Business Process and Business Rule Modeling: A Representational Analysis

Michael zur Muehlen
Stevens Institute of Technology
mzurmuehlen@stevens.edu

Marta Indulska
The University of Queensland
m.indulska@business.uq.edu.au

Gerrit Kamp
Stevens Institute of Technology
gkamp@stevens.edu

Abstract

Process modeling and rule modeling languages are both used to document organizational policies and procedures. However, little work has been done to understand their synergies and overlap. Understanding the relationship between the two modeling types would allow organizations to maximize synergies and reduce their modeling effort. In this paper we use the well-established Bunge-Wand-Weber (BWW) representation theory to compare the representation capabilities of both types of languages. We perform a representational analysis of two rule modeling languages, viz., SRML and SBVR. We compare their representation capabilities with those of four popular conceptual business process modeling languages, and focus on the aspects of maximum ontological completeness and minimum ontological overlap. The outcome of this study shows that no single language is internally complete with respect to the BWW representation model and that a combination of two languages, viz. SRML and BPMN, is better suited for process modeling than any single modeling language.

1. Introduction

Business Process Management (BPM) has been identified as the number one priority of CIOs for the last few years [1]. Organizations are increasingly interested in improving their core process portfolio, and in identifying and quantifying processes with outsourcing potential.

Both process modeling languages and rule modeling languages offer constructs to represent work operations and constraints. This situation presents a selection dilemma for organizations where little guidance exists. While a significant amount of work has been done in terms of evaluating the representational capability of process modeling languages [2], the evaluation of rule modeling languages has received considerably less attention. Furthermore, recent empirical research has identified representational weaknesses in process modeling languages and led to speculation that business rule modeling languages might fill this gap [3]. It is an open question whether the two language types should be used in combination in order to increase the representation capability for process modeling. There is a need for a rigorous analysis of the overlap of the two types of languages in order to identify their potential synergies.

Therefore, the main *goal* of the work we present in this paper is to investigate the representation capability of two rule modeling languages, *viz.,* the Simple Rule Markup Language (SRML) and Semantics of Business Vocabulary and Business Rules (SBVR), and to place this evaluation in the context of previous representation capability evaluations of conceptual process modeling languages. Our evaluation of representation capability is based on the well-established Bunge-Wand-Weber (BWW) representation theory. Our two *research questions* are:

RQ1: What is the representational capability, with respect to the BWW representation theory, of SRML and SBVR?

RQ2: Are SRML and SBVR complementary or substitutive to process modeling languages?

The remainder of this paper is structured as follows. In the next section we present a brief review of business rules and business processes. The section also provides a review of related work on the integration of the two approaches and discusses recent empirical studies that identified deficiencies in process modeling languages. Section 3 presents the justification for the use of the BWW representation theory as a suitable benchmark for the analysis of representation capabilities of process and rule modeling languages. The following section describes the research methodology adopted in this work and provides a justification for the selection of languages under consideration. In section 5, we present a summary of the results of the BWW-based representation analysis of

SRML and SBVR and discuss the results of the analysis in light of representation capabilities of process modeling languages. We conclude the paper in section 6 with a discussion of limitations and future work directions in this area.

2. Background

While, to the best of our knowledge, no representational evaluation of rule modeling languages has been carried out, the background related to this work exists in the documented attempts at the integration of rule- and process-based modeling approaches, some surveys/comparisons of different approaches to business rule specification, as well as empirical studies investigating the weaknesses of process modeling languages.

2.1 Business Rules

A business rule is a statement that aims to influence or guide behavior and information in an organization [4]. Business rules can be categorized in accordance to their source or structure:

- *Mandates;* published policies that must be followed, or consequences will ensue. Examples are the payment of taxes and adherence to the law.
- *Policies;* published standards that should be followed to adhere to acceptable company behavior. Examples are budgets or mission statements.
- *Guidelines;* rules that may or may not apply, depending on circumstances. Examples are methodologies and management styles.

The different structural categories of business rules are [5]:

- *Integrity* (or constraints); For example: each project must have one and only one project manager.
- *Derivation* (conditions resulting in conclusions); For example: platinum customers receive a 5% discount. John Doe is a platinum customer. As a conclusion, John Doe receives a 5% discount.
- *Reaction* (Event, Condition, Action, Alternative-action, Post-condition); For example: an invoice is received. If the invoice amount is more than $1,000 then a supervisor must approve it.
- *Production* (condition, action); For example: if there are no defects in the last batch of cars then the batch is approved.
- *Transformation* (change of state); For example, an employee's age can change from 30 to 31, but not from 31 to 30.

Business rule modeling languages are typically based on formal logic and have strong and precise expressive power [6]. In general, they belong to the declarative modeling category in that they focus on specifying *what* is required to take place, rather than *how* something is accomplished.

2.2 Business Processes

Business Processes are sets of activities that create value for a customer. Organizations are increasingly automating processes using workflow systems, and are building elaborate management systems around their processes. Such management infrastructures integrate modeling, automation, and business intelligence applications.

A variety of modeling languages exists for the specification of process models; they can be classified according to their focal modeling construct:

- *Activity-centered;* processes as a network of tasks or activities.
- *Process object centered*; processes as the legal sequence of state changes of the process object.
- *Resource centered*; process as a network of processing stations that interact with each other.

Process languages appear as Graph-based languages (e.g. BPMN, EPC), Net-based languages (e.g. Petri-nets, flow nets) and Workflow Programming Languages (e.g. BPEL). Such languages, in general, are considered to be of the procedural modeling type, in that they focus on specifying the step-by-step activities that are required to take place in order to perform an action. While they do not provide the same level of precision or formalism as rule modeling languages, their strengths stem from their relative user friendliness and structural properties [6].

2.3 Integration of Business Rules and Business Processes

Early work on the integration of business rules and business processes started appearing shortly after the introduction of the rule modeling concept [7, 8]. Krogstie *et al.* [9] were the first to suggest that business process and rule modeling approaches should be merged to improve the capturing of temporal information for information systems development. They presented a top-down approach for model specification that involves the use of the External Rule Language (ERL) for specification of process logic at the lowest level of decomposition. This concept was further enhanced by McBrien and Seltveit [6], who presented a way to define the structure of rules within the process model. Knolmayer *et al.* [8] introduced a framework where process modeling is refined and linked to workflow execution through some layers of

Reaction Business Rules. Kappel *et al.* [7] use Reaction Business Rules to model the coordination in workflow systems. Kovacic [10] developed a meta-model that represents important business constructs (goal, process, activity and events) and technical constructs (data objects, software components, actions in Information Systems). He demonstrates how rules can link these two categories of constructs. Charfi [11] argues that business rules are often hard-coded into web services and proposes a hybrid approach of separating business processes and business rules. Meng *et al.* [12] introduced a dynamic workflow management system for modeling and controlling the execution of inter-organizational business processes. The system uses an event and rule server to trigger business rules during the enactment of workflow processes in order to enforce business constraints and policies at run-time.

While the integration of the two approaches has been the subject of some early investigation in the research community, anecdotal evidence shows that organizations struggle with effectively capturing business processes and rules. In a recent empirical study of the representational capabilities of Business Process Modeling Notation BPMN [3] we found that organizations supplement their BPMN process models with textual annotations of business rules. This practice introduces problems with rule consistency, reuse, and enforcement – problems that are acknowledged by some of the organizations making use of the textual business rule annotations.

While the need to improve the representation of business rules within process model diagrams is apparent, little is known about which representation aspects, if any, are unique to each of the two types of modeling languages. Previous work by Recker *et al.* [13] has identified a general lack of process modeling language capabilities to adequately model business rules. Similarly, Green and Rosemann [14] found limitations with respect to modeling business rules in their BWW-based investigation of all five views of Architecture of Integrated Information Systems (ARIS), a popular framework for integrated process modeling.

Rule modeling languages are likely candidates to fill such gaps. Indeed, an earlier study by Herbst *et al.* [15] suggests that rule specification languages should be considered as a potential addition to graphical representation languages when modeling for Information Systems (IS) design. While their analysis is not based on any formal framework, they find that many of the popular IS modeling techniques lack the ability to adequately represent business rules. The work of Rosemann *et al.* [2] suggests that the same approach of incorporating business rule modeling languages might apply in the process modeling domain.

In order to effectively integrate graphical business process modeling approaches with business rule modeling approaches, one must understand their synergies and overlap. At the time of writing, the authors are not aware of any attempts to evaluate the expressiveness of rule modeling languages, nor their relationships to conceptual process modeling approaches. The only related work appears to be that of Lu and Sadiq [16], who carried out a comparison of graph-based and rule-based modeling approaches. No specific rule modeling languages were considered as the work was focused on workflow modeling in particular rather than conceptual modeling in general. The authors used a set of workflow patterns as a basis for the evaluation, and found, among other results, that rule- and graph-based modeling approaches had similar levels of expressiveness in terms of the control flows specified by the workflow patterns.

Accordingly, there is a need to provide research and practice with theory-backed guidance as to which rule modeling language provides the best representational power.

3. Representation Theory

An ontology, or representation theory, can be used as a benchmark to make predictions about the capabilities of a grammar to provide complete and clear representations of a real-world domain [17]. The application of an ontology for such a purpose is known as *representational analysis*.

Representational analysis is performed by comparing the constructs of the chosen representation theory with the constructs of the modeling grammar and by identifying any representation equivalence between these. Any deviation from a one-to-one mapping relationship between these constructs indicates potential representational deficiencies in the grammar. Two principal evaluation criteria are *ontological completeness*, i.e., the extent to which the modeling grammar has a deficit of constructs that map to the set of representation theory constructs, and *ontological clarity*, i.e., the extent to which the modeling grammar constructs are deemed overloaded, redundant, or excessive [18]. These criteria provide a theoretical basis on which conceptual modeling languages can be compared with regard to their completeness of representation and clarity.

In this study we use the Bunge-Wand-Weber (BWW) ontology [17], specifically the representation model, since it is understood to contain all necessary constructs to describe things, and the interaction between things, in the real world. While the BWW model is not without its criticisms (for example [19]) and other ontologies could also be applied (for example Chisolm's ontology [20] or the Enterprise Ontology [21]), several studies have shown that BWW is a good basis to study the representational capabilities of conceptual modeling languages [13, 14]. This suitability has been empirically demonstrated in a large number of cases, as summarized in [3]. The choice of BWW representation theory allows us to compare our results to previous BWW-based analyses of process modeling languages.

The BWW representation model consists of some 40 higher-level abstract constructs, which can be grouped into four categories: Things and their Properties, States of a Thing, Events and Transformations, and Systems and their Composition. If a process or rule modeling language construct is found to have a representation for each of the BWW representation model constructs then that language fulfills all the representation requirements criteria necessary to model things and their interactions in the real world (with respect to the BWW representation model), without limiting the user's representation capabilities. While this may be the case, the language may still suffer from lack of clarity (e.g. an overload of constructs), which impacts its usability. A language that is complete and has the lowest levels of construct overload, redundancy and excess should be chosen. When no one language provides the required representation capability in terms of completeness, Green *et al.* [24] show that users will make use of combinations of languages that allow them to obtain maximum representation capability. Green *et al.* [22] discuss two theories for selecting two or more grammars for Information Systems modeling. The first, Maximum Ontological Completeness (MOC), states that users will select combinations of languages that, together, afford them the maximum possible representation power for their domain, i.e., if more constructs from an underlying ontology are incorporated in the chosen grammar, the expressive power of the resulting language combination will be higher. The second, Minimum Ontological Overlap (MOO), states that, when selecting languages to satisfy MOC, users will prefer languages with minimum overlap in ontological constructs, i.e., language combinations where no more than one grammatical construct maps to one BWW construct. Higher levels of construct overlap will create confusion and conflict in the work of the users. Together, the application of the MOC and MOO theories is known as *overlap analysis* and is done with the intended purpose of identifying language combinations with highest expressive power but low construct overlap.

We make use of the BWW representation model, together with the conduct of overlap analysis, in order to analyze the representational capabilities of SRML and SBVR, and identify combinations of business rule and business process modeling languages that are likely to be used in combination.

4. Methodology

4.1 Selection and Analysis of Rule Modeling Languages

A variety of rule languages have been developed over the past decade, as shown in Figure 1. For our initial study we selected two rule modeling language specifications that provided a comprehensive explanation of their vocabularies. The Simple Rule Markup Language (SRML) was selected as a representative example of a rule modeling language with a small vocabulary. A clear definition of its constructs is available and is not based on any other vocabulary [23]. The Semantics of Business Vocabulary and Business Rules (SBVR) was selected since it represents an attempt at the definition of a standardized rule modeling vocabulary [24]. SBVR presents a vocabulary that is intended to become a standard upon which many grammars can be based. For this reason its inclusion in the ontological analysis is useful.

In order to reduce subjectivity and increase internal validity of our research, we employed the extended representational analysis methodology as suggested by Rosemann *et al.* [18]. We followed the reference methodology as closely as possible. In particular, to increase objective comparison, the ERD BWW meta-model was obtained from the authors of [18] and an ERD meta-model of the SRML language was created to guide the mapping between SRML. SBVR was not transformed into an ERD diagram because the SBVR specification contains many UML diagrams of the language constructs, which were sufficient for a thorough understanding of the language.

The three researchers independently conducted the representation analysis of the languages. The independent analyses were followed by coordination

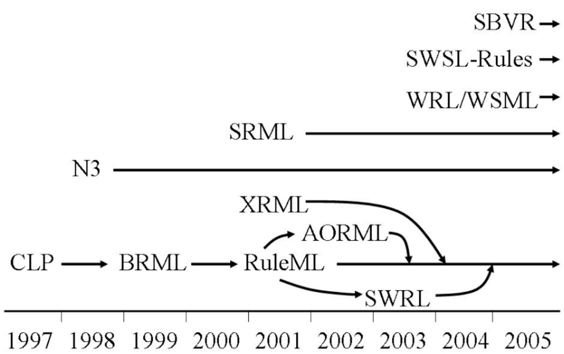

Fig. 1. Development of Rule Languages

sessions, during which consensus was gained about the construct mappings. The analyses and respective consensus development meetings were conducted in turn for each language. First, SRML was analyzed and the analysis was finalized, then SBVR analysis followed.

4.2 Selection of process modeling languages and overlap analysis

The selection and analysis of conceptual process modeling languages was based on existing work. Many BWW-based representational analyses of process modeling languages have already been published, and some of them empirically tested (for an overview please refer to [13]). We specifically focus on conceptual modeling languages rather than executable languages such as Business Process Execution Language (BPEL), since we concentrate on the documentation of policies and processes rather than their execution. Accordingly, we chose Petri Nets (in their original specification), Event-driven Process Chains (EPC), Integrated DEFinition methodology – Process Description Capture Method (IDEF$_3$), and Business Process Modeling Notation (BPMN) as the basis of our comparison. We want to determine if business rule languages can indeed contribute to these popular graphical process modeling languages, as was speculated by Herbst *et al.* [15] for the IS domain.

The analysis of how complementary business rule and process modeling languages are was performed based on the results shown in Table 1. As a first step, we compare the representational capabilities of the six languages and derive a set of BWW representation model constructs that *do not* have a corresponding construct across the chosen languages. This situation implies that full completeness of representation cannot be achieved with the selected languages with respect to the BWW representation model.

As a second step, we apply the process of overlap analysis in order to determine a pair of languages that provides, with respect to the BWW representation model, the highest representation modeling power while having the lowest amount of construct overlap between the languages. Sine the MOC theory takes precedence in such analysis, we focus on all language pairs that provide MOC and select the pair(s) that have the lowest construct overlap.

5. Discussion of Analysis Results

The summary of the BWW representational analysis of SRML and SBVR is shown in Table 1. A tick indicates that the rule modeling language was found to have capability to represent the corresponding BWW representation model construct. The full details of the mapping reasoning are omitted due to paper length limitations.

The analysis shows that the two chosen rule modeling languages are less expressive than their process modeling counterparts, with respect to the BWW representation model. This is perhaps not surprising given that their focus is narrower that that of graphical process modeling languages. For example, while we would expect business rules modeling languages to provide a corresponding representation for a BWW construct such as *state law*, the lack of graphical representation might lessen the need for a representation of *system decomposition* or *system structure*. However, this observation implies that business rule modeling languages like SRML or SBVR in isolation do not provide an equivalent or better means for modeling processes than some of the established and popular process modeling languages (most notably, BPMN).

Closer investigation of Table 1 shows that, even when used in combination, none of the popular process and rule modeling languages considered provides complete coverage for all BWW constructs. In particular, representations for the BWW representation theory constructs of *history, conceivable event space,* and *lawful event space* are missing across all languages under consideration. This implies that even combinations of conceptual process modeling languages and business rule modeling languages are neither able to represent audit trails of activities nor the sets of all the possible or allowed events that can occur in a given situation. More specifically, while the addition of business rule modeling languages appears to alleviate some of the empirically validated weaknesses of the popular BPMN language in particular [3], we still expect the following shortcomings to manifest in practice:

Table 1. BWW Analysis results, including material from [25] and [13]

	Language	**SRML**	**SBVR**	Petri-Net	EPC	$IDEF_3$	BPMN
	Year / Version	**2001**	**2006**	1962	1992	1995	2004 v1.0
	BWW Constructs						
Things	Thing	✓	✓	✓		✓	✓
	Property	✓	✓		✓	✓	✓
	Class		✓	✓			✓
	Kind						✓
States	State	✓		✓	✓	✓	
	Conceivable State Space	✓					
	State Law	✓	✓	✓	✓		
	Lawful State Space	✓		✓			
	Stable State				✓		
	Unstable State			✓			
	History						
Events	Event	✓		✓	✓	✓	✓
	Conceivable Event Space						
	Lawful Event Space						
	External Event				✓		✓
	Internal Event			✓	✓		✓
	Well-defined Event			✓	✓		✓
	Poorly-defined Event						✓
	Transformation			✓	✓	✓	✓
	Lawful Transformation			✓	✓		✓
	Coupling					✓	✓
	Acts on			✓			✓
Systems	System		✓			✓	✓
	System Composition		✓			✓	✓
	System Environment						✓
	System Structure					✓	
	System Decomposition		✓			✓	✓
	Level Structure				✓	✓	✓
	Sub System						✓
# BWW constructs repres.		**10**	**7**	**12**	**11**	**11**	**19**

P1: The lack of corresponding representation for the BWW construct of *history* may have an impact where a log of an entity's state changes is required. The lack of such explicit representation can, for example, impact exception modeling, in particular recovery.

P2: The lack of corresponding representation for the BWW construct of *conceivable event space* implies an inability to model the set of all possible events that can occur within a process. While this representation capability may not always be required in the process modeling domain, the lack of it increases the complexity of identifying events of interest to the process being modeled and events that are not allowed to have a triggering impact.

P3: The lack of corresponding representation for the BWW construct of *lawful event space*, similarly to proposition P2, has a negative impact on the modeling of allowable events in a process model. Specifically, there is no modeling construct that would allow for the representation of all events that are legal in a given process context, thus impacting exception handling modeling.

The overlap analysis of the six chosen languages is summarized in Table 2. Each cell in the table is a quadrant indicating:

1. Unique: The number of BWW constructs represented distinctly by the given combination of languages, free of overlap;
2. Overlap: The number of BWW concepts that can additionally be represented by both languages, with construct overlap;
3. Process Add: The number of non-overlapping BWW constructs contributed by the process modeling language shown in the table column to the rule modeling language shown in the table row; and
4. Rule Add: The number of non-overlapping BWW constructs contributed by the rule modeling language shown in the row to the process modeling language shown in the column.

While the overlap analysis shows synergies between rule modeling languages and process modeling languages, these do not appear to be as dramatic as we expected. It is clear that business process languages are a better choice for modeling organizational procedures, and that BPMN is a construct-rich process modeling language that could be enriched by the addition of SRML for this domain.

Investigation of the overlap results summarized in Table 2 indicates that incorporating the use of SRML with any of the four popular conceptual process modeling languages allows users the ability to represent between fifteen and twenty-three representation theory constructs (i.e. MOC = 23). Minimal ontological overlap is equal to 6, implying that whichever combination of languages is chosen, a minimum of six constructs will be overlapping in the language pair. Considering both MOC and MOO theories, the analysis clearly shows that the combina-

Table 2. Overlap analysis of the selected languages

	Petri Nets		EPC		$IDEF_3$		BPMN	
SRML	8	*5*	9	*5*	9	5	17	*13*
	7	*3*	6	*4*	6	4	6	*4*
SBVR	13	*9*	14	*9*	8	6	14	*13*
	3	*4*	2	*5*	5	2	6	*1*

LEGEND FOR EACH QUADRANT

1. Unique	*3. Process Add*
2. Overlap	*4. Rule Add*

tion of BPMN and SRML provides users with the highest representation power, while having minimal ontological overlap. On the other hand, the analysis also shows that the combination of SRML and Petri Nets is not a good option, given the higher level of overlap and representation of only fifteen representation theory constructs, as compared to twenty-three from the BPMN/SRML combination.

A closer look at SBVR reveals that SRML is a superior option for the integration with conceptual process modeling languages, since it provides representation of three additional representation theory constructs at the same level of construct overlap. The representational capability of SBVR in combination with any of the process modeling languages ranges from thirteen to twenty constructs, while overlap ranges from two to six. MOO indicates in this case that a combination of SBVR and EPC is good from a clarity perspective, although such a combination offers just sixteen representation theory constructs. Table 1 also shows that such a combination would be lacking the representation of the *conceivable* and *lawful state space*, constructs that, intuitively, are important for modeling organizational policies and rules. Accordingly, when a higher representation capability is required, the SBVR and BPMN combination could be chosen, if SRML is not an option.

These initial findings are significant for three reasons. First, they provide guidance to the developers of modeling languages in terms of which areas require improvement. Clearly, there are a number of constructs missing across the board and further investigation is necessary in terms of their criticality and potential addition in future revisions of these modeling languages. Second, our results can provide guidance to organizations in adoption of a specific set of modeling languages for their process documentation efforts. For example, an organization already using BPMN for process modeling has a theoretical basis for the choice of SRML over SBVR. An organization already using Petri Nets would be inclined to adopt SBVR to obtain higher representational power with lower construct overlap, or investigate a switch to BPMN and the associated costs of converting their models. Last, once a pair of languages is in use, organizations may use the results of the analysis (Table 1) as guidance for the development of consistent workaround policies to alleviate the weaknesses their modelers will encounter while using the language pair to model organizational procedures.

6. Conclusions

This paper presents the first theory-based analysis of representational capabilities of two rule modeling languages, *viz*. SRML and SBVR. The consideration of our analyses, together with existing representational analyses of four popular conceptual process modeling languages, has allowed us to provide some initial direction, which combinations of languages provide users with the best representational capabilities. Our findings show that the combination of BPMN with SRML provides users with the highest representation power while suffering an amount of construct overlap that is no higher than that of other language pairs. However, the analysis also shows that even this combination of languages is still deficient in some constructs, *viz. history, conceivable event space,* and *lawful event space*.

While our initial findings encourage further investigation of the integration of process and rule modeling languages, there are some known limitations to our current approach. First, different authors performed the analysis of EPC and IDEF3, and their interpretation of the language constructs may differ from ours. Second, the analysis of Petri Nets is based on an old version of Petri Nets and can be extended to incorporate newer versions (e.g. colored Petri Nets), which may alter the results. Third, the published representational analyses of modeling grammars generally do not include analysis of representational capability of *combinations* of the modeling grammar constructs, focusing instead on representation of each construct in isolation. Fourth, we assume that each BWW representation model construct is equally important for the process modeling domain. In the future, we will conduct an expert study to investigate a more refined ranking of ontological constructs in order to determine the criticality of missing representations. Finally, the results of the mapping need to be tested against a real-world example.

Work is currently underway to include additional rule languages in the evaluation. In particular, SWRL is being considered to complement the analysis of SBVR, and an evaluation of RuleML is in progress.

Other rule modeling languages are also being evaluated for inclusion since more suitable combinations of languages may be identified if more rule languages are included. Further work is also required to develop a cost/benefit calculation that would indicate whether the additional representational capability provided by, for example, SRML over and above BPMN, is worth the complexity of adding an additional language. Last, we see the need for further research that focuses on how to achieve a meaningful and seamless integration of business process and business rule modeling languages.

7. References

[1] Gartner Group, *Creating Enterprise Leverage: The 2007 CIO Agenda*. Egham, UK: Gartner, Inc, 2007.

[2] M. Rosemann, J. Recker, M. Indulska, and P. Green, "A Study of the Evolution of the Representational Capabilities of Process Modeling Grammars," in *Advanced Information Systems Engineering - CAiSE 2006*. vol. 4001, E. Dubois and K. Pohl, Eds. Luxembourg, Grand-Duchy of Luxembourg: Springer, 2006, pp. 447-461.

[3] J. Recker, M. Indulska, M. Rosemann, and P. Green, "How Good is BPMN Really? Insights from Theory and Practice," in *14th European Conference on Information Systems*, Goeteborg, Sweden, 2006.

[4] G. Steinke and C. Nikolette, "Business rules as the basis of an organization's information systems," *Industrial Management + Data Systems,* vol. 103, p. 52, 2003.

[5] G. Wagner, "Rule Modeling and Markup," in *Reasoning Web*, 3564 ed, N. Eisinger and J. Maluszynski, Eds. Msida, Malta: Springer, 2005, pp. 251-274.

[6] P. McBrien and A. H. Seltveit, "Coupling Process Models and Business Rules," in *Information Systems Development for Decentralized Organizations, Proceedings of the IFIP 8.1 WG Conference*: Chapman Hall, 1995.

[7] G. Kappel, S. Rausch-Schott, and W. Retschitzegger, "Coordination in workflow management systems - A rule-based approach," in *Coordination Technology for Collaborative Applications - Organizations, Processes, and Agents*, 1364 ed, W. Conen and G. Neumann, Eds.: Springer, 1998, pp. 99-119.

[8] G. Knolmayer, R. Endl, and M. Pfahrer, "Modeling Processes and Workflows by Business Rules," in *Business Process Management, Models, Techniques, and Empirical Studies*, W. M. P. van der Aalst, J. Desel, and A. Oberweis, Eds. London: Springer, 2000, pp. 16-29.

[9] J. Krogstie, P. McBrien, R. Owens, and A. H. Seltveit, "Information Systems Development Using a Combination of Process and Rule-Based Approaches," in *Third Nordic Conference on Advanced Information Systems Engineering*: LNCS, Springer-Verlag, 1991.

[10] A. Kovacic, "Business renovation: business rules (still) the missing link," *Business Process Management Journal,* vol. 10, p. 158, 2004.

[11] A. Charfi and M. Mezini, "Hybrid web service composition: business processes meet business rules," in *2nd international conference on Service oriented computing*, New York, NY, USA, 2004.

[12] J. Meng, S. Y. W. Su, H. Lam, and A. Helal, "Achieving dynamic inter-organizational workflow management by integrating business processes, events and rules," in *35th Hawaii International Conference on System Sciences*, Hawaii, 2002.

[13] J. Recker, M. Indulska, M. Rosemann, and P. Green, "Do Process Modelling Techniques Get Better? A Comparative Ontological Analysis of BPMN," in *16th Australasian Conference on Information Systems*, Sydney, Australia, 2005.

[14] P. Green and M. Rosemann, "Perceived Ontological Weaknesses of Process Modelling Techniques: Further Evidence," in *10th European Conference on Information Systems*, Gdansk, 2002, pp. 312-321.

[15] H. Herbst, G. Knolmayer, T. Myrach, and M. Schlesinger, "The Specificatioin of Business Rules: A Comparison of Selected Methodologies," in *Methods and Associated Tools for the Information System Life Cycle*, A. A. Verijn-Stuart and T.-W. Olle, Eds. Amsterdam: Elsevier, 1994, pp. 29-46.

[16] R. Lu and S. Sadiq, "A Survey of Comparative Business Process Modeling Approaches," in *10th International Conference on Business Information Systems* Poznan, Poland: Lecture Notes in Computer Science, Springer-Verlag, 2007.

[17] R. Weber, "Ontological Foundations of Information Systems," *Coopers & Lybrand,* 1997.

[18] M. Rosemann, P. F. Green, and M. Indulska, "A Reference Methodology for Conducting Ontological Analyses," in *23rd International Conference on Conceptual Modeling*, Shanghai, 2004, pp. 110-121.

[19] B. Wyssusek, "Ontology and ontologies in information systems analysis and design: a critique," in *Americas Conference on Information Systems* New York, 2004.

[20] R. M. Chisholm, "The basic ontological categories," in *Language, Truth, and Ontology*. vol. 51, K. Mulligan, Ed. Dordrecht: Kluwer Academic Publishers, 1992, pp. 1-13.

[21] M. Uschold, M. King, S. Moralee, and Y. Zorgios, "The Enterprise Ontology," Artificial Intelligence Applications Institute, The University of Edinburgh 1997.

[22] P. Green, M. Rosemann, M. Indulska, and C. Manning, "Candidate Interoperability Standards: An Ontological Overlap Analysis," *Data & Knowledge Engineering,* vol. 62 pp. 274-291, 2007.

[23] Coverpages.com, "Simple Rule Markup Language (SRML)," R. Cover, Ed., 2001.

[24] OMG, "Semantics of Business Vocabulary and Business Rules Specification, Interim Specification," 2006.

[25] P. F. Green, M. Rosemann, and M. Indulska, "Ontological evaluation of enterprise systems interoperability using ebXML," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 17, pp. 713-725, 2005.