Using UML for Conceptual Modeling: Theory and Empirical Test

Joerg Evermann

University of British Columbia
Faculty of Commerce and Business Administration
Management Information Systems Division
evermann@interchange.ubc.ca

Abstract. Successful information systems (IS) development requires the understanding of the real world domain in which the IS is situated in and of which it is a representation. Developing such an understanding is the role of systems analysis, the first major step in IS development. Conceptual models developed during systems analysis are used to support understanding of and communication about the real world domain.
Recent years have seen the emergence of the object-oriented approach in general and UML specifically for IS design and implementation. However, no generally accepted modelling language has been proposed for use during IS analysis.
This study will examine the suitability of UML as a conceptual modelling language. This study comprises two parts. The first part studies UML from an ontological perspective, attaches real-world semantics and derives ontologically grounded rules for applying UML to conceptual modelling. It is argued that by following these rules, modellers will improve the performance of the resultant models. In a second step, the derived rules and proposed advantages must be empirically supported. An experimental study is designed for this purpose.

1 Introduction

This thesis is concerned with conceptual modelling: "Conceptual modelling is the activity of formally describing some aspects of the physical and social world around us for purposes of understanding and communication" (Mylopoulos, 1992). Understanding and describing the real world is the first step in the information system analysis and design (ISAD) process. The result is the *conceptual model* that is used as an input to the design of the information system (IS) to be constructed.

For the IS design phase the use of object-oriented techniques is well accepted. UML (OMG, 1999) has become widely used as a way to describe elements of an information system but is not specifically limited to this. On the other hand, there is no generally accepted language for conceptual modelling of the real world. Clearly, using the same modelling language for analysis and design has the potential advantage of eliminating confusion and translation problems. Hence, the central question addressed by this research is

> *Can object-oriented modelling languages, specifically UML, be used for conceptual modelling and in what way should they be used?*

To address this question, we propose a two step approach. The main emphasis of the thesis is the first step (Sec. 2) involving the theoretical evaluation of the object-oriented approach and UML, leading to predictions and guidelines on how to use object-oriented techniques and UML specifically for conceptual modelling. In a second step these guidelines will be empirically tested (Sec. 3). We close with a discussion of contributions (Sec. 4) and future work (Sec. 5).

2 Theoretical Development

In this research, we propose to examine the usability of UML as a language for describing the real world by mapping its constructs to a set of real-world concepts, that is, to an *ontology.* This mapping will provide real-world semantics to UML constructs originally introduced to model IS elements. Our theoretical analysis rests on three foundations:

- The BWW-Ontology
- Ontological Evaluation
- Use of Meta-Models

2.1 The BWW-Ontology

In order to describe the real world system in a model, we must specify what exists in this world. Ontology is "that branch of philosophy which deals with the order and structure of reality in the broadest sense possible" (Angeles, 1981). A specific ontology makes assumptions about what exists and how things behave. We choose the BWW-Ontology, Bunge's work (Bunge, 1977) as adapted for purposes of IS analysis by Wand and Weber (1989, 1990, 1993), for its formalization and prior successful application to the analysis of IS modelling languages (see Sec. 2.4). Following is a brief summary of ontological concepts.

The world is made up of *substantial things* that physically exist in the world. A thing possesses *properties* which are either *intrinsic*, e.g. color, or *mutual*, e.g. distance between two things. Things can combine to form a *composite thing* which must possess *emergent properties* that are not inherited from the parts. Things can change, but cannot be destroyed or created. Change can be qualitative, which means a property is lost or acquired (usually through composition or interaction), or it may be quantitative, in which case a property's value is changed.

A *law* is a relationship between properties. In particular, a law can be specified in terms of *precedence of properties*: The properties A and B are lawfully related, iff whenever a thing possesses A, it also possesses B ("B precedes A").

A BWW-natural kind is the set of things that have two or more common properties and a *natural kind* is the set of things that have two or more common properties related by laws such as the set of red things whose color and weight are related by law. In our ontology, natural kinds are defined over an existing set of things. In this sense, the things are the primary construct, not the natural kind. It follows that there can be no natural kind without members.

Attributes represent properties of a thing as perceived by an observer. They can be thought of as functions of time, e.g. specifying the colour of thing A at time T. Such functions are called *state functions*. A state is a *complete* assignment of values to the state functions, i.e. values must be assigned to all attributes. The set of state functions used to describe the thing is called a *functional schema* or a *model* of the thing. Depending on which aspects one is interested in, there can be different models describing the same thing. A thing is always in a lawful state, one that is allowed by the laws by which it abides. A state may be stable or unstable. If a thing is in an unstable state, it will spontaneously undergo a series of state transitions until it reaches a stable state.

Interaction is defined through effects of one thing on the state changes of another and can give rise to *binding* mutual properties. *Non-binding* mutual properties are properties of non-interacting things.

2.2 Ontological Evaluation

This work follows the notion of *ontological expressiveness* of modelling languages (Wand and Weber, 1993) which attempts to map language constructs into ontological concepts and vice versa. We use this mapping to transfer specific assumptions about concepts from the chosen ontology to UML. Thus, they lead to modelling rules and guidelines on *how* to use UML to model real-world systems. Such rules and guidelines can serve to reduce semantic ambiguity in conceptual modelling (Wand *et al.*, 1999) but they might not be obvious or applicable when UML is used for IS design purposes only. It is important to note that such rules do not necessarily guide us how to perceive the world. Thus, we might suggest rules on how to model things and classes, but not on how to identify them. Furthermore, since various UML constructs might map into related ontological constructs, we can generate ontologically-based intra- and inter-diagram integrity rules to guide the construction of consistent conceptual models in UML that go beyond the current meta-model.

2.3 Use of Meta-Models

Our analysis is based on the UML meta-model as specified in the UML reference manual (OMG, 1999) and the object constraint language OCL, a part of UML. A meta-model is a description or a model of a model, i.e. it specifies the language elements and the valid combinations of language elements in a model.

The rules derived from the ontological mapping generally serve to specify ontologically meaningful combinations of language constructs in a model and can be formally described in two ways:

- Additions or changes to the UML meta model so that it reflects ontological concepts and ontological assumptions about the real world.

– Specification of constraints in OCL when the ontological assumptions can be expressed using the elements of the existing meta-model or the suggested alterations to the meta-model.

2.4 Prior Research

There have been a number of prior studies concerned with the semantics of UML and the use of ontology in IS development. However, these two streams of research have had little commonalities. The former research aims at formalization of UML in mathematical languages or existing formal languages such as Z. [1] All of these works differ from our proposal in that they attempt to define semantics through internal consistency and coherence, not by relationships to the real world.

Another stream of research deals with the analysis of IS modelling languages in ontological terms (Wand and Weber, 1989, 1990, 1993; Parsons and Wand, 1997; Wand *et al.*, 1999; Opdahl and Henderson-Sellers, 1999; Opdahl *et al.*, 1999; Opdahl and Henderson-Sellers, 2001; Green and Rosemann, 2000) but lacks formalization and has mostly been applied to traditional languages such as data flow diagrams and entity-relationship diagrams. Since UML is the de-facto standard for modern IS design, this lends added importance to this study.

This work also builds on the formal syntax and formal semantics of the UML meta-model and the object constraint language, attempting to bridge the two diverse streams of research. It is the first study to employ formal descriptions of the rules resulting from an ontological analysis.

2.5 Theoretical Results

Rather than provide a complete analysis, what follows is a brief overview over the main results into static structure aspects of UML (Evermann and Wand, 2001b) followed by dynamic aspects and object interactions (Evermann and Wand, 2001a).

Objects are mapped into things, which leads to the rule that an object represents a substantial entity. Correspondingly, attributes may only be used to model properties, thus clearly differentiating among the two. Hence, entities such as "orders" or "jobs" which are often modelled as objects must be interpreted differently. They are mutual properties which should be modelled as associations.

Object creation and destruction have no direct equivalent, but an interpretation is provided through changes of natural kinds. This may occur through loss or acquisition of properties.

Moreover, classes cannot be abstract, they must possess at least one instance. Classes must define at least one attribute. Class attributes must be modeled as attributes of composites to make their character as emergent

---

[1] e.g. The Precise UML Group (e.g. Breu *et al.*, 1998; Evans *et al.*, 1999), http://www.cs.york.ac.uk/puml/publications.html

properties more obvious, e.g. modelling the average horsepower of cars as an attribute of a CarFleet composed of cars (see example below).

A state should be defined as a unique combination of attribute values, with consequences for both state charts and class definitions. Rules to ensure consistency between the two propose that sub-states require additional sets of attributes in the class definition. These must be disjunct for concurrent substates. Action states are interpreted as superstates of a number of substates and it is proposed that this should be made explicit in state charts.

Since messages and state charts necessarily express the same ontological feature, i.e. behaviour, we propose rules to ensure that state transitions correspond to methods and modify the appropriate attributes declared for the states. The specification of behaviour is fundamentally different in the object-approach than in our ontology; whereas the object-approach is prescriptive and procedural, the BWW-ontology is descriptive and declarative with behaviour being a secondary concept derived from laws.

Examining the intended meanings of the association construct shows that it is used to represent non-binding properties as well as potential future interactions between objects, ontologically expressed through laws that relate properties of things.

Presently we are working on the formalization of this discussion by suggesting additions to the UML meta-model and providing formal rules in OCL. Next we provide an example of the result of our rules.

Example We have mapped UML-classes to the ontological concept of a functional schema, a collection of state functions. Ontologically a functional schema is not a substantial thing in the world, and hence nothing that properties or attributes could be assigned to. Class attributes therefore are properties of the collection of individual things that comprise the class. We propose to model this collection explicitly and this leads us to propose the following rule:

Rule 1 *For a property representing a class attribute the composition of the class members must be shown explicitly and the emergent property modelled as an attribute of this composition.*

Consider the example shown in Fig. 1. What would normally be modelled as an attribute of the class, e.g. 'number of aircraft', must instead be modelled as an emergent property of the composition. In Fig. 1 (B) we make this composition explicit by modelling the aircraft fleet, to which we can attach the attribute 'number of aircraft'.

3 Empirical Corroboration

Section 2 outlined the development of real-world semantics and rules for using UML for conceptual modelling. We suggest that these are normative rules that should be followed in practice. To assess their validity, they must be tested empirically.

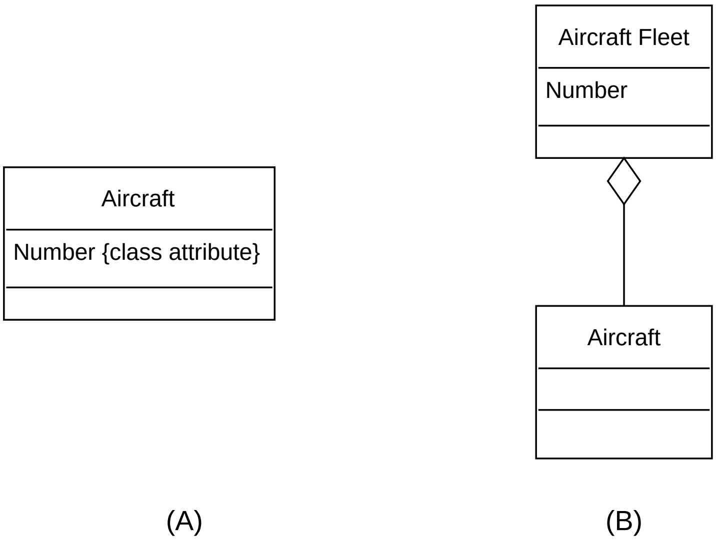

Fig. 1. Class attributes

One of the main purposes of a conceptual model is to serve as a *communication tool* among the participants of the ISAD process (Kung and Solvberg, 1986) to help arrive at a *common* understanding or agreement on what constitutes the problem domain (Schütte and Rotthowe, 1998).

A complete test of our theory involves examining the entire process from model construction to model interpretation. However since examining both of these aspects concurrently introduces confounds, these tasks must be examined separately (Gemino and Wand, 2001).

3.1 Model Construction

The semantics derived in Sec. 2 map language constructs to ontological concepts, thus telling the modeller which language element to use for which modelling situation. Moreover, the derived rules constrain the possible combinations of language elements. Both of these factors help guide the modelling process in the sense that the possible number of valid and allowable models becomes limited by reducing the semantic ambiguity (Wand and Weber, 1993). Hence, the assigned semantics will facilitate *common* understanding by making it more likely that different modellers will arrive at the same model. On the other hand, a language without such semantics allows for considerable ambiguity, the resolution of which will depend primarily on the modeller. Hence, different modellers are more likely to arrive at different models.

Hypothesis 1 *Following the suggested real-world semantics and rules will cause diagrams created by different modellers to be more similar to one another than diagrams created by modellers not following the suggested semantics and rules.*

3.2 Model Interpretation

We make the assumption that ontology is not only what exists in the world, but it also describes what humans believe exists in the world. [2]

A well accepted model of human memory and knowledge encoding is that of a semantic network (Collins and Quillian, 1969; Collins and Loftus, 1975; Anderson, 1995) with concepts as nodes being connected by associations as edges. A semantic network supports reasoning because the associations among concepts allow inferences to be drawn. A specific ontology becomes part of the semantic network. It defines nodes such as thing and property and their associations.

Learning and understanding is the integration of new concepts or new associations into such a network[3] (Anderson, 1995). If the model follows ontological semantics it will exhibit nodes such as things and properties that correspond to already existing ones in the interpreter's mental network. This will facilitate integration and make it possible for the model interpreter to use existing associations for reasoning beyond the information contained in the model. On the other hand, if the model does not follow such semantics it will exhibit model elements that contradict or are incompatible with the existing mental network, leading to improper or no integration at all. In this case, reasoning beyond the the information contained in the model will be more difficult.

Solving problems requires reasoning (Newell and Simon, 1972; Anderson, 1995). Hence, we propose that problem solving can measure the amount of reasoning that a given mental network supports. It is thus a measure of domain understanding (Gemino and Wand, 2001).

Hypothesis 2 *Interpreting a diagram created according to the suggested semantics and rules will lead to better problem solving performance than interpretation of diagrams created without following these semantics and rules.*

3.3 Related Research

Early work on examining the performance of different languages lacked underlying theory (Yadav *et al.*, 1999; Batra *et al.*, 1990), making the interpretation of the results difficult.

Bodart *et al.* (2001) have used the notions of conceptual networks and spreading activation of recall to argue for the use of sub-typing and against the use of optional properties in modelling. Kim *et al.* (2000)'s examination of the integration of multiple diagrams is particularly relevant as it is related to inter-diagram consistency that is one of the outcomes of our

---

[2] Being situated in the world, humans are continuous problem solvers (Newell and Simon, 1972) and hence require mental models and theories of the world. Thus, over time the mental model will come to correspond with the perceived real world and ontology becomes what we believe to exist in the world.

[3] Or the strengthening or weakening of associations.

analysis. It supports the theory that contextual information beyond the diagrams themselves, which an ontological model provides, can help diagram integration.

The cognitive fit model (Sinha and Vessey, 1992; Agarwal *et al.*, 1996, 1999) suggests that when the representation of the problem solving task matches the representation of the problem better task performance results. The cognitive fit model does not contain constructs reflecting model comprehension or understanding. Problem solving performance is seen as the dependent variable, not a measure of it.

Gemino and Wand (2001) use Mayer's model of learning (Mayer, 1989) which suggests that learning outcome is determined by the learning processes, which in turn depend on the learning material, the instructional method and the learner characteristic. Hence, *domain understanding* should be measured as the dependent variable, not recall or comprehension.

3.4 Experimental Design

We propose to investigate the research hypothesis using experimental techniques for reasons of internal validity[4].

3.4.1 Model Construction

Subjects While the target population are system analysts, we choose our sample frame as the set of undergraduate students who have successfully completed a modelling course involving UML. The very lack of experience that sets subjects apart from the target population also enables them to more easily integrate the augmented semantics into their modelling. For professional expert modellers, it could be very hard to overcome their experience and accept new semantics for UML.

Procedure Subjects will be randomly assigned to two treatment conditions. Condition A subjects will receive a short manual of UML to refresh their memory. The manual will be removed and subjects will be given a description of some real-world situation. They will be asked to model this description using class diagrams and state charts. Our ontological interpretation indicates that constructs in these diagrams are highly inter-dependent.

In addition to the short UML manual, condition B subjects will be instructed in the suggested semantics and given a rule summary. The instruction time for group B subjects is the same as for group A subjects [5]. The instructions only relate to those constructs that are relevant to class and state diagrams. Again, the manual will be removed for the test, but the rule summary will be left for the subject to reference. This is done to reduce the cognitive effort associated with memorizing a large number of rules. The remainder will be as in condition A.

---

[4] I.e. the ability to control for external influences.

[5] Otherwise, group A subjects would have less time for elaboration and familiarization with the techniques, leading to a possible confound.

Measures Three expert UML users will be asked to model a base solution using the proposed semantics and rules. The experts will then rate the similarity of each diagram against this base diagram. Condition A models are expected to show generally a lower similarity score and a wider range of scores than models of condition B. Inter-rater agreement will be measured and should be > 0.8.

3.4.2 Model Interpretation

Tested rules Since there are a large number of rules, our empirical verification for the interpretation task will test two representative rules. We follow Gemino and Wand (2001) in their suggestion to use problem solving tasks as a measure for domain understanding and use their instrument as well as their procedure as a guide.

Subjects End user support is an important factor in systems development project success, hence end users form the target population for this study. The sample frame of this study will be the same as for the model constructions study with the same limitations to external validity.

Procedure The experimental design is a 2×2 between-subjects design, where the first factor is the chosen rule and the second factor the diagram type, i.e. a diagram that follows the rule vs. one that does not. Subjects will be assigned randomly to one of the four groups and given a diagram to study. Comprehension questions will be asked with the diagram present. This is to engage the students thoroughly in the study of the diagram and not simply memorize it visually, enhancing understanding (Anderson, 1995). The diagram will then be removed and the set of problem solving questions administered.

Measures The number of problem solving questions answered correctly will be higher for subjects that interpreted diagrams drawn by following the suggested semantics, irrespective of which rule was tested.

4 Expected Contributions

The theoretical study is the first to assign firmly grounded real-world semantics to UML. Given that it is grounded in ontology, it is the first study to suggest real-world semantics for use in conceptual modelling. Hence, it is a first important step towards a commonly accepted meaning for UML constructs.

For practitioners, the theoretical analysis proposes a practical and detailled guide to using UML in IS analysis. This enables all participants to realize the full potential of conceptual models as a communication medium in reasoning about the problem domain.

The empirical validation provides a first measure of model convergence among modellers. Whereas previous works measured correctness, this is not appropriate in most situations (Schütte, 1998; Schütte and Rotthowe, 1998). Furthermore, we suggest a theoretically grounded process by which certain features of a modelling language influence the performance of that language for conceptual modelling whereas previous empirical work was often conducted without underlying theory.

We show as an example how the verification of theoretically developed modelling rules can proceed. By reusing the instrument of Gemino and Wand (2001) and Bodart *et al.* (2001) we help build up cumulative knowledge of the field.

For the practitioner, the empirical test serves to corroborate the theoretical claims and shows that the benefits can be realized in practical modelling situation and system analysis projects.

5 Future Work

The major UML and object-oriented constructs have been examined and mapped to ontological concepts (Evermann and Wand, 2001a,b). The remaining work involves formalizing the suggested rules in OCL.

For the empirical corroboration of our main results we have developed the initial empirical model to test. The experimental design of the thesis will be developed in greater detail. Appropriate experimental materials such as situational descriptions, diagrams and problem solving questions must be devised and the experiments conducted.

6 Conclusion

The main goal of this work is to assign real-world semantics to object oriented modelling language constructs so they can be used for conceptual modelling. We have done this by suggesting a mapping from UML constructs to a specific ontological model. Based on the ontologically derived semantics, we were able to transfer ontological assumptions to UML. From this, we identified rules that we suggest should be followed when applying the object-oriented approach and UML in particular to conceptual modelling of real world domains. These rules can guide the use of UML constructs in specific situations and can help ensure that object-oriented models will have a meaningful ontological interpretation.

The purpose of the empirical testing of our results is to support our hypotheses that the semantics we propose lead to increased consensus among modellers and better domain understanding among interpreters. This will lend additional support to the method of ontological examination and analysis of languages.

References

Agarwal, R., Sinha, A. P., and Tanniru, M. (1996). Cognitive fit in requirements modeling: A study of object and process methodologies. *Journal of Management Information Systems*, **13**(2), 137–162.

Agarwal, R., De, P., and Sinha, A. P. (1999). Comprehending object and process models: An empirical study. *IEEE Transactions of Software Engineering*, **25**(4), 541–556.

Anderson, J. R. (1995). *Cognitive Psychology and its implications.* W.H. Freeman and Company, New York, NY., fourth edition.

Angeles, P. (1981). *Dictionary of Philosophy.* Harper Perennial, New York, NY.

Batra, D., Hoffer, J. A., and Bostrom, R. P. (1990). Comparing representations with relational and eer models. *Communications of the ACM*, **33**(2), 126–139.

Bodart, F., Sim, M., Patel, A., and Weber, R. (2001). Should optional properties be used in conceptual modelling? A theory and three empirical tests. *Information Systems Research*, **12**. Publication forthcoming.

Breu, R., Grosu, R., Hofmann, C., Huber, F., Krüger, I., Rumpe, B., Schmidt, M., and Schwerin, W. (1998). Exemplary and complete object interaction descriptions. *Computer Standards and Interfaces*, **19**, 335–345.

Bunge, M. A. (1977). *Ontology I: The Furniture of the World*, volume 3 of *Treatise On Basic Philosophy.* D. Reidel Publishing Company, Dordrecht, Holland.

Collins, A. and Loftus, E. (1975). A spreading activation theory of semantic processing. *Psychological Review*, **82**, 407–428.

Collins, A. and Quillian, M. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, **8**, 240–248.

Evans, A., France, R., Lano, K., and Rumpe, B. (1999). The UML as a formal modeling notation. In *The Unified Modeling Language UML'98: Beyond the notation, First International Workshop, Mulhouse, France, June 1998.*

Evermann, J. and Wand, Y. (2001a). An ontological examination of object interaction in conceptual modeling. In *Proceedings of the Workshop on Information Technologies and Systems WITS'01, New Orleans, December 15-16, 2001.*

Evermann, J. and Wand, Y. (2001b). Towards ontologically based semantics for UML constructs. In H. Kunii, S. Jajodia, and A. Solvberg, editors, *Proceedings of the 20th International Conference on Conceptual Modeling, Yokohama, Japan, Nov. 27-30, 2001, ER 2001.*

Gemino, A. and Wand, Y. (2001). Comparing object oriented with structured analysis techniques in conceptual modeling. Draft Manuscript.

Green, P. and Rosemann, M. (2000). Ontological analysis of integrated process modelling. *Information Systems*, **25**(2).

Kim, J., Hahn, J., and Hahn, H. (2000). How do we understand a system with (so) many diagrams? cognitive integration processes in diagrammatic reasoning. *Information Systems Research*, **11**(3), 284–303.

Kung, C. and Solvberg, A. (1986). Activity modelling and behaviour modelling. In T. Olle, H. Sol, and A. Verrijn-Stuart, editors, *Information System Design Methodologies: Improving the Practrice.* Amsterdam.

Mayer, R. E. (1989). Models for understanding. *Review of Educational Research*, **59**(1), 43–64.

Mylopoulos, J. (1992). Conceptual modeling and telos. In P. Locuopoulos and R. Zicari, editors, *Conceptual Modeling, Databases and Cases.* John Wiley & Sons, Inc, New York et. al.

Newell, A. and Simon, H. A. (1972). *Human Problem Solving.* Prentice Hall, Englewood Cliffs, NJ.

OMG (1999). *The Unified Modelling Language Specification. Version 1.3.* OMG.

Opdahl, A. and Henderson-Sellers, B. (1999). Evaluating and improving OO modelling languages using the BWW-model. In *Proceedings of the Information Systems Foundation Workshop 1999.* www.comp.mq.edu.au/isf99/Opdahl.htm.

Opdahl, A., Henderson-Sellers, B., and Barbier, F. (1999). An ontological evaluation of the OML metamodel. In E. Falkenberg and K. Lyytinen, editors, *Information System Concepts: An Integrated Discipline Emerging.* IFIP/Kluwer.

Opdahl, A. L. and Henderson-Sellers, B. (2001). Grounding the OML metamodel in ontology. *The Journal of Systems and Software*, **57**, 119–143.

Parsons, J. and Wand, Y. (1997). Using objects for systems analysis. *Communications of the ACM*, **40**(12), 104–110.

Schütte, R. (1998). *Grundsätze ordnungsmäßiger Referenzmodellierung.* Gabler Verlag, Wiesbaden, Germany. Also PhD Dissertation, University of Münster, Germany.

Schütte, R. and Rotthowe, T. (1998). The guidelines of modeling - an approach to enhance the quality in information models. In *Proceedings of the ER'98 Conference*, pages 240–254.

Sinha, A. P. and Vessey, I. (1992). Cognitive fit: An emprirical study of recursion and iteration. *IEEE Transactions of Software Engineering*, **18**(5), 368–379.

Wand, Y. and Weber, R. (1989). An ontological evaluation of systems analysis and design methods. In E. Falkenberg and P. Lingreen, editors, *Information System Concepts: An In-Depth Analysis.* Elsevier Science Publishers B.V., North-Holland.

Wand, Y. and Weber, R. (1990). Mario Bunge's ontology as a formal foundation for information systems concepts. In P. Weingartner and G. Dorn, editors, *Studies on Mario Bunge's Treatise.* Rodopi, Atlanta.

Wand, Y. and Weber, R. (1993). On the ontological expressiveness of information systems analysis and design grammars. *Journal of Information Systems*, (3), 217–237.

Wand, Y., Storey, V. C., and Weber, R. (1999). An ontological analysis of the relationship construct in conceptual modeling. *ACM Transactions on Database Systems*, **24**(4), 494–528.

Yadav, S. B., Bravoco, R. R., Chatfield, A. T., and Rajkumar, T. M. (1999). Comparison of analysis techniques for information requirement determination. *Communications of the ACM*, **31**(9), 1090–1097.