Recker, Jan C. and Indulska, Marta and Rosemann, Michael and Green, Peter (2010)
*The ontological deficiencies of process modeling in practice.*
European Journal of Information Systems, 19(5). pp. 501-525.

# The Ontological Deficiencies of Process Modeling in Practice

ABSTRACT

*Business process modeling is widely regarded as one of the most popular forms of conceptual modeling. However, little is known about the capabilities and deficiencies of process modeling grammars and how existing deficiencies impact actual process modeling practice. This paper is a first contribution towards a theory-driven, exploratory empirical investigation of the ontological deficiencies of process modeling with the industry standard BPMN. We perform an analysis of BPMN using a theory of ontological expressiveness. Through a series of semi-structured interviews with BPMN adopters we explore empirically the actual use of this grammar. Nine ontological deficiencies related to the practice of modeling with BPMN are identified, for example, the capture of business rules and the specification of process decompositions. We also uncover five contextual factors that impact on the use of process modeling grammars, such as tool support and modeling conventions. We discuss implications for research and practice, highlighting the need for consideration of representational issues and contextual factors in decisions relating to BPMN adoption in organizations.*

**Keywords:** *Process Modeling, Usage Behavior, Ontology, BPMN*

# INTRODUCTION

Business process management (BPM) continues to be a top business priority, and building business process capability is a major challenge for senior executives in the coming years(Gartner Group, 2010). Process modeling – the act of capturing and graphically describing the processes in an organization – is an essential means in this endeavor. In fact, process modeling is widely used within and across organizations as a method to increase awareness and knowledge of business processes, to develop or change organizational structures (Bandara *et al.*, 2005), information systems (Dreiling *et al.*, 2006), or web services (van der Aalst *et al.*, 2007).

In managing process modeling initiatives, the type of grammar to be used for process modeling is an important managerial decision (Rosemann, 2006). Generally, the decision for a particular process modeling grammar is associated with substantial investments in tool purchases, training, conventions and methodologies. Furthermore, this decision has significant implications not only for the outcomes and success of modeling initiatives but also for the investment decisions required to establish a productive and sustainable modeling environment. Companies invest a significant amount of time and money to select appropriate modeling tools and to develop organizational conventions to govern process modeling initiatives (Rosemann, 2006). Wolf and Harmon (2008) report that over 50 per cent of the surveyed organizations spent from $500,000 to over $10 million on investments in process modeling training and methods, and similar amounts for related software acquisitions.

Nowadays, the process modeling grammar of choice appears to be the Business Process Modeling Notation (BPMN) (BPMI.org & OMG, 2006). BPMN was developed by an industry consortium whose constituents represent a wide range of process modeling tool vendors. Work on BPMN commenced around 2003, and it was in 2006 that BPMN was released as an Object Management Group (OMG) standard (BPMI.org & OMG, 2006). Since then, BPMN has enjoyed significant uptake and a global community is now dedicated to its further development. The official BPMN web site (www.bpmn.org) already lists currently (as of April 2010) over 60 vendors of process tools that support BPMN, and a fast growing number of organizations in all industries have made BPMN their process modeling grammar of choice (Recker, 2010b).

The attention that BPMN has been receiving, however, has not yet been balanced by a critical analysis of its actual and perceived capabilities and deficiencies. Specifically, the usage of BPMN for process modeling in actual business practice has not yet been examined. Yet, knowledge of the capabilities and issues of BPMN in process modeling practice is instrumental in facilitating well-founded decisions relating to BPMN usage in organizations, and well-informed investments in appropriate tool purchases, training and methodologies.

Accordingly, the *aim of this paper* is to provide a deeper understanding of the *potential* and *actual* issues of process modeling with the current process modeling standard, and to explore the contextual settings in which process modeling with BPMN occurs. The research presented in this paper uses a theory of ontological expressiveness (Wand & Weber, 1993), in conjunction with qualitative, semi-structured interviews, to guide an empirical investigation into the process modeling practice with BPMN. The three *research questions* of this paper are:

- What are the *ontological deficiencies of process modeling with BPMN*?
- Which of these deficiencies are *experienced in actual BPMN modeling practice?*
- Which *contextual factors* affect the BPMN process modeling grammar usage experience?

We proceed as follows. The next section provides an introduction to process modeling with BPMN and then we introduce the theoretical background of our research. Next, we present the propositions we derived from our ontological analysis of BPMN. Then we discuss findings from a series of explorative BPMN-user interviews and distill our findings in a contextualized model of process modeling grammar use. We then discuss how our research informs research agenda on process modeling practice, and then provide a discussion of the implications for process modeling practice before concluding this paper with a brief review of its contributions

## Process Modeling with BPMN

For executive decision makers to develop an informed opinion about the impact and business change of information technology, they require a deep and transparent understanding of operational and managerial processes running in their organization. Process modeling is an approach for describing how businesses conduct their processes and operations. Process models typically include descriptions of at least the activities, events/states, and control flow logic that constitute a business process (Curtis *et al.*, 1992). Additionally, process models may also include information regarding the involved data, organizational/IT resources and potentially other artifacts such as external stakeholders and performance metrics to name just a few (e.g., Scheer, 2000).

Recently, approaches have been suggested for the textual design of business processes (Lee *et al.*, 2008). Yet, typically, process models are designed graphically using so-called process modeling grammars, i.e., sets of graphical constructs and a set of rules how to combine these constructs (Wand & Weber, 2002). While a wide range of different grammars is available for process modeling (Recker *et al.*, 2009), the Business Process Modeling Notation (BPMN) nowadays denotes the grammar of choice for most process modeling efforts, be it as part of process documentation, organizational re-engineering, workflow specification or systems implementation projects (Recker, 2010b).

To allow for multiple application areas, the BPMN specification (BPMI.org & OMG, 2006) differentiates the grammar into a set of core graphical elements and an extended specialized set. The *core set* (a set of basic modeling constructs in BPMN, such as Task, Event, Pool, Lane, Gateway, Normal Flow, Message FLow and Association, see BPMI.org & OMG, 2006) was envisaged to suffice for depicting the essence of business processes in intuitive graphical models, while the *complete set* (all 38 constructs) provides additional constructs to support advanced process modeling concepts such as process orchestration and choreography, workflow specification, event-based decision making and exception handling. Overall, the complete specification of BPMN 1.0 defines 38 constructs plus attributes, grouped into four basic categories of elements, *viz.*, Flow Objects, Connecting Objects, Swimlanes and Artefacts. *Flow Objects*, such as events, activities and gateways, are the most basic elements used to create BPMN models. *Connecting Objects* are used to interconnect Flow Objects through different types of arrows. *Swimlanes* are used to group activities into separate categories for different functional capabilities or responsibilities (e.g., different roles or organizational departments). *Artefacts* may be added to a model, where appropriate, in order to display further related information such as processed data or other comments. Figure 1 shows an example of a BPMN model of a payment process and includes some of the most frequently used BPMN constructs (zur Muehlen & Recker, 2008). For further information on BPMN, the reader can refer to (BPMI.org & OMG, 2006).

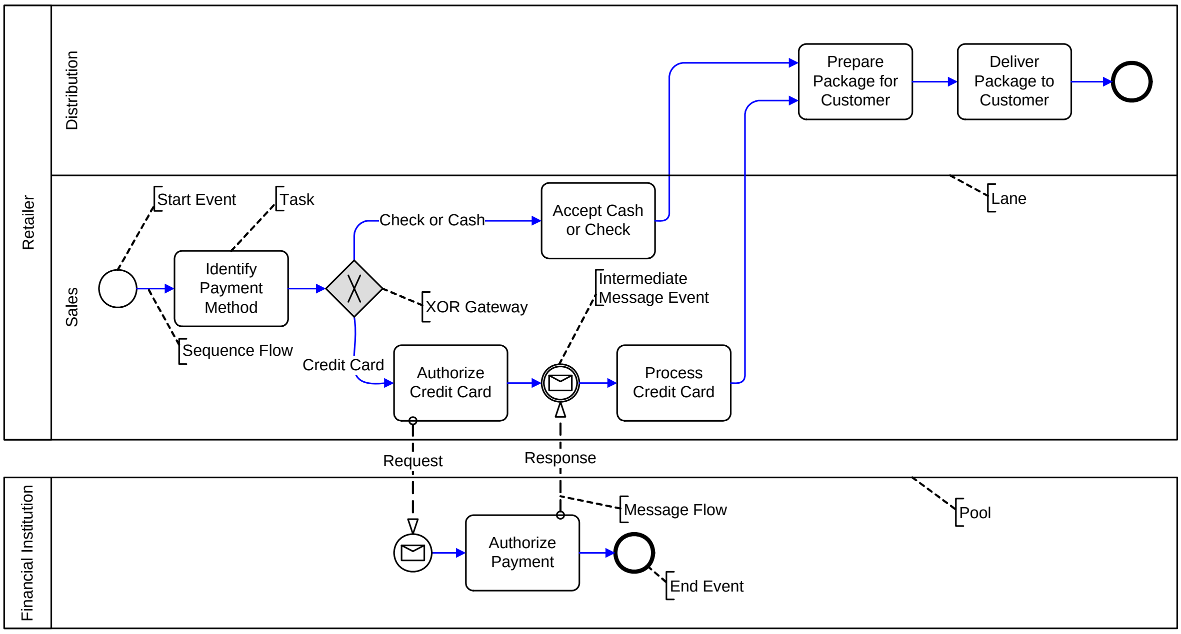


Figure 1: BPMN Example 'Payment Process'

A growing body of research has been carried out on the analysis of BPMN for process modeling, workflow engineering, and systems development. For instance, studies have examined BPMN's capacity to support workflow technology and domain representations (Recker *et al.*, 2007), to facilitate process analysis (Dijkman *et al.*, 2008), and to assist software development (Ouyang *et al.*, 2009). Also, studies have examined BPMN in comparison to other leading process modeling grammars (Recker *et al.*, 2009). In particular, (Recker et al., 2009) found that BPMN, in comparison to other leading process modeling grammars, affords the highest level of representation completeness (from an ontological perspective) but also the lowest levels of ontological clarity. Being the current de facto standard and being representative of a large family of process modeling grammars, our interest in this paper is now to examine the levels of completeness and clarity of BPMN in greater detail, and to extend the comparative ontological analysis in (Recker *et al.*, 2009) by considering empirical data about the usage of BPMN in real-life process modeling practice.

Parallel to the academic debates about BPMN, practitioner debates have emerged that discuss how to 'best' use BPMN (e.g., www.bpm-research.com, www.bpmn-community.org). Several practitioner textbooks have also been published (e.g., Debevoise & Geneva, 2008; White & Miers, 2008). The fundamental questions of how BPMN is *actually* being used in practice, and which issues manifest in practice, however, have not yet been fully examined.

## Theory of Ontological Expressiveness

Our objective in this study is to investigate empirically the modeling issues associated with the capabilities and deficiencies of BPMN for process modeling. To that end, we turn to a theory of ontological expressiveness (Wand & Weber, 1993) to facilitate an examination of this particular process modeling grammar.

This theory is based on the observation that models of business domains and information systems are essentially models of real-world systems. Real-world systems, in turn, can be explained and described using ontology – the study of the nature of the world and what exists in reality – in terms of the properties of, the structure of, and the interactions between, real-world things (Bunge, 1977). Things, their properties, classes of things, and interaction between things are important concepts in the act of conceptual modeling of information systems (Parsons & Wand, 2008). Consequently the application of ontology to this domain has traditionally been of interest to researchers in information systems (Green & Rosemann, 2004).

Wand and Weber's (1993) theory suggests a model of representation, known as the Bunge-Wand-Weber (BWW) representation model, which specifies a set of rigorously defined ontological constructs to describe all types of real-world phenomena that a modeling grammar user may desire to have represented in a conceptual model of an information systems domain. This representation model can serve as a benchmark for the evaluation of the capabilities of modeling grammars to develop models of information systems that are complete and clear. More information about the representation model, its development and its constructs, is available in (Weber, 1997).

Based on this model, Wand and Weber (1993) formulated a theory of ontological expressiveness of modeling grammars. Their theory purports to account for variations in the ability of modelers to develop diagrams of real-world phenomena that are ontologically *complete* and *clear*. Ontological completeness is achieved when the user of a modeling grammar is able to articulate with the grammar all the types of real-world phenomena s/he seeks to have articulated in a model. Ontological clarity is achieved when the user of a modeling grammar is able to articulate in the model all those real-world phenomena in a manner that allows for unambiguous interpretation.

Wand and Weber's (1993) theory of ontological expressiveness is founded on the nature of the mapping between representations and real-world phenomena (similar to theories of recognition and representation in human vision; see, for instance, Edelman, 1998). They argue that for a grammar to be ontologically expressive, such mappings should be isomorphic. Based on this argument, the theory identifies four types of characteristics of conceptual modeling grammars. More precisely, four types of ontological deficiencies of a modeling grammar stemming from a lack of isomorphism (see Figure 2):

1. **Construct deficit**: An ontological construct exists that has no mapping from any modeling construct (a 1:0 mapping).
2. **Construct redundancy**: Two or more modeling constructs map to a single ontological construct (a 1:m mapping).
3. **Construct overload**: A single modeling construct maps to two or more ontological constructs (a m:1 mapping).
4. **Construct excess**: A modeling construct does not map onto any ontological construct (a 0:1 mapping).

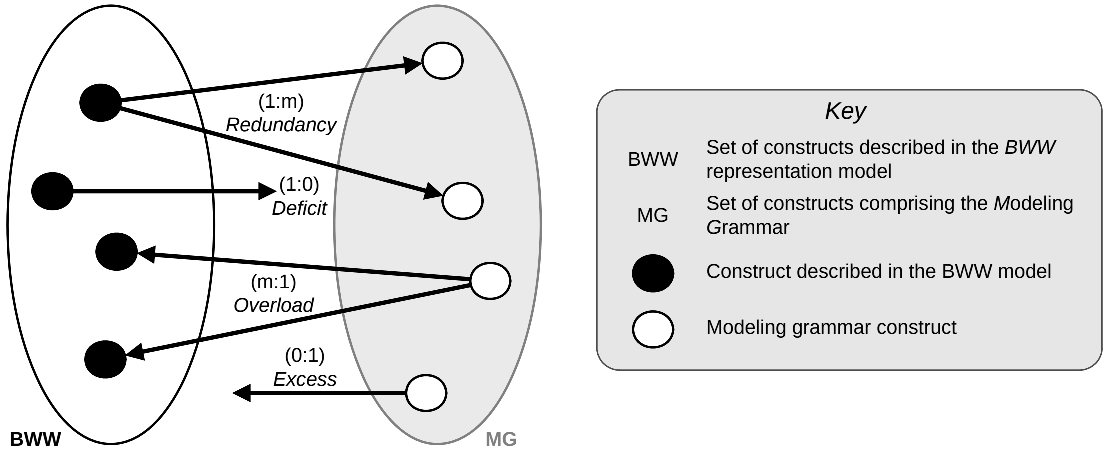


Figure 2: Types of potential ontological deficit and non-clarity

Wand and Weber (1993) argue that lack of ontological completeness –indicated by existence of construct deficit – and lack of ontological clarity – indicated by existence of construct redundancy, overload and excess – undermine a user's ability to use a modeling grammar effectively and efficiently in the act of creating models of real-world phenomena. The process through which construct deficit, redundancy, overload and excess in a modeling grammar can be identified, has become known as the *ontological analysis* of modeling grammars (Rosemann *et al.*, 2009).

Several studies over the last few years have used ontological analyses to test Wand and Weber's (1993) predictions. Green (1996), for instance, showed how construct deficit motivated users of the grammars under observation to employ additional modeling means to articulate those real-type phenomena they felt could not be expressed with the original notation. Bowen *et al.* (2006) and Gemino and Wand (2005) showed how construct excess in a conceptual model resulted in users mis-understanding the model. Shanks *et al.* (2008) demonstrated that construct overload undermined users' ability to understand the information contained in the model.

Other authors (e.g., Bodart *et al.*, 2001; Burton-Jones & Meso, 2006) have undertaken similar empirical tests of the validity of the predictions stemming from Wand and Weber's theory. Most of these studies have examined structured, data-oriented or object-oriented modeling grammars. In the area of process modeling specifically, Green and Rosemann (2000), Recker and Indulska (2007) and Green *et al.* (2007) report on analytical studies of process modeling grammars based on this theory. However, to date, only limited empirical knowledge has been established about the extent to which the theory of ontological expressiveness informs actual process modeling practice. This gap in knowledge motivated the research presented in this paper.

## Ontological Deficiencies of BPMN

Our contention is that ontological deficiencies of a process modeling grammar will also be reflected in the usage of these grammars. More precisely, we contend that these deficiencies will manifest when users attempt to model certain phenomena relevant to process domains.

*Prima facie*, ontologically deficient grammars appear disadvantageous from a theoretical perspective. Yet, the theory of ontological expressiveness is not a psychological or neurophysiological theory of human visual object recognition (Shanks *et al.*, 2008). Yet, the theory of ontological expressiveness provides a formal means of identifying in a process modeling grammar when and where the amount of, and precision of, the real-world semantics the grammar offers to users for process modeling, may be deficient. Based on such an ontological analysis it is possible to articulate a set of propositions about how the identified ontological deficiencies in a grammar will manifest in perceived usability issues in actual process modeling practice.

User perceptions of a modeling grammar, and its graphical constructs, are an important aspect of study. Hitchman (1995), for instance, noted that the perceived usability of the entity-relationship (ER) modeling grammar, an important modeling grammar for database design, is dependent on the user perceptions about its real-world semantics, and only loosely coupled to analytical criteria such as complexity. Also, Krogstie *et al.* (2006) stress the importance of perceived semantic quality (in terms of perceived completeness and perceived validity) to process modeling quality, and note (p. 98) that user perceptions are typically the only feasible way of evaluating semantic quality – which is the interest in our study. Further, Recker (2010a) found that user perceptions about the utility of a modeling grammar specifically are key to forming intentions to continue to work with the grammar.

## Ontological Analysis

To arrive at propositions concerning when and where ontological deficiencies may result in decreased user perceptions about the utility of the BPMN process modeling grammar, in a first step we performed an ontological analysis (Wand & Weber, 1993) of the BPMN grammar. Ontological analysis concerns the bi-directional mapping of process modeling grammar constructs to the ontological constructs specified in the BWW representation model (see Figure 2).

To ensure a rigorous approach towards ontological analysis, we followed an extended methodology that strengthens the reliability and internal validity of such work (Rosemann *et al.*, 2009). Specifically, our analysis was conducted in three steps. First, two of the authors separately read the BPMN specification and mapped the BPMN constructs against BWW constructs in order to create individual first mapping drafts.

Second, the researchers met to discuss and defend their mapping results. Third, the jointly agreed second draft was discussed and refined in several meetings with all four authors. By reaching a consensus at the end of this process, we increased the reliability and validity of this type of research. Rounds two and three of the mapping were concerned specifically with identifying, and discussing, any mappings of BPMN constructs to representation model that were inconsistent between the researchers. Consider this example: in the first, individual mapping round, one researcher classified the BPMN construct “Data Object” as excess. This choice was reasoned in referral to the BPMN specification (BPMI.org & OMG, 2006), which states that the use of this artifact does not affect the other parts of the domain representation contained in the model. Hence it was argued that the Data Object construct does not carry real-world semantics. The other researcher, however, afforded Data Object a mapping to the BWW representation model construct Thing, based on the observation that a Data Object is used to depict information objects, both physical and electronic, and accordingly represents real-world objects such as documents or data records. After discussion and study of specification documents, in the second mapping round both researchers individually revised their mappings. One researcher maintained their mapping of Data Object to Thing while the other mapped it to ”Class.” This was justified by the observation that a Data Object actually does not model a specific document or data record (such as invoice No 47-11) but instead only types of objects (e.g., invoice, policy, customer master record). These two alternative mapping suggestions were presented to, and discussed with, the entire research team who together studied the specification of the constructs and, eventually, agreed to afford the Data Object a mapping to Class. This process was carried out for all other construct mappings.

In order to display inter-coder reliability in the mappings, two types of agreement statistics were derived. Both a raw percentage agreement (Moore & Benbasat, 1991) and Cohen's Kappa (1960) were used to measure the agreement between the coders. Raw percentage agreement for the representation mapping of BPMN to BWW constructs between the two researchers was calculated to be 69 percent in the first round and 87 percent in the second round. Raw percentage agreement was calculated by counting, for each BWW construct, the number of mappings to BPMN that were identical between the two researchers, and calculating the ratio of identical mappings to all possible mappings (38 BWW constructs plus construct excess). The Cohen's Kappa statistic was computed by assigning, for each researcher, a categorical number to each unique construct mapping, and then calculating the inter-rater agreement between these mapping categories. Kappa was calculated to be 0.62 in the first round and 0.83 in the second round. Both measures exceed generally recommended Kappa levels of 0.60 (Moore & Benbasat, 1991). In the third round, the mapping was discussed and refined with all four researchers until a 100 percent agreement across the entire research team was obtained. The Appendix summarizes the outcomes of the final agreed mapping and also displays the agreed reasoning behind the mappings of BWW constructs to/from BPMN constructs. In the Appendix, each BPMN construct is linked to one or more BWW representation model constructs, and a textual description is presented providing the rationale for the mapping. Instances where BPMN constructs are linked to more than one BWW representation model construct indicate the existence of construct overload (e.g., the Lane construct). Instances where one BWW construct maps to more than one BPMN construct indicate the existence of construct redundancy (e.g., for the BWW construct 'External Event'). BPMN constructs that were found not to have a mapping to any BWW construct are classified as construct excess (again, reasoning is provided). All rows in the Appendix where for a

BWW representation model construct no mapping to any BPMN construct was identified, indicate the existence of construct deficit in the BPMN grammar.

## Final Propositions

The outcome of the ontological analysis of BPMN allows us to put forward nine propositions in order to investigate the implications of the lack of ontological completeness and clarity for the actual use of the grammar.

### *Propositions related to Construct Deficit*

First, the bi-directional mapping of BPMN to the BWW constructs identifies a lack of support for state-based concepts (e.g., state, state law, state spaces). Because there is no representation for state, stable state, unstable state, conceivable state space, state law, lawful state space, conceivable event space, and lawful event space, a sufficient focus to identify all important state and transformation laws may not be present during modeling processes with BPMN. Yet, these laws are the basis of what are known in information systems as business rules (Kovacic, 2004). Integrating information about business rules in process models is a missing feature noted previously in other grammars (Green & Rosemann, 2001; Recker & Indulska, 2007). Business rules depict organizational policies and decision-making strategies pertaining to the execution of business processes and thus are essential to capturing the essence of a process. Business rules document how, in a certain state of a process object (e.g., an invoice), certain laws apply to the transformation of this object state to another. Because process modeling and rule modeling grammars are both used to document organizational policies and procedures, a better integration of business rule modeling into process modeling would allow organizations to maximize synergies, avoid content duplication, and thus reduce their overall modeling effort (zur Muehlen & Indulska, 2010). Accordingly, because BPMN has no ontological capacity to depict these state and transformation laws, we expect:

**Proposition 1.** *BPMN users will lack adequate means for the depiction of business rules in process models.*

Second, the representation mapping identified a lack of representation for the ontological construct *history of state changes*. BPMN does not support the design of process models with explicit consideration given to the traceability of the process objects that are the focus of the models. A similar deficiency was also found to exist in the Petri net grammar (Recker & Indulska, 2007). The specification of a history of states that a process object has traversed through its lifecycle, however, could be leveraged for a range of areas of process-related decision-making scenarios. Consider the case of credit history checks or customer relationship management processes, where key decisions are made on the basis of the history of the relevant process object (e.g., a credit card applicant or a frequent flyer member). Accordingly, we expect that BPMN users will encounter difficulties in meeting the potential need for explicit graphical representation of logs of state changes:

**Proposition 2.** *BPMN users will lack means for the depiction of logs of state changes in process models.*

Third, process models can be systematically structured into constituent parts at different levels of abstraction. Graphically representing the process structure and decomposition in a process model can be used, for instance, to demarcate entities in inter-organizational business scenarios. Also, symbols that allow representation of the structure of a process can help to clarify the scope and boundaries of the modeled process. Similar to the case of the EPC grammar (Davies *et al.*, 2004), there is no representation for system structure in BPMN as per the representation mapping. Accordingly, there is no thorough demarcation of the process system and the things within the system. This deficiency can lead to difficulties in the use of BPMN for modeling inter-organizational business processes. We expect that users are unable to coherently articulate the break-down of the modeled process system, accordingly:

**Proposition 3.** *BPMN users will lack means for the specification of process structure and decomposition in process models.*

### *Propositions related to Construct Redundancy*

Fourth, in BPMN a real-world *thing* can be represented by either the Pool or the Lane construct. Real-world things are the process objects, actors, or entities that participate in a process. Each of these things can be represented through multiple BPMN constructs, potentially causing confusion to the user about the redundancy and overlap between the meanings of these two constructs. This situation is similar to that indicated in (Davies *et al.*, 2004), where it is reported that some users of the EPC grammar expressed confusion over the exact type of grammar construct to use for representing a thing. Similar to these findings, in the case of BPMN we expect that users will have difficulty understanding which of these constructs to use for modeling real-world process objects or entities:

**Proposition 4.** *BPMN users will have difficulty differentiating between the Lane and Pool construct use for the graphical articulation of real-world objects in process models.*

Fifth, transformations denote the core steps in a business process, where the process object is being manipulated and changed as a result of the tasks that are being executed. Recker *et al.* (2009) found that of the twelve leading process modeling grammars they considered, 57 % exhibited construct redundancy concerning the ontological concept of a *transformation*. In BPMN, such a *transformation* can be represented by the constructs Activity, Task, Collapsed Sub-Process, Expanded Sub-Process, Nested Sub-Process and Transaction. This situation suggests that these BPMN constructs share a similar meaning in that they can all be used to articulate the steps of transformations occurring during a business process. Given this set of available constructs, we expect that users will be confused as to which construct to use when representing a transformation. Accordingly:

**Proposition 5.** *BPMN users will have difficulty understanding which BPMN construct to use for the graphical articulation of transformations in process models.*

Sixth, similar to transformation, many process modeling grammars have more than one construct to articulate *events*. Indeed, Recker *et al.* (2009) estimated that 71% of process modeling grammars exhibit construct redundancy in this regard. Events denote important triggers in a process, informing the process when to start, when to alter the execution and when to terminate. Such an *event* can be represented by the BPMN constructs Start Event, Intermediate Event, End Event, Message, Timer, Error, Cancel, Compensation and Terminate. Again, we expect users of BPMN to be confused as to which construct to use when representing an event that triggers certain action in a business process. Accordingly:

**Proposition 6.** *BPMN users will have difficulty understanding which BPMN construct to use for the graphical articulation of events in process models.*

### *Propositions related to Construct Overload*

Seventh, Recker *et al.* (2009) found overload deficiencies across a wide range of process modeling grammars (for instance, Petri Nets and EPCs) and speculated that the extra effort required for specifying the representational capacity in which overloaded constructs are used diminishes the ease with which these models can be built. In the case of BPMN, the representation mapping revealed that the BPMN construct Lane maps to the BWW constructs thing, class, kind, system, subsystem, system composition, system environment, system decomposition, and level structure. Accordingly, we expect that users will be required to bring to bear extra model knowledge to understand which real-world concept exactly is being modeled by the Lane construct in any given modeling scenario. Consider, for example, a question whether a Lane in a BPMN model represents a specific organizational entity, an application system, or a set of entities such as a group of actors. It is hence expected that BPMN users often use the Lane construct to model a variety of real-world phenomena, thereby potentially increasing the ambiguity of the resulting process model. Accordingly:

**Proposition 7.** *BPMN users will have difficulty specifying exactly which real-world phenomenon is being graphically articulated by the Lane construct in a process model.*

Eighth, similarly, the BPMN construct Pool was found to map to the BWW constructs thing, system, subsystem, system composition, system environment, system decomposition and level structure. We expect that users will be required to bring to bear extra model knowledge in order to understand which real-world concept is modeled by the Pool construct. Specifically, it is unclear whether a Pool stands for a single organizational entity, whether it is part of a super-ordinate entity, or whether it might be external to a modeled system (for instance, another organization participating in a business-to-business transaction). Accordingly:

**Proposition 8.** *BPMN users will have difficulty specifying which real-world phenomenon is being graphically articulated by the Pool construct in a process model.*

### *Propositions related to Construct Excess*

Finally, the BPMN constructs Link, Off-Page-Connector, Association Flow, Text Annotation, Group, Activity Looping, Multiple Instances, Normal Flow, Event (super type), and Gateway (including all Gateway types) were classified as construct excess as per the representation mapping. Construct excess is also found in other grammars, such as ebXML (Green *et al.*, 2005), BPML, WSCI and WS-BPEL (Green *et al.*, 2007). In the theory of ontological expressiveness, constructs classified as excess have no real-world meaning, and, consequently, their use will cause understandability problems. Users would have to bring to bear extra model knowledge to make sense of these constructs and to understand their nature and purpose. Accordingly, it is expected that users cannot articulate precisely the meaning of these constructs. This situation suggests that modelers might avoid the use of such constructs in their process modeling because the use of excess constructs would only introduce confusion about the nature and purpose of these constructs in a process model, and would therefore decrease the perceived utility of the grammar. Accordingly, we have:

**Proposition 9.** *BPMN users will avoid the constructs Link, Off-Page-Connector, Association Flow, Text Annotation, Group, Activity Looping, Multiple Instances, Normal Flow, Event (super type) and the Gateway construct types when creating process models.*

### *Research Model*

In summary, we suggest nine propositions about how theoretically established ontological deficiencies of the BPMN process modeling grammar are expected to manifest in issues in actual process modeling practice. We conceptualize our propositions in the research model shown in Figure 3, which shows that, as per our propositions, we expect that the ontological deficiencies identified in BPMN will lead to decreased utility perceptions of the users working with the grammar.

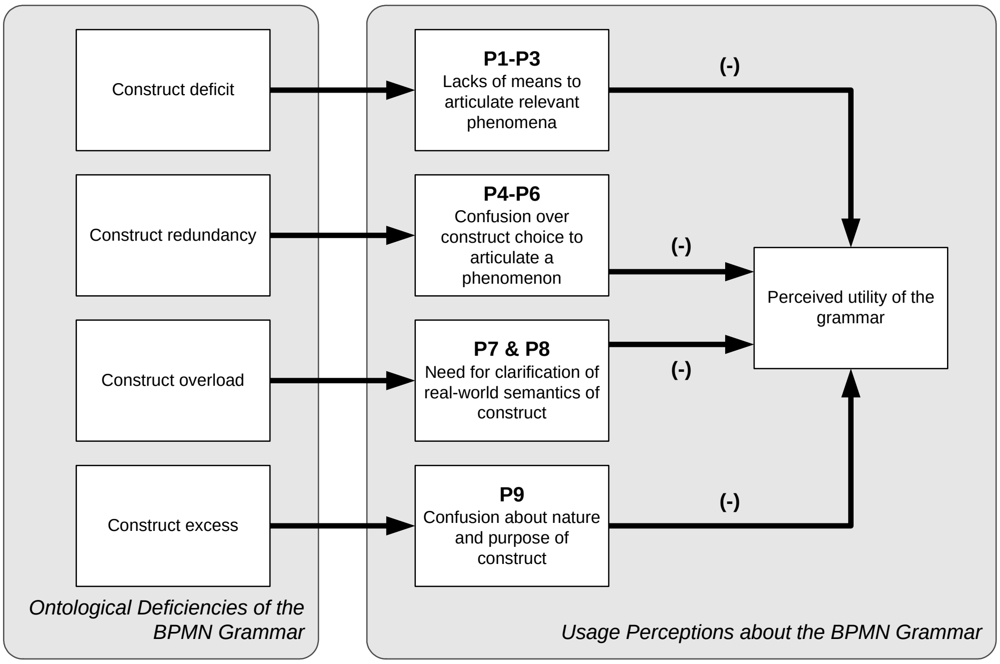


Figure 3: Overall Research Model

# Exploring the Ontological Deficiencies of BPMN

Ontological analyses can suggest whether one modeling grammar has deficits in completeness and/or clarity. However, an ontological disadvantage does not necessarily imply a practically observable disadvantage or issue (Gemino & Wand, 2005). In particular, it is not clear that conclusions derived from an ontological analysis accurately reflect the way that people employ a process modeling grammar in real-life modeling scenarios. Rather, analysis of how people employ and use process modeling grammars in real-life modeling scenarios requires empirical investigation.

In designing our empirical study, we sought not to restrict ourselves to *measuring* purely the theorized shortcomings of BPMN but rather to *explore* the context in which certain ontological deficiencies may or may not occur and why that would be so. We deemed semi-structured interviews (Benbasat *et al.*, 1987) to be an appropriate empirical research method, mainly because they allowed us to carry out research in an area in which few previous studies exist and to examine our phenomenon of interest from the user perspective (Myers & Newman, 2007). Semi-structured interviews are predominantly of qualitative nature and guidelines for their design and conduct have been described extensively in literature. Semi-structured interviews in the context of ontological analysis have briefly been introduced in (Green & Rosemann, 2001). These guidelines have been followed and extended in this study.

In terms of reliability and validity of our empirical study, we followed the guidelines of Yin (2003). To build construct validity into the research model, we maintained an evidence database to provide opportunities to reconstruct data collection and analysis procedures. We used a two-person research team with pre-defined interviewing roles (interview moderator and note taker) to strengthen the validity of the findings and results drawn from the interviews (Dubé & Paré, 2003).

External validity was strengthened with the use of replication logic. In this research, we used the theory of ontological expressiveness, and the propositions drawn from our ontological analysis, to build a semi-structured interview protocol to guide our empirical study across all interview settings. The protocol was developed on the basis of the nine propositions introduced in the previous section. A copy of the interview protocol is available from the authors on request. The protocol consists of two main sections – a section that collects demographic information (Section A) and a section composed of questions related to the propositions (Section B).

All interviews were conducted with the use of the protocol and were also recorded and transcribed. The transcriptions were analyzed using a thematic coding process (Boyatzis, 1998). Thematic analysis attempts to uncover a range of concepts or themes within textual or verbal communications or statements, and to quantify and analyze the presence or strengths of these concepts. In our study, this approach is helpful to uncover issues in the use of the BPMN modeling grammar and to relate these back to our theoretical propositions. To that end, the interview transcriptions were cross-referenced and examined using a data analysis tool.

To guide our empirical investigation, we developed classification schemes to consistently and comprehensively classify participant responses to Section B questions. This work is based on, and further extends, protocols that were previously used by Green and Rosemann (2001). One classification scheme each for construct deficit, redundancy, excess and overload, was developed for this study (see Figure 4). Our pilot tests, as well as prior work using initial versions of the classification scheme (Davies *et al.*, 2004), showed these classifications to be appropriate.

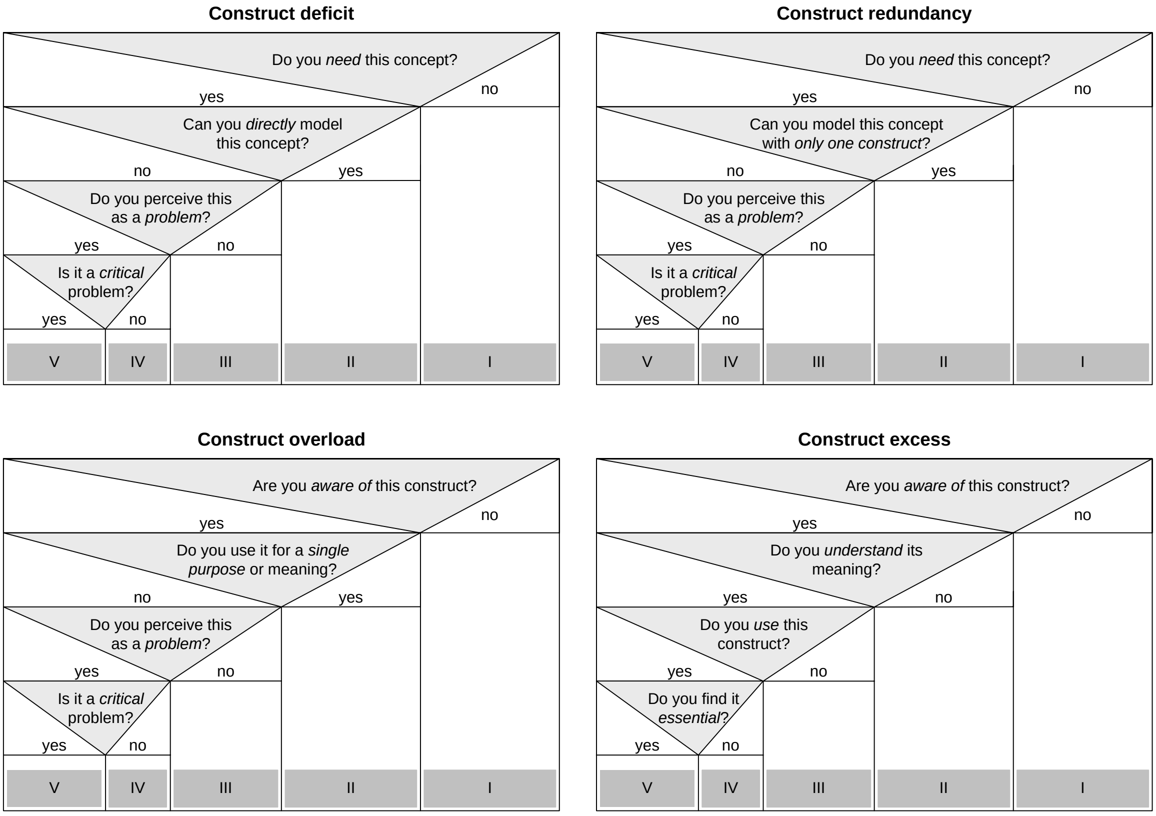


Figure 4: Response classification schemes

As an example, we consider the construct deficit classification scheme (see Figure 4). Under this scheme, the first question asked of the participant is that of need for a particular modeling concept, e.g., “have you ever had the need to graphically represent business rules?” If the response is negative, it is classified as a type I response. If the answer is positive, a further question regarding the ability to model directly the concept is asked, e.g., “can you explicitly graphically represent business rules using BPMN constructs?” This response can be classified as a type II response if the participant can directly model the concept in question. Otherwise, they are asked to indicate if they perceive this inability to be a problem. If not, then a type III response is recorded, otherwise a type IV or V response is recorded, depending on the criticality of the problem. Type V responses can be seen as the strongest form of support for a proposition regarding construct deficit, redundancy or overload. In the case of construct excess, a type III response (i.e. a modeler NOT utilizing an excess construct) is the strongest support for propositions, with a type V response indicating lack of support (i.e. indicating that a construct considered as ‘excess’ is indeed seen as essential in practice).

We perused the four classification schemes to direct our interview strategy and to code appropriately the different responses received. The coding assisted in our qualitative examination by adding quantified information to our response analysis, which, in turn, allowed us to produce more valid, persuasive arguments about the level of support received for our propositions (Sandelowski, 2001).

## Interview Results and Discussion

In our study, six organizations across four Australian states participated as research cases, and a total of nineteen practitioners of these six organizations, incorporating various roles in their respective business environments, (*e.g.*, business analyst, technical analyst, modeling team leader), were interviewed. In total, we obtained 23.5 hours of interview data, resulting in 239 pages of interview transcriptions for analysis.

The case sites visited varied in organizational size, with one organization employing less than 1,000 employees, two organizations employing between 1,000 and 5,000 employees, and the remaining three organizations employing over 5,000. The participants within these organizations ranged in terms of the type of training they received in BPMN, with 31% having finished a formal training course on BPMN, 15% having embarked on an organization-internal course and the remaining 54% having learned BPMN on the job or by reading the specification themselves.

In discussing the findings from the interviews, we follow the four types of ontological deficiency suggested by Weber (1997) and complement these with a discussion of contextual factors and themes that emerged during our data analysis. Table 1 gives a summary of the raw number of response types (as per our classification scheme, see Figure 4) received per proposition. In Table 1, for each proposition, the number of responses is given that matches a particular response type as per Figure 4. For example, for proposition P1, 7 out of 19 interviewees responded that they did not have a need to model business rules with BPMN (response type I). Of the remaining twelve interviewees, 3 indicated that they had no problem modeling business rules directly with BPMN (response type II). Of the remaining nine participants, four indicated that the lack of representation support was not a problem (response type III). The remaining five interviewees noted the lack of support to be a minor problem (response type IV). No interviewee classified the lack of support as a major problem (response type

V). In Table 1, for proposition P9 (which concerns the use of twelve excess constructs in BPMN), the average of responses per response type across all twelve excess constructs is given. The next subsections present in detail the findings from our thematic analysis of the responses.

Table 1. Summary of interview results per proposition

| | | Response type | | | | |
|---|---|---|---|---|---|---|
| Ontological Deficiency | Proposition | I | II | III | IV | V |
| Construct deficit | P1 | 7/19 | 3/12 | 4/9 | 5/5 | 0/0 |
| | P2 | 12/19 | 1/7 | 5/6 | 1/1 | 0/0 |
| | P3 | 2/19 | 5/17 | 6/12 | 3/6 | 3/3 |
| Construct redundancy | P4 | 4/19 | 8/15 | 5/7 | 1/2 | 1/1 |
| | P5 | 0/19 | 18/19 | 0/1 | 0/1 | 1/1 |
| | P6 | 0/19 | 14/19 | 1/5 | 3/4 | 1/1 |
| Construct overload | P7 | 1/19 | 8/18 | 9/10 | 1/1 | 0/0 |
| | P8 | 5/19 | 5/14 | 5/9 | 3/4 | 1/1 |
| Construct excess (average) | P9 | 1.67 | 0.92 | 3.92 | 2.50 | 10.00 |

### *Dealing with Construct Deficit*

**Proposition P1 – regarding the representation of business rules –** has apparent support based on the participants' comments. Questioning about the need to model business rules uncovered that 63 percent of participants (12 responses out of 19; i.e. responses of type II, III, IV and V only) had a need for representing business rules in their process models. Some of the reasons given by the remaining seven participants were that their process models are intended for representatives from the lines of business who may not have the experience to read more complicated diagrams, and that the organization wanted to start with simpler diagrams in order to facilitate process understanding. Of the twelve participants that had a need for business rule specification, 75 percent (9 responses out of 12) stated they could not directly model business rules in BPMN. This finding represents 33 percent of participants (4 responses out of 12) who found workarounds and therefore do not consider this aspect to be a modeling issue, and 42 percent of participants (5 responses out of 12) who consider the inability to directly integrate business rules into a process description to be a problem. Specifically, some comments indicated that having an explicit graphical representation would indeed be preferable:

*"[…] A symbol that says something specifically is a business rule so that you know in future to look at it, mightn't be bad."* (interview transcription data)

This situation suggests that users have trouble identifying the interface between process modeling and business rule modeling grammars, and expect better support in the identification of appropriate interfaces between process logic and business rule logic in a process model. Some of the workarounds used when integration of business rules is not supported (as with BPMN) included narrative descriptions of rules and conditions, using spreadsheets and external tables, and using additional tools that allow users to create hyperlinks to documents, meta-tags and attribute fields. Our evidence, however, suggests that these workarounds are deemed problematic in practice due to the representational separation of the modeled rules and the process model they are of relevance to.

**Proposition P2 – regarding the representation of state history** – had limited support based on the participants' comments. 37 percent of participants (7 responses out of 19) have a need to model the history of state changes. Some of the remaining respondents indicated that there is a need for such modeling but that it has not yet been done in their organization or that they have not yet figured out how to do this task. Another suggested reason for not requiring such direct modeling was the fact that, in this particular organization, process models were predominantly used to convey and communicate business procedures, whereas history of state changes would be a logical software design state:

*"[…] business process modeling is the way I look, see what the business does. Change of state is when you start looking at your data changing and not with your processes, and your entry life cycle, so it's analyzing what the business does."* (interview transcription data)

Of the seven participants that indicated a need for modeling the history of state changes, over 71 percent (5 responses out of 7) indicated that they had no way of modeling this concept directly. This situation, however, was not found to be a problem. Some workarounds involved simply having multiple activities on the diagram with names that implied state changes of things.

**Proposition P3 - regarding the representation of process structure and decomposition -** had apparent support based on the participants' comments. Of the nineteen responses, 89 percent (17 responses out of 19) indicated a need for capturing the process structure and decomposition explicitly in a model. Of these, seventeen participants, 29 percent (5 responses out of 17), indicated that they model process system structures with BPMN directly with the use of pools, lanes, and start/end events while 71 percent (12 responses out of 17) indicated that they lacked direct modeling capabilities in BPMN and had to rely on free form texts, links implemented in the tool to other diagrams (such as IDEF0 models) and further hyperlink functionality provided. Three participants classified the lack of direct modeling capabilities as a minor problem, with another three participants classifying it as a major problem, in fact, a major deficiency. Others indicated that the need for a more explicit graphical representation for process structure and decomposition should indeed be on the agenda for a revision of BPMN:

> *„[…] I think if the standard allows for a large amount of decomposition, my understanding is that it doesn't at the moment, but if, the people see it as that's the way they want to use it, we definitely need something to link the two […]. Because it's designed the way it is, we're not supposed to use it that much, but I know some people have that need."* (interview transcription data)

### *Dealing with Construct Redundancy*

**Proposition P4 - regarding the representation of process-related objects -** had limited support based on the participants' comments. Results indicated that 21 percent of participants (4 responses out of 19) had no limitations in modeling objects with BPMN. These users predominantly used either Pools or Lanes, as predicted by the ontological analysis. However, over 35% of participants (7 responses out of 19) indicated that they encountered problems when trying to directly model such objects with BPMN. Of these seven respondents, 71 percent (5 responses out of 7) indicated that this situation was not a problem, mainly due to the use of organizational modeling conventions regarding the use of tools or complementary textual annotations. Some of the more experienced participants who indicated the issue not to be a problem admitted that it would indeed be a problem if they did not have additional tools into which BPMN was incorporated in their organization, i.e., if they were using BPMN in isolation. These responses suggest the relevance of conventions to the usage experience, and perceived utility, of a process modeling grammar.

**Proposition P5 - regarding the representation of state transformations -** had no apparent support based on the participants' comments. Contrary to our expectations, our results indicate that over 94 percent of participants (18 responses out of 19) indicated their ability to directly model transformations with no confusion. Only one participant indicated a problem and he/she classified it as a major problem. It is interesting to note that some of the respondents used additional workarounds (such as color-coding of activity symbols) to allow a more refined differentiation of transformations, e.g., between automated and manual tasks. Also, methodologies and guidelines for process modeling were in place and they included templates and conventions that reduced any confusion over which BPMN construct to use in which scenarios:

> *„[…] we just use a main one, activity. Certainly naming, naming is one of our challenges. We just try and make them short enough but meaningful enough and what level of detail you put in the name or what level do you attach to documentation. […] we've got a formal naming convention."* (interview transcription data)

**Proposition P6 - regarding the representation of events –** had no apparent support based on the participants' comments. We found that 74 percent of the interviewees (14 responses out of 19) stated that they did not experience any limitations in using BPMN for the modeling of events. In fact, some interviewees stated that the event specializations comprehensively and rigorously allow for the depiction of different events that may impact business operations:

> *"[…] and that's where BPMN is really good, all the options you can think of. You've got the timer event and then you've got a message arriving, all sorts of things."* (interview transcription data)

However, out of the 26 percent of responses (5 responses out of 19) indicating that they encounter limitations in capturing events, 80 percent (4 responses out of 5) categorized this limitation as a problem (minor or major). Our empirical data furthermore suggests that modelers using the core BPMN set (that does not contain event specializations) encounter more difficulty in capturing business events. Only 50 percent of the six core set users (3 responses out of 6) stated that they have no limitations in modeling events, and 67 percent out of the remaining core set users (2 responses out of 3) classified this issue as a (minor) problem. This finding is contradictory to our theoretical proposition, which predicts that confusion would arise when using the full specification set. Core set users often responded that the limited set of constructs is not explicit enough and needs to be extended to capture comprehensively different events. Other interviewees stated that the usage of the core set forced them to complement their BPMN diagrams with additional documents, further refining the events represented in the model. While this situation was not perceived to be a major problem due to this workaround, responses nevertheless indicated that the event specializations in BPMN are perceived as a helpful feature.

### *Dealing with Construct Overload*

**Propositions P7 and P8 - regarding the use of the Lane and Pool constructs -** had apparent support based on the participants' comments. The interview responses clearly indicate ambiguities in the specification of the BPMN Lane and Pool constructs. Of all respondents, only 5 percent (1 response out of 19) and 26 percent (5 responses out of 19) of the interviewees stated they do not use the Lane and the Pool construct respectively. This situation was mostly because, in these cases, the organizational boundaries of a process were self-evident and need not be captured in a model. Conversely, 56 percent and 64 percent of the responses, respectively, used the Pool and Lane constructs for two or more distinct purposes or meanings, with 29 percent of the interviewees (4 responses out of 14) using the Pool construct even for three distinct purposes or meanings. The types of purposes used for the Lane construct included, *inter alia,* roles (used by 61 percent of interviewees), organizational units and business areas (39 percent), scoping (22 percent) and grouping (17 percent). In terms of the Pool construct, purposes included external organizational units and business areas (64 percent), internal organization (50 percent), scoping (29 percent), and grouping (21 percent). Some of the interviewees explicitly mentioned the flexible specification of the constructs as a rationale for inconsistent use:

> *"[…] we sometimes use it* [the Pool symbol] *at an organizational level. Sometimes we use it as a business level, sometimes we use it as sector level, it's not really consistent, because of the nature of the symbol."* (interview transcription data)

In fact, one interviewee, when asked about the usage of the Pool construct, responded that it was a nice concept but difficult to grasp and the reason they did not use it was to keep the models simple and understandable. These findings suggest that end users tend to struggle with the additional usage complexity that comes with flexible grammar specification and interpretation. Instead, they appear to be more receptive towards more rigid construct definition and guidance.

*Dealing with Construct Excess*

**Proposition P9 - regarding potential confusion about nature and purpose of some modeling constructs -** had varying levels of support based on the participants' comments. The levels of support for this proposition ranged from apparent support for three constructs, to limited support for four constructs and lack of support for six constructs. The constructs Off-page connector, Group, and Multiple Instances were classified by over 50 percent of interviewees as being 'not in use' (63 percent, 58 percent, 63 percent, respectively, in total for response types I, II and III). Some other proposed excess constructs, however, were rated as essential for process modeling activities (response type V). Examples include the constructs Normal Flow, Event and Link. The responses indicate that while some constructs (such as Link) may have no real-world semantics, they are nonetheless perceived as important for process modeling as they allow for better demarcation and linking of large-scale process models. The same case holds, for instance, for the constructs Text Annotation and Association Flow. From a representational viewpoint, such constructs may be regarded as 'support constructs' that may be helpful for the act of modeling but have no real-world meaning per se. However, in modeling practice, they seem to be very useful for complementing the graphical models with extra information, for instance additional textual descriptions that help inexperienced model readers to better understand the process specifications:

*„[…] I think they're useful. They are essential. You need some form of clarification. Maybe not in future when everyone's used to these maps, but at the moment it's very limited."* (interview transcription data)

Support constructs such as Text Annotations were more often perceived as essential by the practitioners that used the core BPMN set only (67 percent) than amongst extended set users (46 percent). In these cases, Text Annotations provide the benefit of adding process details to a model without having an explicit graphical representation. It should be noted, however, that these constructs (in particular Link and Text Annotation) appear to be also used in practice to mitigate some of the identified shortcomings (in particular, lack of process decomposition and lack of adequate business rule representation).

### Contextual Factors Affecting Process Modeling Practice

The semi-structured interviews allowed us to gather user feedback on the propositions derived from our theoretical analysis. However, we also sought to gather deeper insights into contextual factors that may have remained undiscovered by the ontological analysis but are key to understanding actual process modeling grammar use.

To that end, the two-person interviewing team used probing questions in addition to the interview questions pertaining to the classification scheme (see Figure 4) to explore contextual factors that may provide a rationale for the insights obtained. We conducted a thematic coding analysis (Weber, 1990) on the transcribed interview data to uncover central, recurring concepts across the responses.

Overall, our study uncovered five contextual factors that appear to influence the usage of BPMN and the perceived criticality of its suggested ontological deficiencies, *viz*. use of modeling tools, use of modeling conventions, modeler experience, the modeling role and the modeling purpose. In identifying and describing these factors, we refer to existing theoretical conceptualizations of conceptual modeling practice (Wand & Weber, 2002) and process modeling practice (Bandara *et al.*, 2005), which distinguish individual factors such as role and experience from socio-organizational factors such as conventions and mandates, purpose/requirements, and employed methods. In the following sections, we examine each of the identified factors in more detail.

### *Modeling Tools*

Process modeling grammars are often implemented, and used, in a modeling tool or even a business process management system (e.g., Dennis *et al.*, 1999). These tools provide extended functionality to support the way grammars can be deployed. For instance, some tools provide model repositories in which models can be stored and linked on different levels of abstraction. Other tools offer a variety of different grammars for process modeling, which, in turn, enable users to complement a grammar with constructs from another grammar if they encounter deficits in the original grammar (Green, 1996). In our study, we found that the modeling tool, in which BPMN is implemented, influences the way users perceive the ontological deficiencies (i.e., lack of completeness or clarity) of BPMN. Some of the deficiencies that exist in BPMN were overcome by means of tool functionality (model repository, meta-tags and links to other documents and grammars, for instance). As one of the interviewees stated:

> *"[…] If you can't afford a good tool, it could be a problem because just the basic set doesn't have it, so that's [inaudible] a problem. From my personal perspective."*
> (interview transcription data)

Hence it would appear that a tool can be used to mask and/or mitigate deficiencies that, *prima facie*, exist in modeling grammars. These findings suggest that, when a grammar is not implemented in a tool with additional modeling functionality, i.e., if BPMN is used in isolation, its deficiencies would manifest more strongly. For example, 75 percent of the interviewees who indicated a need to capture business rules were not able to do so with BPMN. However, some respondents did not consider this situation to be a major problem *only because* they overcame the deficit in BPMN with tool-based workarounds that included narrative descriptions of rules and conditions, spreadsheets, or hyperlinks to external tables and meta-tags. Figure 5 shows an excerpt from one of the interviews, in which it can be seen how the semi-structured interviewing approach allowed probing for detail in a case where an ontological deficiency of BPMN (construct deficit to articulate process structure and decomposition) was masked in the response due to the existence of alleviating tool functionality in the Casewise toolset.

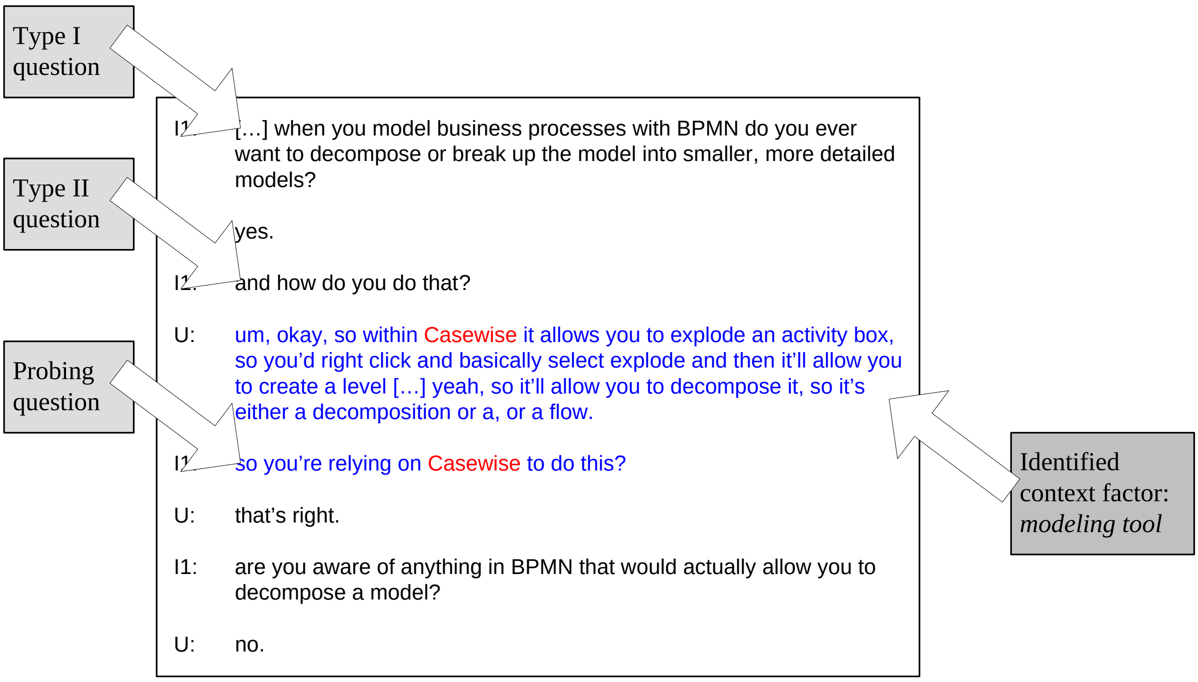


Figure 5: Interview excerpt: Probing for the intervening variable 'modeling tool'

Overall, Table 2 summarizes tool functionality that our interview results suggest to have an impact on how BPMN users would perceive a predicted deficiency of the grammar.

Table 2. Tool functionality influencing grammar usage perceptions

| **Tool functionality** | **Impact** |
|---|---|
| Integrated repository | An integrated model repository stores all models within a central database and facilitates navigation between process models on different levels of conceptual abstraction. Thereby, it may overcome ontological deficiencies within a process modeling grammar related to process decomposition and scoping. |
| Navigation capacity | Navigation capacity allows users to link and access process models from within other process models through hyperlinks. Thereby, it may mask deficiencies within a process modeling grammar related to process decomposition and scoping. |
| Additional attribute fields | Additional attribute fields and meta-tags for language constructs used within a process models allows a user to depict additional information about the context in which a grammar construct is used. Thereby, clarifying information can be annotated to a grammar construct, which may rectify concerns about the real-world meaning of the construct, thereby alleviating concerns about construct overload or redundancy. |
| Access to other modeling grammars | Integrated modeling tools such as System Architect or ARIS allow users to link process models with other conceptual models and to combine different models. Thereby, potential deficits within a grammar can be overcome by allowing the user to employ additional grammars to depict the real-world phenomena that could not be articulated with the original grammar. For instance, some interviewed BPMN users reported that they used UML state chart models to depict business rules. Thereby overcoming problems regarding construct deficit. |

| Tool functionality | Impact |
|---|---|
| Access to new grammar constructs | In advanced modeling tools, the user is allowed to define new or additional grammar constructs to be used in addition to an existing grammar. In the present study, some interviewees reported that they used new constructs in addition to regular BPMN constructs to explicitly describe IT system-supported tasks in a process model. This functionality may mitigate a lack of completeness in a grammar. |
| Hyperlinks to documentation | In some modeling tools, hyperlinks can be created within process models that provide access to additional documentation in the form of spreadsheets or documents. In the present study, for instance, this was often done to document business rules that could not be depicted with BPMN. |
| Method filter | A method filter restricts the set of grammar constructs, or even a set of grammars, to be used by process modelers. It reduces the apparent complexity of a grammar by limiting the user to a reduced set of constructs. While this may induce construct deficit in the modeling task, it may also reduce construct redundancy, overload or excess within a grammar. |

### *Modeling Conventions*

In modeling practice, users often do not employ the modeling grammar in its original or full version. A recent study by zur Muehlen and Recker (2008), which analyzed over 120 real-world BPMN models for frequency of construct use, deduced a core set ("common core") of just six BPMN constructs and an extended set ("extended core") of just six more BPMN constructs.

In practice, organizations often follow a set of modeling conventions that restrict the set of grammar constructs to be used. And indeed, of the six organizations studied, four were using specific modeling conventions. Modeling conventions can be seen as an organization-internal standardization of a grammar. These conventions specify the way the grammar is implemented and put to use for modeling. Two types of impact of the conventions can be differentiated. First, in some cases, modeling conventions *restricted* the use of BPMN to a reduced set of grammar constructs, which sometimes influenced the way BPMN modelers perceived some of the proposed deficiencies in BPMN. For example, this situation is indicated in the following response, which shows that the respondent deemed the organizationally restricted set of grammar constructs as problematic:

> *"[…] minor, because you just create the descriptions and they'll read it from there, but sometimes, when you're in a meeting with high level managers, the last thing they want to do is read descriptions. They want to look at it and understand it and walk away, so I'm definitely for more BPMN figures."* (interview transcription data)

Second, in some cases modeling conventions *amended* the BPMN specification. For instance, some organizations found deficits in BPMN for differentiating manual from IT system-supported tasks and introduced color-coding as a means to graphically articulate these two types of tasks:

> *"[…] we also used colors to shade the boxes to differentiate between the two, because some people didn't think that certain steps could be performed manually and certain steps can be performed on the system and they wanted to differentiate between both, so we used shading to show one versus the other."* (interview transcription data)

### *Modeler Experience, Role and Purpose*

Experience and role of the modeler, together with the purpose for which modeling is conducted, also appear to impact the usage of BPMN and the perception of its issues. Different individual abilities, and different requirements towards modeling, influence how, and which, BPMN constructs are used. An expert data modeler, for instance, exhibits higher degrees of innovation and adaptability in conceptual modeling (Shanks, 1997). As such, he or she is able, for instance, to revert to a repertoire of modeling 'workarounds' when encountering modeling problems that stem from a grammar deficiency. Similarly, an expert process modeler is more likely to have created many process models and he/she would have used a larger set of constructs. Hence, such an expert modeler is more likely to be aware of any weaknesses in their use. A novice modeler, on the other hand, may not have had to model a large variety of processes, may not have had the need to use all grammar constructs, and hence may be unaware of the weaknesses or may not have enough experience to recognize them as such.

The differences in process modeling expertise and the impact of experience levels on the usage of BPMN were clearly evident in the interviews. Table 3 summarizes the number of responses received for each proposition for three types of modelers, based on their reported level of modeling experience. We classified interviewees with less than two years of process modeling experience as modelers with limited experience (six in total), those interviews with between two and five years of modeling experience as having a medium level of experience (five in total), and those interviewees with more than five years of modeling experience as having an extensive level of experience (eight in total).

Table 3. Summary of responses to propositions, by modeler experience

| Deficiency type | Proposition | Participant level of experience | Response type I | II | III | IV | V |
|---|---|---|---|---|---|---|---|
| Construct deficit | P1 | limited | 0 | 3 | 0 | 3 | 0 |
| | | medium | 4 | 0 | 0 | 1 | 0 |
| | | extensive | 3 | 0 | 4 | 1 | 0 |
| | P2 | limited | 4 | 0 | 2 | 0 | 0 |
| | | medium | 4 | 0 | 1 | 0 | 0 |
| | | extensive | 4 | 1 | 2 | 1 | 0 |
| | P3 | limited | 1 | 0 | 3 | 2 | 0 |
| | | medium | 0 | 2 | 1 | 0 | 2 |
| | | extensive | 1 | 3 | 2 | 1 | 1 |
| Construct redundancy | P4 | limited | 1 | 3 | 2 | 0 | 0 |
| | | medium | 2 | 1 | 0 | 1 | 0 |
| | | extensive | 1 | 4 | 3 | 0 | 1 |
| | P5 | limited | 0 | 6 | 0 | 0 | 0 |
| | | medium | 0 | 4 | 0 | 0 | 1 |
| | | extensive | 0 | 8 | 0 | 0 | 0 |
| | P6 | limited | 0 | 3 | 1 | 2 | 0 |
| | | medium | 0 | 5 | 0 | 0 | 0 |
| | | extensive | 0 | 6 | 0 | 1 | 1 |
| Construct overload | P7 | limited | 2 | 1 | 2 | 1 | 0 |
| | | medium | 2 | 0 | 2 | 1 | 1 |
| | | extensive | 1 | 4 | 1 | 1 | 0 |
| | P8 | limited | 0 | 3 | 4 | 0 | 0 |
| | | medium | 0 | 2 | 3 | 0 | 0 |
| | | extensive | 1 | 3 | 2 | 1 | 0 |
| Construct excess (average) | P9 | limited | 1.25 | 0.25 | 1.17 | 0.5 | 2.58 |
| | | medium | 0 | 0.25 | 1.58 | 0.25 | 2.92 |
| | | extensive | 0.42 | 1.25 | 1.17 | 1.75 | 4.5 |

The data in Table 3 shows that there are differences in the responses received between the three types of modelers considered. Overall, across the construct deficit, redundancy and overload propositions, type V responses (perceived major problem) were only reported by modelers with a medium or extensive level of experience. Similarly, type III responses for these propositions (not a problem due to the existence of workarounds) were in 60 % of all cases reported by modelers with a medium or extensive level of experience. These findings suggest that more experienced modelers are increasingly aware of ontological deficiencies in a grammar, as well as in potential workarounds that could be used to mitigate these deficiencies. Indeed, several of the interviewees commented that, at the present stage of their modeling experience, they could only see minor consequences of the weakness for practical use, that is, “a nice feature to have.” Others explicitly stated that they were not yet at a level of modeling knowledge that would allow them to use BPMN in the way it was intended. As one interviewee stated:

> *“[…] I haven’t done enough with it. […] sometimes you don’t have this start and end, but because you need start and end to have these, it sort of looks nice but then you have this dependency between activities, so I don’t really know how to do that.”* (interview transcription data)

Notably, experienced modelers were more likely to consider the ‘excess’ constructs as essential to their BPMN modeling experience. We speculate that this situation might indicate that the experienced modelers are able to make better use of the constructs, relative to the less experienced modelers, to alleviate other BPMN shortcomings. However, the use of excess constructs requires further empirical study.

In conclusion, it appears that the level of modeling expertise indeed impacts the way process modeling tasks are performed and how users perceive, and use, a process modeling grammar.

A second important individual difference factor would appear to be the modeling role that a user occupies in a modeling initiative. Interviewees that occupied different roles (most notably, systems analysts versus business analysts) often reacted differently to the same set of propositions. Some of the modelers with technical background explicitly mentioned the divergence between their own background and the more business-oriented way that BPMN was used in the organization:

> *“[…] it’s a broad change in terms of the way that you’re thinking with BPMN, because it’s very high level. It’s pretty much [inaudible] what the business processes are. I’m working in sort of a semi-technical business area. I’m used to writing use cases a lot so,*

*at first I'm so used to detail and I have to change my mindset into writing it in higher level."* (interview transcription data)

Other interviewees conceded that they have not yet made use of certain BPMN constructs because their role in the modeling initiative was business-oriented and in that role they have not encountered a need for using some of the constructs:

*"[…] I haven't really, I suppose because I don't go down into the software, the actual software development side of describing the processes. I stop with just describing the business and the business requirements. I haven't had a need to describe anything other than actual people, in the business area."* (interview transcription data)

In terms of process modeling purpose, we uncovered that modelers who create BPMN models for business application areas (e.g., continuous improvement or knowledge management) were more likely to use a smaller set of constructs than modelers using BPMN modeling in technical application areas, e.g., process execution or simulation. Figure 6 shows the relative dominance of BPMN use (use of the *core set* versus use of an *extended or complete set* of BPMN constructs), in contrast to the primary application purpose of process modeling, as named by the interviewees. For example, Figure 6 shows that there is a net use of BPMN core constructs over that of the extended set of BPMN constructs by interviewees modeling for knowledge management purposes.

Figure 6 suggests that for typical business modeling purposes (e.g., organizational redesign, process documentation, continuous process management or knowledge management), the BPMN core element set was often deemed sufficient by the organizations in terms of providing an adequate set of BPMN constructs for process modeling. In contrast, in terms of more IT-oriented purposes (marked with an '*' in Figure 6) such as workflow management, selection and configuration of ERP software, process simulation or systems requirements specification, Figure 6 indicates that organizations tended to use more frequently, if not exclusively, an extended or complete set of BPMN constructs so as to add expressiveness to the process models.

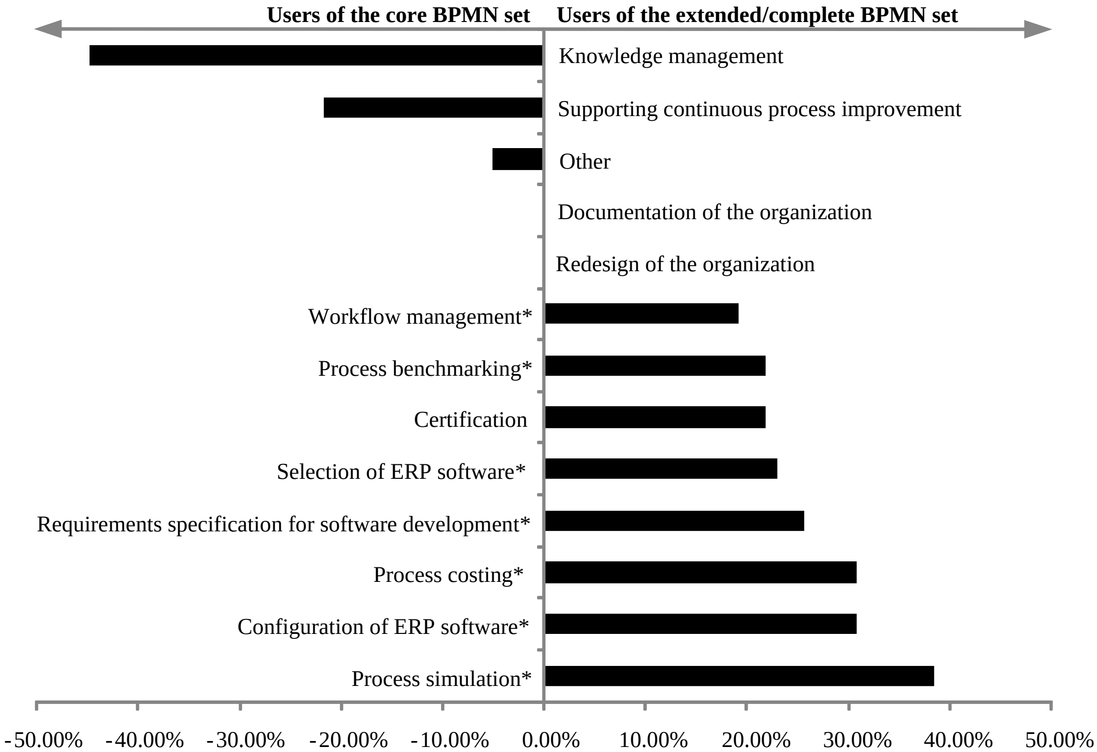


Figure 6: Relative dominance of BPMN use (core set versus extended/complete set) by process modeling purpose

# Discussion

This paper presented a comprehensive analysis of BPMN that incorporates both a theoretical model for evaluation and semi-structured interviews to gain insights on the generated propositions and to contextualize the results. Based on our empirical analysis, we identify three levels of support (unsupported, limited, apparent) for the nine propositions advanced in our study. Table 4 summarizes the propositions and the empirical insights gathered about these propositions. It also suggests corresponding implications about the theory of ontological expressiveness we used in our study, based on the qualitative judgment of the research team on the basis of content analysis of the interview data in relation to the theoretical arguments.

Table 4. Summary of interview results: Levels of proposition support

| **Deficiency type** | **No** | **Proposition** | **Level of support** | **Implication for theory** |
|---|---|---|---|---|
| Construct | P1 | Lack of means for capturing business | Apparent | Largely confirms |

| Deficiency type | No | Proposition | Level of support | Implication for theory |
|---|---|---|---|---|
| deficit | | rules | | theory propositions. |
| | P2 | Lack of means for capturing state histories | Limited | |
| | P3 | Lack of means for capturing process decomposition | Apparent | |
| Construct redundancy | P4 | Confusion over articulation of real-world objects | Limited | Largely refutes theory propositions. |
| | P5 | Confusion over articulation of state transformations | Unsupported | |
| | P6 | Confusion over articulation of events | Unsupported | |
| Construct overload | P7 | Clarification of the 'Lane' construct | Apparent | Confirms theory propositions. |
| | P8 | Clarification of the 'Pool' construct | Apparent | |
| Construct excess | P9 | Confusion about nature and purpose of some modeling constructs | Apparent (3 of 12)<br>Limited (3 of 12)<br>Unsupported (6 of 12) | Inconclusive. |

Our findings suggest that end users require capability in BPMN for the articulation of business rules that govern organizational processes, as well as for the specification of process structure and decomposition. These deficits motivate users to employ additional means for their modeling, thereby increasing complexity, and decreasing consistency, of their modeling. Also, end users struggle with the ambiguities of BPMN constructs Lane and Pool, and look for more guidance in the use of these constructs. We further found that end users manage redundancy in BPMN rather well and are able to cope with different constructs that share similar meanings. We also found interesting usage patterns relating to a number of support modeling constructs, such as text annotations and grouping constructs, which are used in organizations to further enhance the process specifications on an informal level. To that end, our study mostly confirmed the theoretical arguments advanced by Wand and Weber (1993) regarding the implications of construct deficit and construct overload on the utility of a modeling grammar. In contrast, the arguments regarding the issue of construct redundancy remained largely disconfirmed in our empirical study. Support for the arguments related to construct excess are inconclusive and require further investigation. We outline some directions for future research below.

In our study we uncovered, and examined, a range of personal, organizational and situational factors that impact process modeling practice. Through our interview analysis we uncovered the presence of five factors that inform a contextualized model of the use of graphical grammars for process modeling. First, modelers with different levels of *process modeling experience* range in terms of how many modeling scenarios they have encountered and also in their portfolio of modeling workarounds.

This experience, in turn, affects how these modelers perceive strengths and weaknesses of a grammar and it also affects how the modelers employ a grammar for process modeling. Second, the individual background of the process modelers (their *role in the modeling initiative*) is linked with their view of a process model – in accordance to their set of skills, their training background and their cognitive abilities. Third, the *purpose* for which a process model is to be used determines the set of real-world phenomena that is relevant to be modeled and thereby the way a grammar is being used to model real-world domains. Fourth, the act of process modeling is typically supported through a *modeling tool* that offers a set of advanced and extended features. The functionality of the tool can assist a process modeler in overcoming deficiencies existing in a grammar, which would prevent the modeler from perceiving them as problematic. Fifth, *modeling conventions* specify the way in which a grammar is put to use. They standardize organization-wide the semantics of the grammar. Thereby, certain deficiencies may not manifest due to amendments to the original specification of the grammar.

## A Call for further Research into the Practice of Process Modeling

Our study explored the manifestations and consequences of ontological deficiencies in practice, thereby advancing our understanding of the real-world challenges analysts encounter when using process modeling grammars for organizational documentation and process improvement. We identified (a) that ontological deficiencies exist in the current process modeling standard BPMN and (b) that these deficiencies force end users to mitigate these shortcomings through workarounds or other tools and techniques.

These findings suggest that process modeling as an organizational practice requires further attention in research, to advance the development of more effective and appropriate support for analysts in process improvement projects. Most notably, our work identifies a need for further development of the capabilities of process modeling grammars. The uncovered ontological deficiencies can trigger a number of related design science efforts to improve and extend current process modeling grammars such as BPMN. Aside from the ontological issues pertaining to overloaded, redundant and excessive construct specifications, we would like to point out the need for more development towards the integration of process modeling and business rule representations (zur Muehlen & Indulska, 2010). Our study shows that BPMN users would prefer to be able to describe visually in their process models where and how business rules affect the depicted processes, and they have trouble with identifying the interface between process modeling and business rule modeling using the BPMN grammar.

Second, we believe that our work serves both as motivation and input to the extension of research into the practice of process modeling. Some of the conjectures we derived from our exploratory examination (e.g., the dynamics between grammar-related modeling issues and task- and user-based intervening factors) call for appropriate empirical research strategies that further operationalise and test these propositions. We interviewed multiple BPMN users with various backgrounds from various organizations to overcome single case bias, however, we nonetheless identify the sample size of nineteen interviewees as a potential threat to the external validity of our findings. All of our interview participants were based in Australia. While we see no indications why Australian modelers would not be representative of process modelers in other settings, we would invite fellow studies to examine process modeling practice beyond this regional scope.

Yet, we believe that our study has uncovered a rich and comprehensive first explanation of process modeling grammar use that can stimulate and guide further empirical research in this emerging relevant domain of IS practice. At present, only little is known about process modeling practice and process modeling grammar usage overall. The work presented in this paper explores a range of situational factors that frame the contextual setting in which process modeling grammars are being put to use. Our work can be leveraged in the development and testing of research models aimed at explaining process modeling grammar adoption, usage, and eventually success.

Third, we believe that our results suggest that the theory of ontological expressiveness provides a fruitful basis from which insights into potential and actual issues with the use of a process modeling grammar can be obtained. Table 4 shows that the suggested implications of construct deficit and construct overload largely manifested in actual process modeling practice. The implications of construct redundancy, however, were largely unsupported in our interviews, suggesting that the process modeling community is able to deal with this deficiency. Propositions regarding the implications of construct excess are inconclusive and warrant further examination to clearly understand as well as eliminate potential problems in practice. Soffer and Wand (2007), for instance, examined compositions of excess constructs in process modeling grammars (most notably, the use of gateway constructs in flow pattern compositions), and showed that such compositions can form an ontologically meaningful construct. Future research could now examine whether modelers recognize ontologically meaningful compositions of excess constructs and thereby alleviate deficiencies attributed to the individual constructs.

Our findings further suggest that there is a need to consciously explore the setting in which a process modeling grammar is put to use and in which potential issues with the grammar may (or may not) become apparent. The theory of ontological expressiveness offers an explanatory framework to guide these investigations (the premises of which can be tested using quantitative data). While a focus on ontological deficiencies has been found to be useful to guide studies on modeling grammars (Green & Rosemann, 2004), we note that the use of the theory of ontological expressiveness provides a filtering lens to our study of BPMN. We only examined propositions (regarding construct deficit, redundancy, overload and construct excess) created in light of the premises of this theory. The use of other theoretical or ontological models could have led to different mapping results, and consequently different propositions. For instance, we identified excessive constructs in BPMN as per Wand and Weber's (1993) theory. Given the inconclusive results obtained through our empirical analysis, the question of whether or not these constructs lack "real-world" meaning as the theory suggests, remains to be answered. The use of empirical (preferably qualitative) data further increases the scope of the theory in that it provides evidence and contextual information about the insights that can be gained from the theory. Our study illustrated how the organizational context of process modeling can be examined in detail by using the explanatory framework offered by the theory of ontological expressiveness together with the use of qualitative data collection and thematic analysis. Specifically, our exploratory study uncovered the existence of a range of contextual factors such as modeling purpose, modeler role, or modeling tool, which influenced usability perceptions of BPMN. While we were able to detail some implications of these contextual factors (for instance, how a number of tool functionalities can be used to mask or mitigate ontological deficiencies in the BPMN grammar), further research is needed to examine how process modeling grammars are implemented in modeling tools, and how these tool implementations of a grammar, in conjunction with additional tool features, are being put to use by users in organizational modeling initiatives.

## Implications for Practice

Aside from a call for further development efforts to advance the capabilities of process modeling grammars, our work also informs the current practice of process modeling, by providing suggestions for how to deal with the current capabilities, and deficiencies, of the grammar in use. Several stakeholders in process modeling projects benefit from our study at present, until more advanced process modeling grammars become available.

First, to decision makers concerned with adopting BPMN, our results provide validated evidence of BPMN modeling issues of which modelers should be aware. For example, knowing that BPMN exhibits a limitation in the modeling of business rules, an organization should consider additional tools together with a set of business rule modeling conventions, or it should adopt a business rule modeling grammar. Such a move will help to ensure consistent modeling and prevent the need for correction of the models at a later stage.

Second, decision makers should consciously explore the organizational setting they establish for process modeling activities. For instance, the level of expertise of the process modelers as well as the existence of modeling conventions impact the way a process modeling grammar is put to use, and thereby influence the quality of the models produced.

Third, designers are guided in their efforts to develop grammars that allow end users to fully and effortlessly create descriptions of the real-world domains with which they are concerned. By showing in our study empirically which issues exist in a grammar and how these issues impact user evaluations, grammar developers should examine each of our identified and confirmed deficiencies closely, to identify grammar improvements to be incorporated in future releases, that will ultimately warrant not only popularity of the grammar but also instrumentality and productivity for process modeling.

Last, we uncovered that a large share of modeling practitioners (54 percent of respondents) did not undergo formal training in BPMN. We found that this lack of training partially explained some of the issues encountered when modeling with BPMN. We therefore posit that more advanced and more widely available training programs need to be developed, to offer more comprehensive, and specialized training in the use of BPMN.

# Conclusions

In this study, we contribute to process modeling research by providing a theoretical and empirical analysis of the ontological deficiencies of the BPMN grammar. Our study provides theoretical and empirical evidence on the strengths and weaknesses of using BPMN, and it also discusses a range of contextual factors that impact the use of BPMN. These findings assist an understanding about capabilities and issues in the use of BPMN, and, in turn, they can guide executive decision makers in their investment decisions when choosing to implement a process modeling initiative in an organization. In summation, our study has uncovered a rich and contextualized understanding of usage behavior associated with process modeling grammars. It has provided also further evidence for the utility of the ontological theory of grammar expressiveness to aid our understanding of process modeling practice.

# References

AAGESEN G and KROGSTIE J (2010) Analysis and design of business processes - using bpmn. In *Handbook on business process management* (ROSEMANN M and VOM BROCKE J, Eds), Springer.

BANDARA W, GABLE GG and ROSEMANN M (2005) Factors and measures of business process modelling: Model building through a multiple case study. *European Journal of Information Systems* 14(4), 347-360.

BENBASAT I, GOLDSTEIN DK and MEAD M (1987) The case research strategy in studies of information systems. *MIS Quarterly* 11(3), 369-388.

BODART F, PATEL A, SIM M and WEBER R (2001) Should optional properties be used in conceptual modelling? A theory and three empirical tests. *Information Systems Research* 12(4), 384-405.

BOWEN PL, O'FARRELL RA and ROHDE F (2006) Analysis of competing data structures: Does ontological clarity produce better end user query performance. *Journal of the Association for Information Systems* 7(8), 514-544.

BOYATZIS RE (1998) *Thematic analysis: Coding as a process for transforming qualitative information.* Sage, Thousand Oaks, California.

BPMI.ORG and OMG (2006) Business process modeling notation specification. Final adopted specification. Object Management Group.

BUNGE MA (1977) *Treatise on basic philosophy volume 3: Ontology i - the furniture of the world.* Kluwer Academic Publishers, Dordrecht, The Netherlands.

BURTON-JONES A and MESO P (2006) Conceptualizing systems for understanding: An empirical test of decomposition principles in object-oriented analysis. *Information Systems Research* 17(1), 38-60.

COHEN J (1960) A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* 20(1), 37-46.

CURTIS B, KELLNER MI and OVER J (1992) Process modeling. *Communications of the ACM* 35(9), 75-90.

DAVIES I, ROSEMANN M and GREEN P (2004) Exploring proposed ontological issues of aris with different categories of modellers. In *15th Australasian Conference on Information Systems*, Australian Computer Society, Hobart, Australia.

DEBEVOISE T and GENEVA R (2008) *The microguide to process modeling in bpmn.* BookSurge Publishing, Charleston, South Carolina.

DENNIS AR, HAYES G and DANIELS RM (1999) Business process modeling with group support systems. *Journal of Management Information Systems* 15(4), 115-142.

DIJKMAN RM, DUMAS M and OUYANG C (2008) Semantics and analysis of business process models in bpmn. *Information and Software Technology* 50(12), 1281-1294.

DREILING A, ROSEMANN M, VAN DER AALST WMP, HEUSER L and SCHULZ K (2006) Model-based software configuration: Patterns and languages. *European Journal of Information Systems* 18(6), 583-600.

DUBÉ L and PARÉ G (2003) Rigor in information systems positivist case research: Current practices, trends, and recommendations. *MIS Quarterly* 27(4), 597-635.

EDELMAN S (1998) Representation is representation of similarities. *Behavioral and Brain Sciences* 21(4), 449-498.

GARTNER GROUP (2010) *Leading in times of transition: The 2010 cio agenda.* EXP Premier Report No January2010, Gartner, Inc, Stamford, Connecticut.

GEMINO A and WAND Y (2005) Complexity and clarity in conceptual modeling: Comparison of mandatory and optional properties. *Data & Knowledge Engineering* 55(3), 301-326.

GREEN P (1996) An ontological analysis of isad grammars in upper case tools. The University of Queensland, Brisbane, Australia.

GREEN P and ROSEMANN M (2000) Integrated process modeling. An ontological evaluation. *Information Systems* 25(2), 73-87.

GREEN P and ROSEMANN M (2001) Ontological analysis of integrated process models: Testing hypotheses. *Australasian Journal of Information Systems* 9(1), 30-38.

GREEN P and ROSEMANN M (2004) Applying ontologies to business and systems modeling techniques and perspectives: Lessons learned. *Journal of Database Management* 15(2), 105-117.
GREEN P, ROSEMANN M and INDULSKA M (2005) Ontological evaluation of enterprise systems interoperability using ebxml. *IEEE Transactions on Knowledge and Data Engineering* 17(5), 713-725.
GREEN P, ROSEMANN M, INDULSKA M and MANNING C (2007) Candidate interoperability standards: An ontological overlap analysis. *Data & Knowledge Engineering* 62(2), 274-291.
HITCHMAN S (1995) Practitioner perceptions of the use of some semantic concepts in the entity-relationship model. *European Journal of Information Systems* 4(1), 31-40.
KOVACIC A (2004) Business renovation: Business rules (still) the missing link. *Business Process Management Journal* 10(2), 158-170.
KROGSTIE J, SINDRE G and JØRGENSEN HD (2006) Process models representing knowledge for action: A revised quality framework. *European Journal of Information Systems* 15(1), 91-102.
LEE J, WYNER GM and PENTLAND BT (2008) Process grammar as a tool for business process design. *MIS Quarterly* 32(4), 757-778.
MOORE GC and BENBASAT I (1991) Development of an instrument to measure the perceptions of adopting an information technology innovation. *Information Systems Research* 2(3), 192-222.
MYERS MD and NEWMAN M (2007) The qualitative interview in is research: Examining the craft. *Information and Organization* 17(1), 2-26.
OUYANG C, VAN DER AALST WMP, DUMAS M, TER HOFSTEDE AHM and MENDLING J (2009) From business process models to process-oriented software systems. *ACM Transactions on Software Engineering Methodology* 19(1), 2-37.
PARSONS J and WAND Y (2008) Using cognitive principles to guide classification in information systems modeling. *MIS Quarterly* 32(4), 839-868.
RECKER J (2010a) Continued use of process modeling grammars: The impact of individual difference factors. *European Journal of Information Systems* 19(1), 76-92.
RECKER J (2010b) Opportunities and constraints: The current struggle with bpmn. *Business Process Management Journal* 16(1), 181-201.
RECKER J and INDULSKA M (2007) An ontology-based evaluation of process modeling with petri nets. *Journal of Interoperability in Business Information Systems* 2(1), 45-64.
RECKER J, ROSEMANN M, INDULSKA M and GREEN P (2009) Business process modeling: A comparative analysis. *Journal of the Association for Information Systems* 10(4), 333-363.
RECKER J, ROSEMANN M and KROGSTIE J (2007) Ontology- versus pattern-based evaluation of process modeling languages: A comparison. *Communications of the Association for Information Systems* 20(48), 774-799.
ROSEMANN M (2006) Potential pitfalls of process modeling: Part a. *Business Process Management Journal* 12(2), 249-254.
ROSEMANN M, RECKER J, GREEN P and INDULSKA M (2009) Using ontology for the representational analysis of process modeling techniques. *International Journal of Business Process Integration and Management* 4(4), 251-265.

SANDELOWSKI M (2001) Real qualitative researchers do not count: The use of numbers in qualitative research. *Research in Nursing & Health* 24(3), 230-240.
SCHEER A-W (2000) *Aris - business process modeling.* Springer, Berlin, Germany.
SHANKS G (1997) Conceptual data modelling: An empirical study of expert and novice data modellers. *Australasian Journal of Information Systems* 4(2), 63-73.
SHANKS G, TANSLEY E, NUREDINI J, TOBIN D and WEBER R (2008) Representing part–whole relations in conceptual modeling: An empirical evaluation. *MIS Quarterly* 32(3), 553-573.
SOFFER P, WAND Y and KANER M (2007) Semantic analysis of flow patterns in business process modelling. In *Business process management - bpm 2007* (ALONSO G, DADAM P and ROSEMANN M, Eds), pp 400-407, Springer, Brisbane, Australia.
VAN DER AALST WMP, BENATALLAH B, CASATI F, CURBERA F and VERBEEK HMV (2007) Business process management: Where business processes and web services meet. *Data & Knowledge Engineering* 61(1), 1-5.
WAND Y and WEBER R (1993) On the ontological expressiveness of information systems analysis and design grammars. *Journal of Information Systems* 3(4), 217-237.
WAND Y and WEBER R (2002) Research commentary: Information systems and conceptual modeling - a research agenda. *Information Systems Research* 13(4), 363-376.
WEBER R (1997) *Ontological foundations of information systems.* Coopers & Lybrand and the Accounting Association of Australia and New Zealand, Melbourne, Australia.
WEBER RP (1990) *Basic content analysis.* Sage, Newbury Park, California.
WHITE SA and MIERS D (2008) *Bpmn modeling and reference guide.* Lighthouse Point, Florida, Future Strategies.
WOLF C and HARMON P (2008) *The state of business process management - 2008.* www.BPTrends.com.
YIN RK (2003) *Case study research: Design and methods.* Sage Publications, Thousand Oaks, California.
ZUR MUEHLEN M and INDULSKA M (2010) Modeling languages for business processes and business rules: A representational analysis. *Information Systems* 35(4), 379-390.
ZUR MUEHLEN M and RECKER J (2008) How much language is enough? Theoretical and practical use of the business process modeling notation. In *Advanced information systems engineering - caise 2008* (LÉONARD M and BELLAHSÈNE Z, Eds), pp 465-479, Springer, Montpellier, France.

## *Appendix: Results from the Ontological Analysis of BPMN*

| BWW Construct | BPMN Construct | Reasoning behind mapping |
|---|---|---|
| THING | Lane, Pool | Both a Pool and a Lane can represent specific participants (organizational units or persons) in a BPD. |
| PROPERTY | | |
| IN GENERAL | Attributes of Pools, Attributes of Lanes | Both the Pool and the Lane construct in BPMN have Attributes that capture the properties in general of the thing they represent. An example of this is the Name of a Lane (which can, for instance, be instantiated with the name of a stakeholder involved in a business process), the parent organizational structure in which the stakeholder works (parentPool) or the name of the super-ordinate organizational entity (Participant). The Attributes concept provided in BPMN, however, must be instantiated for every Pool and Lane in a BPD. |
| IN PARTICULAR | N/A | |
| HEREDITARY | N/A | |
| EMERGENT | N/A | |
| INTRINSIC | N/A | |
| MUTUAL: NON-BINDING | N/A | |
| MUTUAL: BINDING | N/A | |
| ATTRIBUTES | N/A | |
| CLASS | Lane, Data Object | A Data Object represents a document that is used as input or created as output during the course of a process. This can be an invoice, for example. However, the Data Object does not represent a specific object or thing but rather a type of document that can be instantiated in a specific instance of a process (e.g., invoice no. 4711).<br>The BPMN construct Lane can be nested, in which case Lanes share a common property (i.e., parentLane). When used in this manner, a Lane can be used to represent a group (i.e., class) of things such as departments or people (e.g., managers). |
| KIND | Lane | A Lane can be nested within another Lane that, as per definition, belongs to a Pool. Such a Lane would then have two properties common to other Lanes (i.e., parentLane and parentPool), which in turn makes it a kind of a thing, i.e., a specific sub-type of the concept Lane. |
| STATE | N/A | |
| CONCEIVABLE STATE SPACE | N/A | |
| STATE LAW | N/A | |
| LAWFUL STATE SPACE | N/A | |
| EVENT | Start Event, | The BPMN constructs Start Event, Intermediate Event and End Event allow for the modeling of certain triggers for a |

| BWW Construct | BPMN Construct | Reasoning behind mapping |
|---|---|---|
| | Intermediate Event, End Event Message, Timer, Error, Cancel, Compensation, Terminate | certain action to follow in a BPD. A Message can either be a start or an end message. In both cases this denotes a concept that evokes a transition between states of a thing. For instance, an arriving message could cause a process to cancel. Similarly, a message detailing a change request would lead to a change in an invoice document. A Timer is an event that, at a given point of time, triggers a certain action (such as, for instance, sending a follow-up note to a customer or canceling an order due to missing payment). Likewise, an Error is an event that may arise and that requires a particular action to be taken (namely, to cancel a process and to perform a rollback of related transactions if necessary). Cancel, Compensation and Terminate are all considered events that can arise in a thing given a particular action of a thing (here, a cancel request, a termination request, or a compensation request for a particular process scenario). |
| CONCEIVABLE EVENT SPACE | N/A | |
| LAWFUL EVENT SPACE | N/A | |
| TRANSFORMATION | Activity, Task, Collapsed Sub-Process, Expanded Sub-Process, Nested Sub-Process, Transaction | Each of Activity, Task, Transaction, Collapsed Sub-Process, Expanded Sub-Process, Nested Sub-Process are constructs that allow for the representation of a mapping of a thing from one state to another. For instance a refund sub-process will take a thing (e.g., a person) from one state to another (e.g., from a state of being poor to a state of being wealthy). An Activity is the same as a Task, both are concepts used to express how to perform certain action that lead to state changes (e.g., the task "approve credit card application" leads to changes in the status of the application, such as, for instance, from "in progress" to "rejected".) A Transaction is simply a special type of activity as it specifies those actions that are controlled through a transaction protocol (such as BTP or WS-transaction). |
| LAWFUL TRANSFORMATION | Default Flow, Uncontrolled Flow, Exception Flow | The BPMN constructs Default Flow, Uncontrolled Flow and Exception Flow are directed arcs that show the order of activities that will be performed in a process. They explicitly dictate what task is allowed after a certain action has occurred. They specify the legal order of tasks that can be performed at any given point and in turn the events that are lawful to occur subsequent to a given action in a process. |
| STABILITY CONDITION | Rule<br>Conditional flow | The BPMN constructs Rule and Conditional Flow both embody the specification of a transformation by means of a condition expression that is to be evaluated. A Rule is basically an expression that evaluates some process data at runtime to determine whether a Sequence Flow is being activated or not. A Conditional Flow is basically a Sequence Flow with an extra condition expression that is evaluated at runtime to determine whether or not the flow will be used. |
| CORRECTIVE ACTION | 'exception task', compensation activity | The 'Exception Task' in BPMN is a task that is linked to the Exception Flow mechanism and specifies what to do when the Exception Flow is triggered. Both this Exception Task and the Compensation Activity construct in BPMN represent types of lawful transformation and express behavior linked to a certain execution condition. |
| HISTORY | N/A | |
| ACTS ON | Message Flow | The Message Flow construct in BPMN depicts the interactions between participants of a process and indicates the direction of the interaction (e.g., from a supplier to a vendor). |
| COUPLING | Message Flow | The Message Flow construct in BPMN further contains association attributes connecting source and target object in a relationship. Thereby, it affords the representation of Coupling. |
| SYSTEM | Pool, Lane | The BPMN construct Pool describes different participants and their processes within a given process. |

| BWW Construct | BPMN Construct | Reasoning behind mapping |
|---|---|---|
| | | A Lane may be nested or defined in a matrix. In these cases the Lane construct represent a set of things between which couplings exist. |
| SYSTEM COMPOSITION | Pool, Lane | A Pool is composed of Lanes that define all participants within a Pool, which corresponds to defining all things within a system. As a Lane may be nested it may further have things in its own composition. |
| SYSTEM ENVIRONMENT | Pool, Lane | Within one BPD, it is possible to make use of several Pools (e.g., to model business-to-business interactions). Within such a BPD, a Pool outside of another Pool would depict the things not in the system of the other Pool. Along similar lines, different nested Lanes within a Pool or Lane can represent different sets of process participants (e.g., different departments), so one Lane would mark the environment of the other Lane or of another Pool. |
| SYSTEM STRUCTURE | N/A | |
| SUBSYSTEM | Pool, Lane | Given that multiple Pools and Lanes are allowed in a BPD, each Pool that represents a process partner or participant (e.g., one of several organizations participating in an inter-organizational process scenario) in a multi-pool BPD is in its essence a subsystem of the super-ordinate system represented by the BPD. A Lane is by definition a subset of a parent Pool. |
| SYSTEM DECOMPOSITION | Pool, Lane | A Pool in a multi-pool BPD defines a system within an (inter-organizational) system, thereby graphically articulating the decomposition of the system. The same holds by definition for Lanes used in Pools. |
| LEVEL STRUCTURE | Pool, Lane | Different Pools in a BPD define the sub-structure of an inter-organizational process and allow for differentiation between the hierarchy of these participants (for instance, by using a black box versus a white box approach).<br>Lanes can be nested, which in turn allows for the explicit graphical specification of the hierarchical structure of the systems expressed by the Lanes. |
| STABLE STATE | N/A | |
| UNSTABLE STATE | N/A | |
| EXTERNAL EVENT | Start Event, Intermediate Event, End Event, Message, Timer, Error, Cancel, Compensation | All of the Event subtypes (Start, Intermediate, End) can be external or internal, depending on the context of their use. A Message, for instance, may be an environmental component when sent by a customer outside the considered system (i.e., the process) or internal when sent from another process participant contained in the process description. Along similar lines, an Error, Cancel or Compensation event may arise due action external to the considered process or internal to the process (e.g., a compensation or cancel request from the customer on the one hand and from an internal department or process stakeholder within the organization on the other hand). A Timer is an external event as it denotes a concept to visualize how a change of state is incurred due to virtue of time (which, per definition, is a concept external to all systems). A Terminate event, on the other hand, is a form of internal event as it denotes a visualization concept to demarcate how a process can be stopped (without consideration of consequences) by virtue of action of internal stakeholders (e.g., process owners or process managers) but not by environmental components or stakeholders. |
| INTERNAL EVENT | Start Event, Intermediate Event, End Event, Message, Error, Cancel, Compensation, Terminate | |
| WELL-DEFINED EVENT | Compensation, End Event | In BPMN, the Compensation construct (in connection with a Compensation Association) is used to indicate that compensation is necessary. It triggers a defined sub-process with a specified transformation leading to a certain defined state (i.e., it specifies exactly how transaction that occurred during the course of a process have to be roll- |

| BWW Construct | BPMN Construct | Reasoning behind mapping |
|---|---|---|
| | | backed in order to arrive at the state of the process prior to enactment of the transaction that have been requested to be compensated). Similarly, an End Event is an indication of the completion of a process. As such, it marks a point where the state of a thing is changed to its final state. Hence, the state of any thing after the occurrence of this event can always be predicted (simply because it remains unchanged within this particular process). |
| POORLY-DEFINED EVENT | Message, Timer, Error, Cancel, Terminate, Start Event, Intermediate Event | A (part of a) process that relies on a Message to arrive cannot be predicted in its behavior due to the uncertainty of the actual content of the message. For instance, it is impossible to predict whether a customer note details a request to cancel a purchase order or to add another item to the order. The same holds in principle for the uncertainty of occurrence of an Error, Cancel or Terminate. In all of those cases the definition of the subsequent state is indeterminate as it is impossible to uniquely ascertain the occurrence of these types of event. Start Event and Intermediate Event are in their essence event subtypes that may resemble any specific event. Thereby, they are per definition poorly-defined as subsequent transformations and states cannot be predicted due to lack of information. |
| *ONTOLOGICAL EXCESS* | Link, Off-Page Connector, Gateway Types, Association Flow, Text Annotation, Group, Activity Looping, Multiple Instances, Normal Flow, Event (super type), Gateway (super type) | The Link and Off-Page Connector constructs in BPMN are graphical mechanisms for connecting processes that cross the boundaries of one or several documents. They do not bear any representational meaning.<br>Similarly, the constructs Association Flow, Text Annotation and Group are mechanisms to further annotate any object in a BPD with additional information.<br>Activity Looping and Multiple Instances are graphical representations that depict a composed series of transformations but not a transformation as such.<br>The BPMN Gateway sub types are merely graphical elements. All required conditions as to the branching and merging of processes have to be specified in the following Sequence Flows but not in the Gateway itself (specifically, each Gateway must have an associated Sequence Flow, which must have its Condition attribute set to Expression and must have a valid ConditionExpression).<br>Normal Flow, Event (super type), and Gateway (super type) are classes of constructs that are in the specification subdivided into different modeling constructs with specific semantics and therefore do not have dedicated semantic meaning in a process model. |