

An Ontology-Based Evaluation of Process Modeling with Petri Nets

Jan Recker
Business Process Management Group
Queensland University of Technology
Brisbane, Australia
j.recker@qut.edu.au

Marta Indulska
UQ Business School
The University of Queensland
Brisbane, Australia
m.indulska@business.uq.edu.au

Abstract: Selecting an appropriate business process modeling technique forms an important task within the methodological challenges of business process management and workflow engineering projects. Many of the available techniques have been developed on the basis of Petri nets, which is a popular modeling technique for workflow-oriented applications. Despite the popularity of Petri nets, however, a comprehensive evaluation of this standard notation is still required. Such an evaluation would afford a deeper understanding of the capabilities and shortcomings of this fundamental process modeling technique. This paper presents the first contribution towards a theoretically sound analysis of the representational capabilities of Petri nets. Based on an established methodology, we conducted a representational analysis of Petri nets using a representation model based on the Bunge ontology. Through this work, we identify a number of issues related to the practice of process modeling with Petri nets in contemporary process management initiatives. Our findings contribute to the ongoing revision and extension of process modeling techniques based on Petri nets and as such may lead to more mature solutions to business process modeling and management.

Introduction

Business Process Management (BPM) is continuously ranked as a top business priority and building business process capabilities is still seen as a major challenge for senior executives within the coming years [1]. BPM is a structured, coherent and consistent way of understanding, documenting, modeling, analyzing, simulating, executing and continuously changing end-to-end business processes and all involved resources in light of their contribution to business performance [2]. Anecdotal evidence suggests increasing demand for a more disciplined approach towards BPM, which motivates many organizations to make substantial investments in BPM-related initiatives. Indeed, a recent study of 348 organizations found that 58% of the organizations spent up to US\$500,000 on BPM in 2005 [3]. Roughly 34% of organizations spent between US\$500,000 and US\$4 million, with 53% of organizations stating that their BPM efforts would be increasing in 2006. This continual increase in BPM interest in turn has triggered significant related

academic and commercial work aiming towards advanced business process management solutions, for example [4]. One prominent example in this context is the increased popularity of business process modeling [5]. Not surprisingly, a wide range of process modeling techniques has been proposed over time. Nowadays, techniques range from simple flowcharting techniques to business modeling approaches such as Event-driven Process Chains and software engineering-driven techniques such as UML, to formalized and executable techniques such as BPEL. In the process of developing and extending new process modeling techniques, a number of authors, for example [6-8], argue for the use of *Petri nets* [9, 10] as a formal foundation (a meta process modeling technique, one could say). Their argument is that most available techniques for process design are informal, in the sense that the diagrams used have no formally defined semantics [11]. This situation in turn leads to shifting interpretations depending on modeler, application domain, and characteristics of the business processes at hand. Examples of such informal techniques are ISAC, DFD, SADT, and IDEF. Since these techniques are inadequate for directly driving process-aware information systems, Petri nets have been proposed as a more precise way of modeling processes. Petri nets not only provide an abundance of analysis techniques to evaluate process modeling languages, for example [7, 12] but also have been used as a technique from which new process modeling techniques, such as LOOPN++ [13] or YAWL [14], have been derived.

The relevance of Petri nets to process modeling can hence be seen as undisputed. Surprisingly, however, an evaluation of whether or not Petri nets is actually a 'good' process modeling technique, on which the development of more advanced techniques can be based, has not yet been investigated. The goodness of a modeling technique can be understood as its capability to facilitate complete descriptions of relevant real-world domains while at the same time being clear in the usage of the language constructs provided [15]. It then follows that an analysis of the strengths and weaknesses of existing modeling approaches can be used as the basis for developing new and improved techniques [16].

Accordingly, our *research objective* is to study the representational capabilities of the Petri nets process modeling technique in order to ultimately be able to gauge the capabilities of process modeling techniques that have been using Petri nets as their foundation. One well established method of evaluating the representational capability of process modeling techniques that purport to capture real-world scenarios is through the use of a model of representation based on a theory of ontology. A promising candidate is the Bunge-Wand-Weber (BWW) representation model [15, 17, 18] that has been derived from an ontology proposed by Bunge [19]. The selection of this ontology-based model can be justified on at least three counts. First, unlike many other foundational theories based on ontology, the BWW model has been derived with the Information Systems discipline in mind [20]. Second, while the BWW model does not denote a unique case of IS-specific ontology, refer, for instance, to [21], there is an established track record and demonstrated usefulness of representational analyses of modeling techniques using the BWW representation model [22]. Third, the BWW model officiates as an upper ontology for the modeling of Information Systems [23], and the foundational character and comprehensive scope allows for wide applicability.

The *aim of this paper* then is to (a) analyze, using an ontology-based theory of representation, the capacity of Petri nets to provide faithful representations of relevant real-world domains and (b) develop propositions about the modeling

strengths and weaknesses of Petri nets for modeling business processes. We proceed as follows. The next section gives an overview of the theory and model that underlies our research. We also recapitulate previous research on representational analyses of process modeling techniques, as well as Petri nets. We then present our analysis of Petri nets and discuss our findings and propositions. The paper concludes with a summary of our contributions, a discussion of research limitations and an outlook on future work.

Background and Related Work

The BWW Theory of Representation

The theoretical foundation and rigor of this type of research is derived from the selected foundational ontology. Research based on the BWW models can be traced back to the comprehensive and detailed work of Mario Bunge [19] and its accomplishments. *Ontology* studies the nature of the world and attempts to organize and describe what exists in reality, in terms of the properties of, the structure of, and the interactions between real-world things [24]. The initial and ongoing development of such ontologies, and the comparison of different foundational ontologies, is a challenging task that is located in the discipline of philosophy and has its roots in Aristotle's [25] foundational work on metaphysics.

It was the adoption of Bunge's ontology by Wand and Weber [15, 17, 18] into a model of representation that facilitated the wider uptake of this theoretical model within the Information Systems community. As over the last few decades numerous conceptual modeling techniques, used to define requirements for building information systems, have emerged with no consistent theoretical foundation underlying their conception, Wand and Weber were concerned that this situation would result in the development of information systems that were unable to completely capture important aspects of the real world. Wand and Weber suggest that ontology can be used to help define and build information systems that contain the necessary representations of real world constructs. Hence, they adapted and refined the ontology of Bunge into a set of models for the evaluation of information modeling techniques (and the scripts prepared using such techniques) based on the premise that computerized information systems are representations of real world systems.

While the Bunge-Wand-Weber set of models actually comprises three models [17, 26], *viz.*, the representation model, the state-tracking model and the decomposition model, it is the representation model that is typically used for the evaluation of conceptual modeling techniques. The application of the representation model to Information Systems foundations has been referred to by a number of researchers [22]. Some minor alterations have been performed over the years by Wand and Weber [15, 17] and Weber [26], but the current key constructs of the BWW model can be grouped into the following clusters: things including properties and types of things; states assumed by things; events and transformations occurring on things; and systems structured around things. For

more information on the BWW set of models and the constructs specified within refer, for example, to [26].

The process of using the BWW representation model as a type of reference benchmark for the evaluation of the representational capabilities of a modeling technique forms the core of the research method of *representational analysis* [27]. In this process, the constructs of the BWW representation model (*e.g.*, thing, event, transformation) are compared with the language constructs of the process modeling technique (*e.g.* event, activity, actor). The basic assumption is that any deviation from a 1-1 relationship between the corresponding constructs in the representation model and the modeling technique leads to a situation of representational deficiency and/or ambiguity in the use of the technique potentially causing confusion to the end users.

A representational analysis, in principle, can also be used to compare the BWW representation model against a composition of language constructs in a modeling technique (*e.g.*, does the composition event-function-event have an ontologically well-formed meaning?). The composition of language constructs to ontologically well-formed 'chunks' of model semantics has been labelled a production rule, for example see [15]. Typically, however, representational analyses consider the case where *one* construct in a modeling language can or cannot be mapped to a number of representation model constructs. Accordingly, in our analysis we restrict our analysis to these 1:1 mappings.

The set of BWW constructs is well defined in various languages. Wand and Weber [18], for instance, use set theory to formalize the set of constructs, and Rosemann and Green [28] developed a semi-formal description of the set of BWW constructs by means of a meta model using the Extended Entity-Relationship (EER) modeling notation [29]. The process of applying these constructs as part of a representational analysis, however, is less specified. It was only recently that more advanced procedural models have been proposed [30] that guide researchers through the process of comparing the representational model with the selected modeling technique.

Generally, representational analysis focuses on mapping relationships that are not 1:1. Such cases are classified as theoretical, *i.e.*, potential, representational shortcomings. These undesirable situations can be further categorized into the following four types, as shown in Figure 1 [26]:

- *construct overload* describes a situation in which a construct in the process modeling technique represents two or more representation model constructs (m:1 relationship);
- *construct redundancy* is the opposite case, *i.e.* one construct in the representation model is depicted by two or more constructs in the process modeling technique (1:m relationship);
- *construct excess* is the case in which at least one construct in the process modeling technique does not map to any construct in the representation model (0:1 relationship); and
- *construct deficit* describes the case in which at least one construct in the representation model does not map to any construct in the process modeling technique (1:0 relationship).

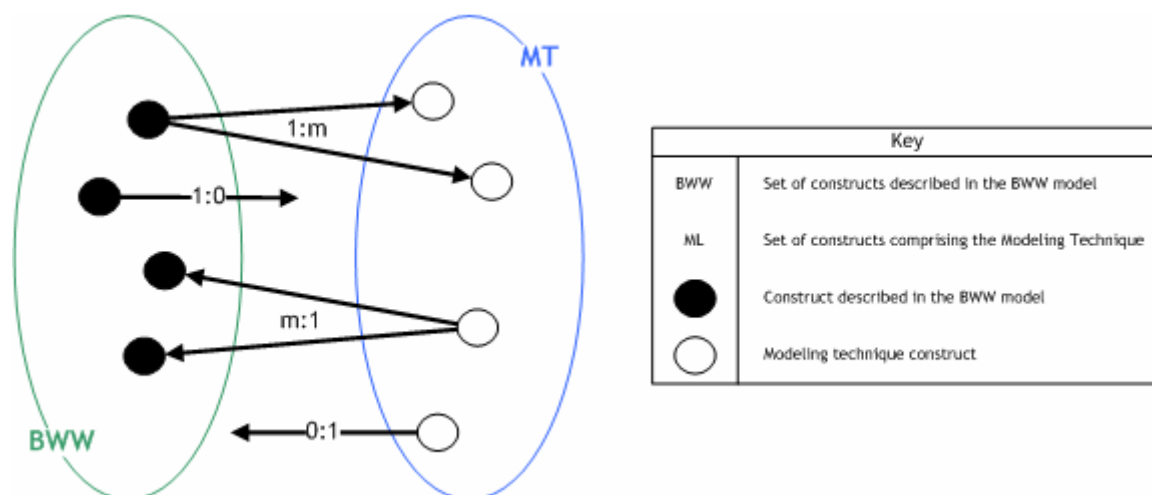


Figure 1: Types of potential representational shortcomings

Based on these four types of undesirable situation, it is possible to make predictions as to the representational capabilities of a process modeling language for providing *complete* and *clear* representations of the domain being modeled [26]. In particular, if a process modeling technique provides constructs for each element of the representation model, *i.e.*, construct deficit is not present, it is regarded as *ontologically complete*. In turn, the *ontological clarity* of a process modeling technique can be measured by the degrees of construct overload, construct redundancy, and construct excess.

Such issues form important input for further revisions and improvements of existing process modeling techniques. In this phase, it is important to communicate the theoretical and empirical research outcomes back to the developers of the modeling technique or tool providers. The assumption of this type of research is that identified and communicated issues related to a process modeling technique have the potential to guide revisions of these techniques and ultimately lead to an increased quality of process models. The availability of improved models provides the involved stakeholders with better opportunities to achieve the goals underlying the process modeling initiative. From a research viewpoint, this is the phase in which frameworks for the evaluations of the quality of a process model can be utilized, such as, for instance, the semiotic quality framework [31].

BWW-related Work

The BWW model has over the years reached a significant level of maturity, adoption and dissemination, and has been used in over thirty research projects for the evaluation of different modeling grammars, including data models [15], schema modeling [32], object-oriented models [33] and reference models [34]. It also has a strong track record in the area of process modeling, with contributions coming from various researchers. In this section, we briefly summarize those BWW related studies that focus specifically on process modeling techniques.

Keen and Lakos [35] determined essential features for a process modeling scheme by evaluating six process modeling techniques in a historical sequence by using the BWW representation model. Their evaluation is restricted to the assessment of the

ontological completeness of the considered techniques. From their analysis, Keen and Lakos concluded that, in general, the BWW representation model facilitates the interpretation and comparison of process modeling techniques. They propose the BWW constructs of system, system composition, system structure, system environment, transformation, and coupling to be essential process modeling technique requirements. However, we found that these findings are not entirely reflected in the leading process modeling techniques [36].

Green and Rosemann analyzed the Event-driven Process Chain (EPC) notation [37] with the help of the BWW model both analytically [38] and empirically [39], assessing both ontological completeness and clarity. Empirically confirmed shortcomings were found in the EPC notation with regard to the representation of real world objects and business rules, and in the thorough demarcation of the analyzed process.

Green *et al.* [40, 41] compared different modeling standards for enterprise system interoperability, including Business Process Execution Language for Web Services v1.1 (BPEL4WS) [42], Business Process Modeling Language v1.0 (BPML) [43], Web Service Choreography Interface v1.0 (WSCI) [44], and the eXtensible Markup Language Business Process Specification Schema (ebXML BPSS) v1.01 [45]. These standards, which proclaim to allow for specification of intra- and inter-organizational business processes, have been analyzed in terms of their ontological completeness. The study found that ebXML provides a wider range of language constructs for specification requirements than other techniques, indicated through its comparatively high degree of ontological completeness. In addition, a minimal ontological overlap (MOO) analysis [17, 26] was conducted in order to determine the set of modeling standards with a minimum number of overlapping constructs but with maximal ontological completeness (MOC). The study identified two sets of standards that together allow for the most expressive power with the least overlap of constructs, *viz.*, ebXML and BPEL4WS, and, ebXML and WSCI.

Recker *et al.* [46] used representational analysis to identify shortcomings in the Business Process Modeling Notation (BPMN) v1.0 [47] from the viewpoint of both clarity and completeness, resulting in the finding that there would appear to be representational shortcomings in BPMN, for example, in the modeling of business rules, and the usage of the Lane and Pool constructs [48].

Most recently, we consolidated the previous representational analyses of popular modeling techniques and supplemented this collection of research with our own analyses of Petri nets [36]. After normalizing the various analyses so that they can be compared, we found that there is a trend of increasing ontological completeness of process modeling methods over time, which, however, appears to be afforded by a decreasing ontological clarity of the methods. As part of the study, we briefly discussed the ontological completeness of Petri nets in comparison to other process modeling techniques. The present work builds on the representation mapping performed in [36] and extends the analysis of Petri nets to establish a comprehensive understanding of the extent of ontological completeness and clarity that Petri nets exhibit. Also, in the present work we show and discuss how the identified instances of representational deficiencies could potentially lead to issues in the actual use of the technique.

Introduction to Petri nets and related work

Petri nets have been around since the sixties [9]. They were originally invented as a graphical and mathematical technique for describing and studying information processing systems that are characterized as being concurrent, asynchronous, distributed, parallel, nondeterministic and/or stochastic [10]. As a graphical technique, Petri nets can be used as a visual aid for communicating between different communities (for instance, between business analysts and systems designers) while, as a mathematical technique, Petri nets provide a means for setting up state equations, algebraic equations and other models governing the behaviour of process-oriented information systems.

Petri nets were first used in application areas of office information systems. Authors such as Ellis [49], Holt [50] and Zisman [51] worked on process model-driven office information systems that used Petri nets for describing office procedures. Since then, Petri nets have been applied for a number of purposes, such as developing a set of workflow patterns for the analysis of process-aware information systems [52], for task planning research [53] or for manufacturing [8].

In the area of process modeling with Petri nets, at least three reasons are suggested for using Petri nets as a process modeling technique [6]:

1. formal semantics despite the graphical nature;
2. state-based instead of event-based; and
3. abundance of analysis techniques.

However, only little research has actually investigated the ‘goodness’ of Petri nets for popular process modeling applications such as process improvement, strategic alignment, knowledge management, etc. Indeed it is this gap in research, as well as the apparent foundation for process modeling that Petri nets play, that motivates our research on their representational capabilities.

While Petri nets have been subjected to some refinement and extensions, eventually incorporating aspects such as color [54] and time [55] in order to improve expressiveness, the basic concepts of Petri nets still remain the same; A Petri net is a directed graph together with an initial state called the *initial marking*. The underlying graph of a Petri net is a directed, weighted, bipartite graph consisting of two nodes, *viz.*, *places* and *transitions*, which are connected via (weighted) *arcs*. Graphically, places are depicted as circles while transitions are marked as boxes. A marking assigns to each place a nonnegative integer k , meaning that the place is marked with k *tokens* (represented as black dots). Places have a finite *place capacity* restricting the maximum number of tokens they can hold. A transition has a certain number of input and output places representing pre- and post-conditions for the firing of the transition. Processes can be modeled and simulated by changing the (initial) marking of a state according to the following firing rules:

1. a transition is enabled if each of its input places is marked with as many tokens as the weight of the arc leading to the transition described;
2. an enabled transition may or may not fire; and
3. a firing of an enabled transition removes the tokens from the input places in accordance to the arc weight and adds tokens to each output place in accordance to the weight(s) of the arc(s) leading from the transition to its output places.

Figure 2 gives an example of a Petri net illustrating the process of the well-known chemical reaction $2\text{H}_2 + \text{O}_2 \rightarrow 2\text{H}_2\text{O}$. Two tokens in each input place of the transition in (a) show that two units each of H_2 and O_2 are available, enabling the transition to fire. After firing, the marking will change to the one in (b) in which the transition is no longer enabled.

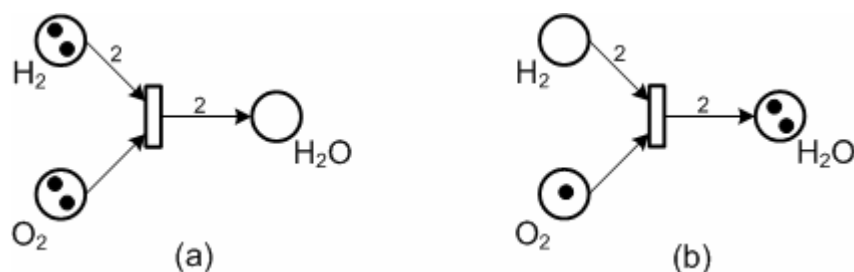


Figure 2: Example of a Petri net

The basic concepts of Petri nets are, as our analysis will show, quite generic. They allow for a wide range of process modeling-related semantics. Correspondingly, the high level of conceptual abstraction that Petri nets are afforded have led to the emergence of at least two “schools” of Petri net interpretation. Some proponents of Petri nets, especially in the early years, have interpreted places as activities with transitions being triggers or results of these activities (see, for example, [56]). A second, more recent and larger school interprets transitions as activities (see, for example, [6, 14, 52, 57]). Given the more widespread adoption of the latter school of thought we will follow their line of thinking in our analysis.

Also, there are a number of concepts associated with compositions of the basic language constructs of Petri nets. For instance, a deadlock marks a situation in which the flow of a marking is stopped due to an unavailability of the required number of input tokens. The deadlock concept is very important to Petri nets-based evaluation approaches [12, 57] but does not denote a language construct in itself (rather an inferential information that can be deduced from a corresponding arrangement of Petri nets constructs). Accordingly, such concepts are not a part of the Petri nets construct definition that we consider in our analysis.

An Ontology-based Representational Analysis of Petri nets

Research Design and Mapping Results

Generally, the focus of representational analyses is on the bi-directional comparison of constructs specified in the underlying theory of representation with the language constructs of the modeling technique. Such comparison leads to statements about the two situations of ontological completeness and clarity. In order to follow a rigorous approach towards evaluation, we followed an established methodology [30]. Specifically, our analysis was conducted in three steps. First, two researchers separately read the Petri net specification and mapped the Petri net constructs against BWW constructs in order to create individual first analysis drafts. Second, the researchers met to discuss and defend their mapping results, resulting in a jointly agreed second draft. Third, the jointly agreed second draft was discussed and refined in several meetings with two BWW experts outside the

research team. By reaching a consensus over the final mapping result we feel that we have significantly increased the objectivity and rigor in this type of research. Adopting this methodology has also allowed the derivation of agreement statistics between the individual researchers. A raw agreement percentage [58] was calculated to be .69 in the first round and improved to .92 in the second round, thereby indicating the improved reliability through the joint mapping. This second draft was refined with the external experts until agreement amongst all researchers reached 1.00.

The analysis of the representation mapping found that Petri nets allow the representation of BWV real-world concepts such as *thing, class, state law, lawful state space, event, transformation, stability condition* (part of lawful transformation), *acts-on, unstable state, internal event* and *well defined event*. The remainder of BWV constructs, however, remain without representation in the Petri nets technique. Table 1 summarizes the outcome of the refined analysis and also displays a brief reasoning behind the mappings of BWV constructs to Petri nets constructs. Stemming from this analysis, the discussion of the proposed strengths and weaknesses of Petri nets is presented in the following section.

BWV Construct	Construct Explanation	Petri Net Construct	Reasoning behind mapping
THING	Elementary unit in the BWV model. Real world is made up of things. Two or more things can be associated into a composite thing	Place	A place may represent an instance of a set of objects - for example, a specific corporate customer.
PROPERTY	Things possess properties. A property is modeled via a function which maps the thing into some value.		
in general	E.g., Attribute 'weight' represents a property that all humans possess.	-	N/A
in particular	E.g., The weight of a specific human being.	-	N/A
hereditary	A property of a composite thing which belongs to a component thing.	-	N/A
emergent	A property determined by a collection of properties of things within the composite thing.	-	N/A
intrinsic	Inherent properties of individual things.	-	N/A
mutual: non-binding	Properties shared by two or more things that do NOT make a difference to the things involved.	-	N/A

BWW Construct	Construct Explanation	Petri Net Construct	Reasoning behind mapping
mutual: binding	Properties shared by two or more things that DO make a difference to the things involved.	-	N/A
attributes	Names used to represent the properties of things.	-	N/A
CLASS	A set of things that can be defined via their possessing a single common property.	Place	A place may represent sets of objects - e.g., all corporate customers.
KIND	A set of things that can be defined via their possessing two or more common properties.	-	N/A
STATE	The vector of values of all property functions of a thing; that is the values of attributes we have assigned to a thing.	Place, Initial marking, Token	A place indicates a state of values. These values are represented by a number of tokens. An initial marking is a starting state. A token represents a certain value state of an object.
CONCEIVABLE STATE SPACE	The set of all possible states of a thing each of which represents some combination of possible values for the individual state functions.	-	N/A
STATE LAW	Restricts the values of the properties of a thing to a subset that is deemed lawful because of natural or human laws.	Place capacity	Place capacity defines a rule on an object - e.g., there can be a maximum of 1000 corporate customers.
LAWFUL STATE SPACE	The set of states of a thing that comply with the state laws of the thing. Usually a proper subset of the conceivable state space.	Place capacity	Place capacity determines the set of states that are allowable by defining possible values.
EVENT	An event that arises in a thing, subsystem or system by virtue of the action of some thing in the environment of the thing, subsystem or system.	Transition	A transition leads to an event, changing the state values of an object.
CONCEIVABLE EVENT SPACE	The set of all possible events that can occur in a thing.	-	N/A

BWW Construct	Construct Explanation	Petri Net Construct	Reasoning behind mapping
LAWFUL EVENT SPACE	Set of all events in a thing which are lawful.	-	N/A
TRANSFORMATION	A mapping from one state to another.	Transition	Transition changes a state to another.
LAWFUL TRANSFORMATION	Defines which mappings between two states of a thing are lawful. Usually defined as the transformation from one lawful state space into another.	-	N/A
stability condition	Specifies the states that are allowable under the transformation law.	Arc weight	Arc weight defines how many tokens are required so that the transition can be executed.
corrective action	Specifies how the values of the property functions must change to provide a state acceptable under the transformation law.	-	N/A
HISTORY	Chronologically ordered states that a thing traverses in time are the history of a thing.	-	N/A
ACTS ON	A thing acts on another if its existence affects the history of another thing.	Arc	Arc define a relationship between things, also defining the direction of the relationship.
SYSTEM	A set of things is a system if for any bi-partitioning of the set there exist couplings between the things in the two subsets.	-	N/A
SYSTEM COMPOSITION	The set of all things in the system.	-	N/A
SYSTEM ENVIRONMENT	The set of things that are NOT in the system composition but that interact with things in the system.	-	N/A
SYSTEM STRUCTURE	The set of couplings that exist among the things in the system and among the things in the environment of the system.	-	N/A
SUBSYSTEM	A system whose composition and structure are subsets of the composition	-	N/A

BWW Construct	Construct Explanation	Petri Net Construct	Reasoning behind mapping
	and structure of another system.		
SYSTEM DECOMPOSITION	The 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 systems in the decomposition.	-	N/A
LEVEL STRUCTURE	Defines the partial order over the subsystems in a decomposition to show which subsystems are components of other subsystems or the system itself.	-	N/A
STABLE STATE	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 of a system.	-	N/A
UNSTABLE STATE	A state that will be changed into another state by virtue of the action of transformations in the system.	Place, Initial Marking, Token	An initial marking of a place triggers transitions and thus the whole process, resulting in a change of state. A token in a place also triggers transitions.
EXTERNAL EVENT	An event that arises in a thing, subsystem or system by virtue of the action of some thing in the environment of the thing, subsystem or system.	-	N/A
INTERNAL EVENT	An event that arises in a thing, subsystem or system by virtue of lawful transformations in the thing, subsystem or system.	Transition	A transition leads to an event changing the state values of an object.
WELL-DEFINED EVENT	An event in which the subsequent state can ALWAYS be predicted given that the prior state is known.	Transition	A transition leads to an event changing the state values of an object. The subsequent state can be predicted through the transition specification.
POORLY-DEFINED EVENT	An event in which the subsequent state CANNOT be predicted given that the prior state is known.	-	N/A
<i>EXCESS</i>	Constructs for which a 1-1 mapping could not be found.	-	N/A

Table 1. Summary of Petri nets representational mapping

Some words of caution ought to be said. First, as indicated earlier, we considered only the elementary graphical constructs of Petri nets in our mapping and not (potentially meaningful) combinations of these. Second, we did not consider rules that would apply to a given combination of Petri nets constructs (such as a firing rule, which would apply to a model fragment in which each input place to a transition is marked with the required number of tokens). Our exclusion of such rules rests on the observation that these rules specify the dynamic behaviour that would stem from the static description articulated in the model. A graphical model, however, by definition is static and hence we can only evaluate its capacity to be a faithful representation of a real-world domain based on its static properties.

Discussion and Propositions

Based on the representation mapping outcomes presented in Table 1, we are able to derive eleven propositions (see Table 2) that suggest potential shortcomings in the use of the Petri nets technique for process modeling. The first eight propositions relate to the earlier discussed concept of construct deficit. In other words, these propositions relate to situations in which no mapping was found from a BWW construct to a Petri nets construct, indicating that users of the Petri nets technique will be unable to represent some relevant concepts in the real world. Research shows that in such situations users find workarounds - employing an additional modeling technique, changing some of the constructs, etc - in order to increase their modeling capability with the technique in question [59]. Proposition nine relates to the concept of construct redundancy, where more than one Petri nets construct was found to map to a particular BWW construct. Again, such instances are undesirable since they allow choice in the interpretation of the resulting process and, hence, introduce confusion over what real-world concept is actually being modeled. Finally, the last two propositions relate to the concept of construct overload. The existence of construct overload is exhibited by multiple mappings of a single BWW construct to Petri nets constructs. Where such cases exist, users will be required to bring to bear additional model knowledge in order to understand why a particular Petri nets construct is being used (over another construct which has the same real-world representation). Interestingly, unlike representational analyses of other popular process modeling techniques [36], the representational analysis of Petri nets shows that this technique does not exhibit construct excess with respect to the BWW model. Hence, the usual weaknesses associated with having additional constructs with no apparent real-world meaning (as far as the representation model is concerned) do not apply in the case of Petri nets.

Type	No	Description and Explanation
Construct Deficit	P1	Because there is no representation in Petri nets for <i>conceivable state space</i> , <i>conceivable event space</i> , <i>lawful event space</i> , as well as various types of <i>property</i> , the definition of states in Petri net process models will lack focus. From a user perspective, the depiction of business rules may be unclear, in particular when exceptions are concerned (lack of lawful and conceivable event space). Also, users may be confronted with insufficient means for describing the properties of real-world objects that traverse different states during the course of a process.
	P2	Because there is no specific representation in Petri nets for <i>kind</i> , the modeling of subtypes of a class of concepts will be impossible for users of Petri nets. This may, for instance, result in an unclear specification of which types of things (belonging to the same class of things) are allowed to participate in a transaction and which are not.
	P3	Because there is no specific representation for a BWW <i>lawful transformation</i> , users will have difficulty in adequately modeling business rules in process models specified with Petri nets. The <i>lawful transformation</i> construct affords the representation of events that are allowed to occur in the system, hence without such representation it is also difficult to identify exception events.
	P4	Because there is no representation for BWW construct <i>history</i> , there is no means in Petri nets to track or log the various state changes that an entity may traverse through during the course of a transaction. Such lack of representation can lead to limitations in the recovery and reliability of interacting entities, such as inter-organizational systems.
	P5	Because no representation for <i>system composition</i> and <i>system environment</i> was found to exist, process models created using Petri nets will be unclear from the perspective of differentiating between internal and external entities to the organization. Furthermore, users will also have to bring to bear additional model knowledge in order to determine the full set of external entities (<i>e.g.</i> clients, other departments, etc.) that can interact with the system (<i>e.g.</i> the user's department) and the set of internal entities (<i>e.g.</i> employees within the user's department) that can affect the system.
	P6	Because there is a lack of representation for <i>system structure</i> , <i>subsystem</i> , <i>system decomposition</i> , and <i>level structure</i> , the definition of the boundaries and layers of process or transaction systems will be problematic. In other words, it will be difficult to demarcate the system from other systems (<i>e.g.</i> when modeling inter-organizational processes, or inter-departmental processes) and also difficult to structure or identify relationships between entities that are internal or external to the systems (further made problematic by proposition P5).
	P7	Because there is a lack of representation for the BWW construct <i>external event</i> , events that occur outside of the system and potentially affect the transformations within the system are not able to be specifically depicted. The depiction of such events would be important in order to accurately identify the risk level of a particular process and will become increasingly important as new reporting requirements are introduced by various governing bodies. Thus, the lack of such representation requires the users to acquire additional information in order to identify which events are external yet pose an effect on the internal system, which in turn would reduce the clarity of the process model.
	P8	The lack of representation for <i>poorly defined event</i> implies that events that may potentially become exceptions (<i>i.e.</i> events that are not deterministic) are not easily identified within process models specified using the Petri nets technique. Similarly to the lack of representation of <i>external event</i> , the lack of this representation makes the models less clear and may result in the

Type	No	Description and Explanation
		incorrect estimation of the risk associated with a given process.
Construct Redundancy	P9	Because the BWW constructs <i>state</i> and <i>unstable state</i> were both found to map to a number of Petri nets constructs, <i>viz.</i> , place, initial marking and token, users will have trouble identifying the correct construct to use in order to represent the states of an entity. More specifically, users may have difficulty deciding whether a state of, for example, an application, should be specified by using an explicit place construct named after the state of the entity (<i>e.g.</i> “processed”) or whether this state can be depicted by using a specific number of tokens. Furthermore, the readers of the model will also suffer confusion in terms of understanding how the state was modeled.
	P10	Because the Petri nets construct ‘place’ was found to map to a number of BWW constructs (and hence a number of real-world concepts), users of the technique will have to have extra model knowledge in order to understand which of real-world concepts is being modeled by each of these constructs. For example, ‘place’ was found to map to <i>thing, class, lawful state space, state, state law, and unstable state</i> - hence, it may be difficult for readers of the model to identify whether the place construct is being used to represent a specific entity, a class of entities or a specific status of an entity that traverses through a transaction.
Construct Overload	P11	Because the Petri nets construct ‘transition’ maps to BWW <i>event</i> and <i>transformation</i> , there is a lack of distinction between the events that can occur to an entity and the change in state that the entity will undergo as a result of the events taking place. For example, a user reading a process model specified in Petri nets, may be confused as to why a certain change of state took place without any specific event being depicted.

Table 2. Propositions derived from the representational analysis

A scan of the set of propositions shows clearly that the majority of potential weaknesses of the Petri net technique stems from the lack of specific representation for various real-world constructs. The implications of some of these are quite worrying in the domain of process modeling. In particular, we refer to the possible weaknesses in adequately modeling events/transformations as well as modeling business rules that govern those transformations. While it would be possible that in modeling practice these issues would be found to be resolved via workarounds by Petri net users (for example, by employing an additional technique for modeling those concepts that cannot be expressed in Petri nets - think of event decomposition diagrams or UML state charts, for instance), such workarounds are generally utilized inconsistently and in an unspecified manner, and hence tend to introduce lack of clarity into the process models. Lack of clarity in turn increases the amount of time that users spend creating and interpreting the models, and also increases the chances for misunderstandings. Moreover, any additional weaknesses, such as the identified construct deficit and construct redundancy issues, only add to the confusion as users struggle to understand the purpose and context of constructs when using Petri nets to model real-world process scenarios.

Following from our discussion of implications, we argue that there is much benefit in empirically testing these propositions and determining which of these are highly significant to the domain of process modeling and which do not appear to be relevant. Such empirical validation then informs process modeling technique developers of the problems that users are facing, which need to be addressed.

Conclusions

Contributions

This paper identified a need for the evaluation of the representational capability of Petri nets, given their history as a foundation for many other process modeling techniques. We argue that the identification of representational weaknesses in the foundational technique will ultimately enable improvements in the techniques stemming from the foundational technique. In order to contribute the first step to this goal, we used a well known ontology-based model of representation - the BWW representation model - as a benchmark upon which the Petri net technique can be evaluated.

Our work shows that Petri nets do suffer from a number of representational deficiencies when modeling real-world domains. These deficiencies can be classified in terms of construct deficit, construct redundancy, and construct overload. In particular, our research shows that there is a large number of BWW constructs, *viz. property, kind, conceivable state space, conceivable event space, lawful event space, lawful transformation, history, system, system composition, system environment, system structure, subsystem, system decomposition, level structure, stable state, external event, and poorly-defined event*, that do not appear to have representation in Petri nets as far as the BWW model is concerned. This situation implies that Petri nets are not able to, in their true original form, model the various real-world concepts that are covered by the missing BWW constructs. Given this insight, the recent extensions of the original Petri nets technique, for example [54] and [55], can be assumed to address some of these deficits. It would be interesting to see how these extended versions of Petri nets perform in light of a representational analysis. Additionally, we also found that there are a few cases of construct overload and redundancy - situations which have in several instances been shown to impact the users' understanding of the resulting models [60, 61]. While it still remains to be empirically tested which of the proposed weaknesses of Petri nets hold in practice, and which do not appear to have an impact in the domain of process modeling, the outcomes are a significant first attempt at the evaluation of Petri nets.

Limitations

While attempts were made to increase the rigor of this research, the work still has some limitations. First, despite the use of an extended representational analysis methodology employing multiple coders, some subjectivity will always exist in the process of such analyses. However, as the obtained raw percentage agreement levels in our mapping shows, we spent considerable effort in increasing the reliability and validity of our analysis. Second, our research denotes a form of analytical study, which in turn can only result in theoretical propositions. The findings from our work call for appropriate empirical research strategies in order to confirm or falsify the implications drawn from our analysis. In this paper we have indicated some propositions that require further operationalisation and testing. We would also like to invite other researchers to contribute in this field of study. Last,

we based our representational analysis on the model of representation stemming from Bunge's work. Clearly, there are other ontologies available [e.g., 21, 62] that may have been used for the purpose of analyzing the strengths and weaknesses of Petri nets and it would be interesting to match the results from such analyses with our propositions.

Outlook

Our study forms a theoretical cornerstone on which empirical knowledge about process modeling with Petri nets can be established. In particular, our forthcoming research sets out to empirically validate the propositions drawn from the analysis presented in this paper. One widely established research method for proposition testing in modeling is by means of experiments [63]. Experiments would allow us to devise stimuli that directly tap into our developed propositions whilst at the same time controlling for potentially confounding contextual variables. A number of experiments have been carried out on basis of the BWV model, for example [60, 61], hence we will be able to refer to a substantial amount of guidance in designing and carrying out such a study. An interesting aspect to study in an experiment would for instance be the comparison of traditional Petri nets to extended Petri nets in their facilitation of domain understanding and problem solving capacities. Our theory would in this case suggest that the more differentiated (thus ontologically clearer) and extended (thus ontologically more complete) Petri net model would outperform the traditional Petri net model.

References

- [1] Gartner Group: Creating Enterprise Leverage: The 2007 CIO Agenda. EXP Premier Report January 2007. Gartner, Inc, Stamford, Connecticut, 2007
- [2] Australian Community of Practice: BPM Round Table. QUT, 2004, available at: <http://www.bpm-roundtable.com>
- [3] Wolf, C., Harmon, P.: The State of Business Process Management - 2006. www.BPTrends.com, 2006
- [4] Sinur, J.: Magic Quadrant for Business Process Analysis. Gartner Research Note M-22-0651 March. Gartner, Inc, Stamford, Connecticut, 2004
- [5] Davies, I., Green, P., Rosemann, M., Indulska, M., Gallo, S.: How do Practitioners Use Conceptual Modeling in Practice? *Data & Knowledge Engineering* 58(3), 2006, 358-380
- [6] van der Aalst, W.M.P.: Three Good Reasons for Using a Petri-net-based Workflow Management System. In: Wakayama, T., Kannapan, S., Khoong, C.M., Navathe, S.B., Yates, J. (eds.): Information and Process Integration in Enterprises: Rethinking Documents. The Kluwer International Series in Engineering and Computer Science, Vol. 428. Kluwer Academic Publishers, Boston, Massachusetts 1998, 161-182
- [7] Verbeek, H.M.V., Basten, T., van der Aalst, W.M.P.: Diagnosing Workflow Processes using Woflan. *The Computer Journal* 44(4), 2001, 246-279
- [8] Desrochers, A.A., Al'Jaar, R.Y.: Applications of Petri nets in Manufacturing Systems: Modelling, Control and Performance Analysis. IEEE Press, Piscataway, New Jersey, 1995
- [9] Petri, C.A.: Fundamentals of a Theory of Asynchronous Information Flow. In: Popplewell, C.M. (ed.): IFIP Congress 62: Information Processing. North-Holland, Munich, Germany 1962, 386-390

- [10] Murata, T.: Petri Nets: Properties, Analysis and Applications. *Proceedings of the IEEE* 77(4), 1989, 541-580
- [11] van der Aalst, W.M.P.: Challenges in Business Process Management: Verification of Business Processing Using Petri Nets. *Bulletin of the EATCS* 80, 2003, 174-199
- [12] van der Aalst, W.M.P., Dumas, M., Ouyang, C., Rozinat, A., Verbeek, H.M.V.: Conformance Checking of Service Behavior. *ACM Transactions on Internet Technology* forthcoming, 2007, in press
- [13] Keen, C.D., Lakos, C.: Information Systems Modelling using LOOPN++, an Object Petri Net Scheme. In: Sol, H.G., Verbraeck, A., Bots, P.W.G. (eds.): Proceedings of the 4th International Working Conference on Dynamic Modelling and Information Systems. Delft University Press, Noordwijkerhout, The Netherlands, 1994, 31-52
- [14] van der Aalst, W.M.P., ter Hofstede, A.H.M.: YAWL: Yet Another Workflow Language. *Information Systems* 30(4), 2005, 245-275
- [15] Wand, Y., Weber, R.: On the Ontological Expressiveness of Information Systems Analysis and Design Grammars. *Journal of Information Systems* 3(4), 1993, 217-237
- [16] Bubenko, J.A.: Information Systems Methodologies - A Research View. In: Olle, T.W., Sol, H.G., Verrijn-Stuart, A.A. (eds.): Information Systems Design Methodologies: Improving the Practice. North-Holland, Amsterdam, The Netherlands 1986, 289-318
- [17] Wand, Y., Weber, R.: On the Deep Structure of Information Systems. *Information Systems Journal* 5(3), 1995, 203-223
- [18] Wand, Y., Weber, R.: An Ontological Model of an Information System. *IEEE Transactions on Software Engineering* 16(11), 1990, 1282-1292
- [19] Bunge, M.A.: Treatise on Basic Philosophy Volume 3: Ontology I - The Furniture of the World. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1977
- [20] 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
- [21] Guizzardi, G.: Ontological Foundations for Structural Conceptual Models. Telematica Instituut, Enschede, The Netherlands, 2005
- [22] Green, P., Rosemann, M.: Applying Ontologies to Business and Systems Modeling Techniques and Perspectives: Lessons Learned. *Journal of Database Management* 15(2), 2004, 105-117
- [23] Evermann, J.: Towards a Cognitive Foundation for Knowledge Representation. *Information Systems Journal* 15(2), 2005, 147-178
- [24] Shanks, G., Tansley, E., Weber, R.: Using Ontology To Validate Conceptual Models. *Communications of ACM* 46(10), 2003, 85-89
- [25] Aristotle: The Metaphysics. Translated by John H. McMahon. Prometheus Books, Buffalo, New York, 1991
- [26] Weber, R.: Ontological Foundations of Information Systems. Coopers & Lybrand and the Accounting Association of Australia and New Zealand, Melbourne, Australia, 1997
- [27] Rosemann, M., Green, P., Indulska, M., Recker, J.: Using Ontology for the Representational Analysis of Process Modeling Techniques. *International Journal of Business Process Integration and Management*, forthcoming, in press
- [28] Rosemann, M., Green, P.: Developing a Meta Model for the Bunge-Wand-Weber Ontological Constructs. *Information Systems* 27(2), 2002, 75-91
- [29] Chen, P.P.-S.: The Entity Relationship Model - Toward a Unified View of Data. *ACM Transactions on Database Systems* 1(1), 1976, 9-36
- [30] Rosemann, M., Green, P., Indulska, M.: A Procedural Model for Ontological Analyses. In: Hart, D., Gregor, S. (eds.): Information Systems Foundations: Constructing and Criticising. ANU E Press, Canberra, Australia 2005, 153-163
- [31] Lindland, O.I., Sindre, G., Solvberg, A.: Understanding Quality in Conceptual Modeling. *IEEE Software* 11(2), 1994, 42-49
- [32] Weber, R., Zhang, Y.: An Analytical Evaluation of NIAM's Grammar for Conceptual Schema Diagrams. *Information Systems Journal* 6(2), 1996, 147-170
- [33] Opdahl, A.L., Henderson-Sellers, B.: Ontological Evaluation of the UML Using the Bunge-Wand-Weber Model. *Software and Systems Modeling* 1(1), 2002, 43-67
- [34] Fettke, P., Loos, P.: Ontological Evaluation of Reference Models using the Bunge-Wand-Weber Model. In: Ross, J.W. (ed.): Proceedings of the 9th Americas Conference on Information Systems. Association for Information Systems, Tampa, Florida, 2003, 2944-2955

- [35] Keen, C.D., Lakos, C.: Analysis of the Design Constructs Required in Process Modelling. In: Purvis, M. (ed.): Proceedings of the International Conference on Software Engineering: Education and Practice. IEEE Computer Society, Dunedin, Ireland, 1996, 434-441
- [36] Rosemann, M., Recker, J., Indulska, M., Green, P.: A Study of the Evolution of the Representational Capabilities of Process Modeling Grammars. In: Dubois, E., Pohl, K. (eds.): Advanced Information Systems Engineering - CAiSE 2006. Lecture Notes in Computer Science, Vol. 4001. Springer, Luxembourg, Grand-Duchy of Luxembourg 2006, 447-461
- [37] Keller, G., Nüttgens, M., Scheer, A.-W.: Semantische Prozessmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)". Working Paper 89. Institut für Wirtschaftsinformatik, Universität Saarbrücken (in German), Saarbrücken, Germany, 1992
- [38] Green, P., Rosemann, M.: Integrated Process Modeling. An Ontological Evaluation. *Information Systems* 25(2), 2000, 73-87
- [39] Green, P., Rosemann, M.: Ontological Analysis of Integrated Process Models: Testing Hypotheses. *The Australian Journal of Information Systems* 9(1), 2001, 30-38
- [40] Green, P., Rosemann, M., Indulska, M.: Ontological Evaluation of Enterprise Systems Interoperability Using ebXML. *IEEE Transactions on Knowledge and Data Engineering* 17(5), 2005, 713-725
- [41] Green, P., Rosemann, M., Indulska, M., Manning, C.: Candidate Interoperability Standards: An Ontological Overlap Analysis. *Data & Knowledge Engineering*, In Press, forthcoming
- [42] Andrews, T., Curbera, F., Dholakia, H., Goland, Y., Klein, J., Leymann, F., Liu, K., Roller, D., Smith, D., Thatte, S., Trickovic, I., Weerawarana, S.: Business Process Execution Language for Web Services. Version 1.1. BEA Systems, International Business Machines Corporation, Microsoft Corporation, SAP AG and Siebel Systems, 2003, available at: <http://xml.coverpages.org/BPELv11-May052003Final.pdf>
- [43] Arkin, A.: Business Process Modeling Language. BPMI.org, 2002, available at: <http://www.bpmi.org/>
- [44] Arkin, A., Askary, S., Fordin, S., Jekeli, W., Kawaguchi, K., Orchard, D., Pogliani, S., Riemer, K., Struble, S., Takacs-Nagy, P., Trickovic, I., Zimek, S.: Web Service Choreography Interface (WSCI) 1.0. BEA Systems, Intalio, SAP, Sun Microsystems, 2002, available at: <http://www.w3.org/TR/wsci/>
- [45] OASIS: ebXML Business Process Specification Schema Version 1.01. UN/CEFACT and OASIS, 2001, available at: <http://www.ebxml.org/specs/ebBPSS.pdf>
- [46] Recker, J., Indulska, M., Rosemann, M., Green, P.: Do Process Modelling Techniques Get Better? A Comparative Ontological Analysis of BPMN. In: Campbell, B., Underwood, J., Bunker, D. (eds.): Proceedings of the 16th Australasian Conference on Information Systems. Australasian Chapter of the Association for Information Systems, Sydney, Australia, 2005
- [47] BPMI.org, OMG: Business Process Modeling Notation Specification. Final Adopted Specification. Object Management Group, 2006, available at: <http://www.bpmn.org>
- [48] Recker, J., Indulska, M., Rosemann, M., Green, P.: How Good is BPMN Really? Insights from Theory and Practice. In: Ljungberg, J., Andersson, M. (eds.): Proceedings of the 14th European Conference on Information Systems. Goeteborg, Sweden, 2006
- [49] Ellis, C.: Information Control Nets: A Mathematical Model of Office Information Flow. In: Roth, P.F., Nutt, G.J. (eds.): Proceedings of the Conference on Simulation, Measurement and Modeling of Computer Systems. ACM Press, Boulder, Colorado, 1979, 225-240
- [50] Holt, A.W.: Diplans: A New Language for the Study and Implementation of Coordination. *ACM Transactions on Information Systems* 6(2), 1988, 109-125
- [51] Zisman, M.D.: Representation, Specification and Automation of Office Procedures In: Wharton School of Business. University of Pennsylvania, Philadelphia, Pennsylvania, 1977
- [52] van der Aalst, W.M.P., ter Hofstede, A.H.M., Kiepuszewski, B., Barros, A.P.: Workflow Patterns. *Distributed and Parallel Databases* 14(1), 2003, 5-51
- [53] Cao, T., Sanderson, A.C.: Intelligent Task Planning Using Fuzzy Petri Nets. World Scientific, Singapore, 1996
- [54] Jensen, K.: Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. 2nd edn. Springer, Berlin, Germany et al., 2003
- [55] Marsan, M.A., Conte, G., Balbo, G.: A Class of Generalised Stochastic Petri Nets for the Performance Evaluation of Multiprocessor Systems. *ACM Transactions on Computer Systems* 2(2), 1984, 93-122

- [56] Baer, J.-L.: A Survey of Some Theoretical Aspects of Multiprocessing. *ACM Computing Surveys* 5(1), 1973, 31-80
- [57] Kiepuszewski, B., ter Hofstede, A.H.M., van der Aalst, W.M.P.: Fundamentals of Control Flow in Workflows. *Acta Informatica* 39(3), 2003, 143-209
- [58] Moore, G.C., Benbasat, I.: Development of an Instrument to Measure the Perceptions of Adopting an Information Technology Innovation. *Information Systems Research* 2(3), 1991, 192-222
- [59] Green, P.: Use of Information Systems Analysis and Design (ISAD) Grammars in Combination in Upper CASE Tools - An Ontological Evaluation. In: Siau, K., Wand, Y., Parsons, J. (eds.): Proceedings of the 2nd CAiSE/ IFIP8.1 International Workshop on Evaluation of Modeling Methods in Systems Analysis and Design. University of Nebraska-Lincoln, Barcelona, Spain, 1997, 1-12
- [60] Gemino, A., Wand, Y.: Complexity and Clarity in Conceptual Modeling: Comparison of Mandatory and Optional Properties. *Data & Knowledge Engineering* 55(3), 2005, 301-326
- [61] Bodart, F., Patel, A., Sim, M., Weber, R.: Should Optional Properties Be Used in Conceptual Modelling? A Theory and Three Empirical Tests. *Information Systems Research* 12(4), 2001, 384-405
- [62] Chisholm, R.M.: A Realistic Theory of Categories: An Essay on Ontology. Cambridge University Press, Cambridge, Massachusetts, 1996
- [63] Poels, G., Burton-Jones, A., Gemino, A., Parsons, J., Ramesh, V.: Experimental Research on Conceptual Modeling: What Should We Be Doing and Why? In: Embley, D.W., Olivé, A., Ram, S. (eds.): Conceptual Modeling - ER 2006. Lecture Notes in Computer Science, Vol. 4215. Springer, Tucson, Arizona 2006, 544-547

About the Authors



Jan Recker is a PhD Candidate at the Business Process Management research group at the Faculty of Information Technology, Queensland University of Technology Brisbane, Australia. He received his BScIS and MScIS from the University of Muenster, Germany in 2004. His research interests include Business Process Modeling, Representation Theory, Process Configuration and Reference Modeling. Findings from his research have been published in more than thirty refereed scholarly publications.



Marta Indulska is a Lecturer at the UQ Business School, The University of Queensland, Brisbane, Australia. She obtained her PhD in Computer Science, in the research area of Information Systems, at the University of Queensland, in 2004. Marta's main research areas are Business Process Management and Representation Theory. She has published and presented her work at numerous international conferences. Her work has also been published by journals such as IEEE Transactions on Knowledge & Data Engineering, Australian Accounting Review, and Data & Knowledge Engineering. Her teaching focuses on topics in Electronic Commerce and Information Systems.