Information Systems **15** (2006) 91–102
11. Gemino, A., Wand, Y.: A framework for empirical evaluation of conceptual modeling techniques. Requirements Engineering **9** (2004) 248–260
12. Green, P., Rosemann, M.: Applying ontologies to business and systems modeling techniques and perspectives: Lessons learned. Journal of Database Management **15** (2004) 105–117
13. Guizzardi, G.: Ontological Foundations for Structural Conceptual Models. Volume 015 of Telematica Instituut Fundamental Research Series. Enschede, The Netherlands (2005)

14. Agerfalk, P.J., Eriksson, O.: Action-oriented conceptual modelling. European Journal of Information Systems **13** (2004) 80–92
15. Evermann, J.: Towards a cognitive foundation for knowledge representation. Information Systems Journal **15** (2005) 147–178
16. Milton, S., Kazmierczak, E.: An ontology of data modelling languages: A study using a common-sense realistic ontology. Journal of Database Management **15** (2004) 19–38
17. Bunge, M.A.: Treatise on Basic Philosophy Volume 3: Ontology I - The Furniture of the World. Kluwer Academic Publishers, Dordrecht, The Netherlands (1977)
18. Wand, Y., Weber, R.: On the deep structure of information systems. Information Systems Journal **5** (1995) 203–223
19. Weber, R.: Ontological Foundations of Information Systems. Coopers & Lybrand and the Accounting Association of Australia and New Zealand, Melbourne, Australia (1997)
20. Stachowiak, H.: Allgemeine Modelltheorie. Springer-Verlag (1973)
21. Weber, R., Zhang, Y.: An analytical evaluation of niam's grammar for conceptual schema diagrams. Information Systems Journal **6** (1996) 147–170
22. Parsons, J., Cole, L.: What do the pictures mean? guidelines for experimental evaluation of representation fidelity in diagrammatical conceptual modeling techniques. Data & Knowledge Engineering **55** (2005) 327342
23. Recker, J., Rosemann, M., Indulska, M., Green, P.: Business process modeling: A maturing discipline? BPM Center Report BPM-06-20, BPMcenter.org (2006)
24. Green, P., Rosemann, M.: Integrated Process Modeling. An Ontological Evaluation. Information Systems **25** (2000) 73–87
25. Opdahl, A.L., Henderson-Sellers, B.: Ontological evaluation of the uml using the bunge-wand-weber model. Software and Systems Modeling **1** (2002) 43–67
26. Rohde, F.: An ontological evaluation of jacksons system development model. Australian Journal of Information Systems **2** (1995) 77–87
27. Agarwal, R., Sinha, A.P., Tanniru, M.: The role of prior experience and task characteristics in object-oriented modeling: An empirical study. International Journal of Human-Computer Studies **45** (1996) 639–667
28. Khatri, V., Vessey, I., V. Ramesh, P.C., Sung-Jin, P.: Understanding conceptual schemas: Exploring the role of application and is domain knowledge. Information Systems Research **17** (2006) 81–99
29. Becker, J., Rosemann, M., Schütte, R.: Grundsätze ordnungsmässiger Modellierung. Wirtschaftsinformatik **37** (1995) 435–445
30. Rosemann, M., Sedera, W., Sedera, D.: Testing a framework for the quality of process models a case study. In Kim, J.S., Lee, J.K., Suh, K., Wei, K.K., Lee, H., eds.: Proceedings of the 5th Pacific Asia Conference on Information Systems (2001) 978–991
31. Rosemann, M., Green, P.: Developing a meta model for the bunge-wand-weber ontological constructs. Information Systems **27** (2002) 75–91
32. Bodart, F., Patel, A., Sim, M., Weber, R.: Should optional properties be used in conceptual modelling? a theory and three empirical tests. Inf. Sys. Research **12** (2001) 384–405
33. Gemino, A., Wand, Y.: Complexity and clarity in conceptual modeling: Comparison of mandatory and optional properties. Data & Knowledge Engineering **55** (2005) 301–326
34. Wyssusek, B.: On ontological foundations of conceptual modelling. Scandinavian Journal of Information Systems **18** (2006) 63–80
35. Recker, J.: Towards an understanding of process model quality. methodological considerations. In Ljungberg, J., Andersson, M., eds.: 14th European Conf. on Inf. Systems (2006)