On the Expressiveness of Business Process Modeling Notations for Software Requirements Elicitation

Carlos Monsalve[1,2]
[1]Facultad de Ingeniería en Electricidad y Computación
Escuela Superior Politécnica del Litoral (ESPOL)
Apartado 09-01-5863. Guayaquil, Ecuador
email: monsalve@espol.edu.ec

Alain April[2], Phd, Alain Abran, Phd[2]
[2]Department of Software Engineering
École de Technologie Supérieure
1100 rue Notre-Dame Ouest. Montréal, Canada
email: {alain.april, alain.abran}@etsmtl.ca

Abstract—**Business process models have proved to be useful for requirements elicitation. Since software development depends on the quality of the requirements specifications, generating high-quality business process models is therefore critical. A key factor for achieving this is the expressiveness in terms of completeness and clarity of the modeling notation for the domain being modeled. The Bunge-Wand-Weber (BWW) representation model is frequently used for assessing the expressiveness of business process modeling notations. This article presents some propositions to adapt the BWW representation model to allow its application to the software requirements elicitation domain. These propositions are based on the analysis of the Guide to the Software Engineering Body of Knowledge (SWEBOK) and the Guide to the Business Analysis Body of Knowledge (BABOK). The propositions are validated next by experts in business process modeling and software requirements elicitation. The results show that the BWW representation model requires to be specialized by including concepts specific to software requirements elicitation.**

Business process model; software requirements; survey; BWW representation model; empirical research; BPMN; expressiveness

I. INTRODUCTION

A software development project is highly dependent on the quality of the software requirements process, including elicitation, analysis, specification, and validation [1,2]. A high-quality software requirements elicitation task depends, among other things, on the active participation of all the stakeholders [1,2]. Conceptual modeling is useful during software requirements elicitation as it helps to understand the subject matter it represents [1,2]: one of the most popular conceptual modeling representation approaches is Business Process Modeling (BPM) [3] which is the activity of representing business processes of an enterprise, so that they may be analyzed. To ensure the effectiveness of BPM, it is important that the appropriate BPM notation be selected. A good BPM notation needs to be complete and clear, that is, capable of expressing all the relevant concepts in the domain under study [1]. How can we assess if our BPM notation is complete and clear?

One analysis technique proposed for assessing the expressiveness in terms of completeness and clarity of a BPM notation [4,5] is the BWW representational analysis [6]. This analysis is based on an ontological model (i.e. the BWW representation model), and it assesses the expressiveness of a notation in terms of its capability to represent the various concepts of the representation model. BWW representational analysis has been used to assess a number of BPM notations [4].

To improve the relevance of this assessment some authors have recommended to select the ontology, or a subset of it, that best fits the specific domain that encompasses the modeling process [4,7]. No previous research was found in which the fitness of the BWW representation model for the specific domain of software requirements elicitation was studied previously. This paper presents our research work to analyze the BWW representation model within the context of the software requirements elicitation domain.

Software engineers and business analysts are both professionals trained to perform software requirements elicitation tasks. Each of these two professions can rely on a guide to its body of knowledge and obtain certification:

1) the Guide to the Software Engineering Body of Knowledge (SWEBOK) [2], and

2) the Guide to the Business Analysis Body of Knowledge (BABOK) [8].

One of the "knowledge areas" (KA) of the SWEBOK is the Software Requirements KA, and requirements elicitation is one of the key topics within this KA. Software engineers, to conduct software and system requirements elicitation in a software project often use this knowledge. On the other hand, certified business analysts often use the BABOK guidance to conduct early requirements elicitation also known as business requirements. The SWEBOK and the BABOK are used here as the key references for analyzing the BWW representation model within the specific context of requirements elicitation as a whole.

Studies based on the SWEBOK and the BABOK recommendations were used, in this research, to derive a set of propositions aiming at adjusting the BWW representation model to better fit the software requirements elicitation domain. These propositions were also empirically tested through a survey administered to a group of experts within the framework of the 2nd International Symposium in Software Engineering Management held in the summer of 2011 in Montreal, Canada.

Finally, to show the usefulness of these research findings, this paper contributes to the assessment of a popular BPM notation, the Business Process Modeling Notation (BPMN) [9] by applying the propositions supported by the survey results in the analysis of previous assessments of the expressiveness of BPMN performed by Recker, Indulska, Rosemann and Green [10].

This paper is structured as follows. Section II presents an overview of the BWW representational analysis and its representation model. Section III presents the analysis of the BWW representation model using the SWEBOK and the BABOK as key references. Section IV presents the design and the findings of the survey conducted to validate the research results. Section V discusses the usefulness of the findings using an example of use on the BPMN notation. Finally, section VI concludes with a review of the contributions of this research, its limitations, and future work planned.

II. Overview of the BWW Representation Model and Representational Analysis

Wand and Weber have proposed a set of ontological models [6,11] to represent the basic concepts that allows the description of an information system, its structure, and its behavior. They point out that, from these ontological models, it is the representation model that is useful for assessing the expressiveness of any modeling notation [12]. The definitions of the various concepts of the representation model can be found in [13], *thing* being the elementary concept: according to the representation model "the world is made up of things" that "might be of interest to users of information systems".

Wand and Weber argue that a modeling notation is complete only if it is possible to map each of the concepts of the representation model to the modeling notation constructs [12] (see (1) and (4) in Fig. 1). They also argue that a modeling notation is clear only if each of its constructs has a one-to-one mapping with the concepts of the representation model. Any deviation from this one-to-one mapping constitutes an ontological deficiency of the modeling notation, affecting its clarity.

Wand and Weber present three types of ontological deficiencies: construct overload (see (2) in Fig. 1), construct redundancy (see (3) in Fig. 1), and construct excess (see (5) in Fig. 1). A complete description of these ontological deficiencies can be found in [6,12].

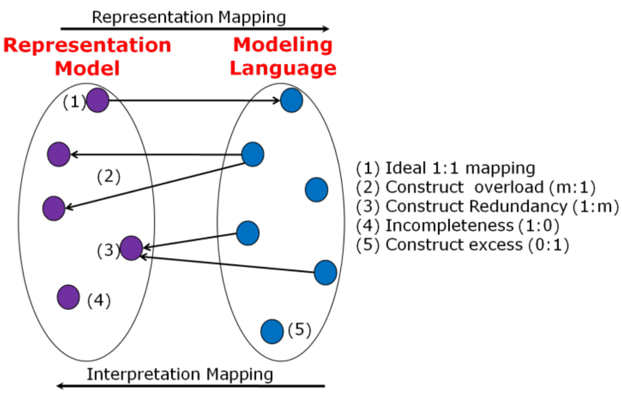


Fig. 1. Representational Analysis (adapted from Wand and Weber [16])

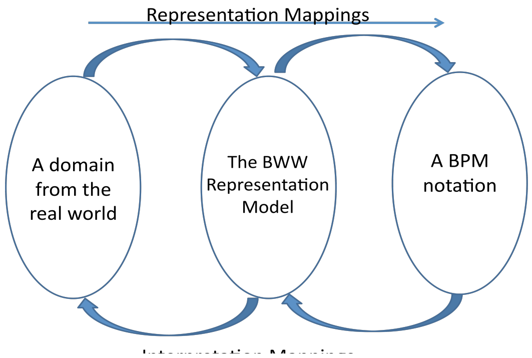
In addition, Wand and Weber proposed two types of mappings between the representation model and the notation being assessed: (1) the representation mapping from the representation model to the modeling notation, and (2) the interpretation mapping from the modeling notation to the representation model [12] (see top and bottom of Fig. 1). The representation mapping requires mapping each ontological concept to its corresponding modeling notation constructs, and provides useful information for identifying the degree of completeness of the modeling notation assessed, as well as potential redundancy deficiencies of the modeling notation. The interpretation mapping requires mapping each modeling notation construct to its corresponding ontological concepts, and provides useful information for identifying the potential degree of overload of a modeling notation, as well as its constructs in excess.

The ontological assessment of a modeling notation based on the representation model is called the BWW representational analysis [14], and the underlying ontological model is called the BWW representation model. In today's literature, it is possible to find proposals of other similar ontological assessment techniques, but most of these have not achieved the popularity of representational analysis for the assessment of BPM notations [4,5]. Moreover, the BWW representational analysis has reached a "high maturity" level, and its ontological model is considered to be "very well understood"[5].

III. The BWW Representation Model Within the Software Requirements Elicitation Domain

Wand and Weber point out that several transformations might be required to go from the real world to its final representation (e.g. from the real world to a conceptual model, and from a conceptual model to a programming language), and that each of these transformations involves a specific mapping [6,12]. The research work reported here went from a very specific domain of the real world (i.e. software requirements elicitation) to a BPM notation, having the BWW representation model as an intermediate step (see Fig. 2). Therefore, two sets of mappings were required: (1) the mappings between the specific domain and the BWW representation model, and (2) the mappings between the BWW representation model and the BPM notation. This section of the paper presents the results of the first set of mappings.

Mapping the software requirements elicitation domain knowledge to the BWW representation model was done in the following way. The SWEBOK and the BABOK were selected as the key references containing generally accepted knowledge of the software requirements elicitation domain. Three researchers were involved in the mapping process. Two research groups of two researchers were formed. One of the researchers participated in both groups. This allowed the research team to have continuity in the research work, and at the same time reduced subjectivity in the elaboration of the mappings, and the interpretation of the results. The first research group worked on the SWEBOK-BWW mappings, and the second one worked on the BABOK-BWW mappings. Each research group was asked to review its key reference (i.e. the



SWEBOK or the BABOK) to identify the relevant concepts associated with requirements elicitation. Each group took the precaution of reviewing carefully its reference to ensure that the list of relevant concepts would be comprehensive. Each group required several review iterations to reach a consensus on its final candidate list of relevant concepts. Then, each group was asked to classify each item on its list of concepts into one of two sets: (1) those that should be represented in a business process model; and (2) those that should be supported by a modeling tool. Again, several iterations were required to reach a consensus on a common classification within each group. From the two resulting sets of concepts (i.e. those attributed to a model and those attributed to a modeling tool), the set of concepts that should be represented in a business process model was taken as the reference for the next step. The final step consisted on mapping the BWW representation model with the set of concepts each group selected. Once again, several iterations were needed to reach a consensus within each group. Table I shows the combined results of the mappings produced by both groups. Only those BWW concepts that have a mapping with both the SWEBOK and the BABOK are presented.

Some relevant concepts selected from the SWEBOK or the BABOK do not appear in Table I as they could not be mapped to the BWW representation model. One possible explanation for this could be that the BWW representation model is incomplete and cannot fully describe the knowledge of the software requirements elicitation domain. Among these concepts, goals and objectives were the only concepts to belong to both SWEBOK and BABOK. This lead to the formulation of the first proposition:

P1: Practitioners require representing goals and objectives in a business process model intended to be used for software requirements elicitation.

In terms of overloading, it can be observed from Table I that the most overloaded BWW concepts are: *thing*, *state law*, *transformation*, *acts on*, and *coupling*. Regarding the overload of *thing*, initial findings of empirical research [15] suggest that there is a need in having a specialization for *things*. This leads to the second proposition:

P2: Practitioners require specialized modeling constructs to represent each of the following concepts: roles (i.e. internal users), external stakeholders (e.g. customers, providers), devices, objects, business units (i.e. departments), software interfaces and software components.

Concerning the overload of *state law*, recent studies have shown that most of the popular BPM notations lack a construct that represents a state law [4,16]. This has been observed, and confirmed through empirical research, as a potential difficulty for modelers attempting to represent business rules in BPM [4,10,17]. As a consequence, several researchers are currently working on the representation of business rules in BPM [18-20], and therefore we do not cover here this deficiency.

Regarding the overload of *transformation*, this coincides with the observed results already published by previous studies that evaluated various BPM notations using the BWW representation model [4,16]. These previous results have suggested a potential need to specialize the BWW concept *transformation*. This leads to the third proposition:

P3: Practitioners require specialized modeling constructs to represent each of the following concepts: actions (i.e. tasks), activities, sub-processes, and processes.

Finally, the overload of *acts on* and *coupling* is generated by the explicitness of the SWEBOK and the BABOK in defining various types of interactions, dependencies, or relationships between users, stakeholders, roles, entities, software components, and the environment of the system represented. The stakeholder interpreting a business process model might be confused by the various meanings adopted by the BPM notation to represent these two BWW concepts (i.e. *acts on* and *coupling*). The BWW concepts *acts on* and *coupling* also contribute to a redundancy in the BWW concepts. Any type of interaction or dependency (e.g. interaction between users) might be described either by the *coupling* concept or by the *acts on* concept. This choice hinders the modeling of a business process because, for instance, a modeler has to decide whether to choose a modeling construct that represents the *coupling* concept or a modeling construct that represents the *acts on* concept to describe an interaction

TABLE I MAPPINGS BETWEEN THE BWW REPRESENTATION MODEL, SELECTED SWEBOK CONCEPTS AND SELECTED BABOK CONCEPTS

SWEBOK Concepts	BWW Concepts	BABOK Concepts
Users, roles, third party, devices, software components, entities from the problem domain, interfaces with the environment	Thing	Stakeholders, users, roles, classes, objects, business units, departments
Conceptual modeling, state models, object models	Conceivable state space	Behavioral characteristics of a solution (features and functions)
How are tasks done, what the software product is not expected to do, functional requirements, non-functional requirements, emergent properties, industry practices, product parameters	State law	Rules, business constraints, responsibilities, constraints
Conceptual modeling, event traces, usage scenarios	Conceivable event space	Interactions people-solution-system, IT responses

between users. This leads to the last proposition:

P4: Practitioners require specialized modeling constructs to represent each of the following types of interactions: interactions between roles, interactions between roles and external stakeholders, interactions between software components, interactions between roles and software components, interactions between external stakeholders and software components, interactions between objects, interactions between business units.

IV. SURVEY: DESIGN AND RESULTS

The propositions from Section III are then validated. After analyzing possible validation methods, one approach, with its corresponding participant's profile, was proposed by the research team: use a controlled survey with a number of experienced practitioners that use BPM in software requirements elicitation.

To ensure the validity of the results, the survey was designed and conducted using the principles recommended by Kitchenham and Pfleeger [21] as well as those recommended by Salant and Dillman [22]. Based on those recommendations a validation protocol was elaborated and is described next.

The objective of the survey was to test all four propositions. The practitioners selected could be software engineers, business analysts, or professionals from related backgrounds.

The survey was administered to the participants following a semi-supervised format: that is, it was conducted as a workshop during an international software engineering symposium. A member of the research team introduced the motivation and objectives of the survey, and was available to answer any question from the participants. The participants were volunteers and had the opportunity to quit the survey at any time.

Because the target audience is very specialized, the population was difficult to determine. Therefore, a non-probabilistic sample (i.e. purposive sampling) was chosen for the survey [21,22]. Nineteen participants were present at the beginning of the survey workshop; seventeen of them completed the survey and returned their answers. Similar previous studies have been conducted by other authors [10,17,23-25] using groups of 4 to 21 participants.

The questionnaire was designed based on the structure proposed in [24] to help determine the "level of severity" [4] of each of the problems addressed in the propositions. This structure has been used and validated by several previous studies conducted by other authors [10,24,25]. The questionnaire was pre-tested three times with the help of experienced IT professionals having more than 5 years of experience in software development. The pre-tests were planned for both: (1) improving the quality of the questionnaire, and (2) ensuring an appropriate duration for answering the questions. All the questions were close-ended questions to limit the respondents' answers to the survey and to facilitate the coding of the answers.

Three major variables, that define the demographics of the set of participants, were collected: (1) their profession or job function, (2) their number of years of experience using BPM, and (3) their number of years of experience doing software requirements elicitation.

Over half of the participants (53%) had more than 2 years of experience in BPM, and close to 50% of the participants had more than 6 years of experience in software requirements elicitation. Regarding the profession or job function, 6 out of 17 had been classified under "other" whereas:

- three indicated to having several professions over the years, even though the question explicitly solicited to choose the answer that best describes their profession or job function.
- one was a Ph.D. researcher.
- the other two indicated that they were measurement consultant and IT process improvement specialist respectively.

65% of the participants agreed that goals and objectives are necessary to be represented in a business process model. Moreover, 70% of the participants who agreed on that need indicated that not being able to directly represent goals or objectives in a business process model constitutes a problem. These findings support **proposition P1**.

Over 64% of the participants have expressed the need of a BPM construct specifically designed to represent the following concepts: external stakeholders and business units. This need was expressed by 47% of the participants for the following concepts: internal roles, software components and software interfaces. These findings partially support **proposition P2**.

Over 64% of the participants expressed the need of a BPM construct specifically designed to represent the following concepts: tasks, activities, sub-processes, and processes. This finding supports **proposition P3**.

Over 52% of the participants expressed the need of a BPM construct that has been specifically designed to represent the following concepts: interaction between roles, and interaction between roles and external stakeholders. This need was expressed by 41% of the participants for the following concepts: interactions between software components, interactions between roles and software components, and interactions between business units. These findings partially support **proposition P4**.

Finally, from the four propositions originally formulated to guide this empirical research, two have been supported by the results of the survey, and another two have been partially supported. Table II summarizes these survey results.

V. APPLYING THE FINDINGS TO PREVIOUS BPMN REPRESENTATIONAL ANALYSIS

For purposes of showing the usefulness of the findings reported in sections III and IV, some results of previous BPMN representational analyses performed by Recker, Indulska, Rosemann and Green [10] are analyzed.

Recker, Indulska, Rosemann and Green already published [10] that BPMN does not support the BWW concept of *system structure* and, based on this construct deficit, they derived a proposition indicating that BPMN users often face difficulties in representing inter-organizational business processes and structuring business process models "into constituent models". However, this observation had "insignificant support" from the nineteen BPMN users they interviewed to test this assumption. This is supported by our results provided in Table I where *system structure* is not part of the list of BWW concepts from the mappings performed using the SWEBOK and the BABOK as key references. Actually, system structure was not mapped to the SWEBOK concepts or the BABOK concepts. Based on Table I the assessment of a BPM notation should consider not including the BWW concept *system structure*.

Concerning the BWW concept *thing*, Recker, Indulska, Rosemann and Green found that BPMN can represent a *thing* using two different modeling constructs: therefore, BPMN presents a construct redundancy. They derived a proposition indicating that BPMN users will have problems when choosing between the two BPMN modeling constructs to represent for instance 'organizational departments'. However, this had only a "limited support" from the BPMN users they interviewed. The perception of their interviewees converges with our results of proposition P2, which is supported in terms of the need of a modeling construct that has been specifically designed to represent business units.

Finally, a similar result is reported by Recker, Indulska, Rosemann and Green regarding the BWW concept *transformation*. Based on the redundancy of BPMN for this concept, they derived a proposition indicating that BPMN users "will get confused as to which construct is to be used when representing a transformation." This proposition had "insignificant support" from the BPMN users they interviewed. The perception of their interviewees converges with our results of proposition P3 that establishes the need to specialize the BWW concept *transformation* to allow the representation of tasks, activities, sub-processes and processes.

VI. CONTRIBUTIONS, FUTURE WORK, AND THREATS

Representing goals and objectives in a business process model has been identified as a need if a business process model is intended to be used for software requirements elicitation. Not being able to represent goals and objectives is perceived as a problem.

The specialization of the BWW concept *thing* is perceived as necessary to differentiate between two concepts that are relevant to software requirements elicitation: external stakeholders and business units. Further empirical research should be conducted in order to study practitioners' perceptions regarding other concepts: internal roles, software components and software interfaces. In addition, the results of the survey show that such a specialization is not relevant for the concepts: objects and devices.

TABLE II SUMMARY OF SURVEY RESULTS

	Propositions	Observations
Supported	P1, P3	
Partially Supported	P2, P4	Not all the types of *things* require a specifically designed modeling construct. Only some types of interactions require a specifically designed modeling construct.

The specialization of the BWW concept *transformation* is perceived as necessary to differentiate between various concepts that are relevant to software requirements elicitation: processes, sub-processes, activities and tasks. Such specializations reduce the number of redundancies that previous representational analyses have found when assessing BPM notations [25].

Some importance is reflected in the survey results regarding the ability to represent, with a BPM construct specifically designed for that purpose, the interactions between internal roles and the interactions between internal roles and external stakeholders. Further empirical research should be conducted in order to study practitioners' perceptions regarding other types of interactions. In addition, the survey results show that it is not relevant to represent in a distinctive way the interactions between external stakeholders and software components or the interactions between objects.

Taking these findings into account when performing a representational analysis of a BPM notation can increase the relevance of its assessment results. For instance, the number of actual redundancy deficiencies found in a BPM notation can turn to be more accurate as a consequence of accepting the specialization of the BWW concepts *thing* and *transformation*.

A discussion on the validity threats of this research follows. To increase construct validity, the version of the BWW representation model used for all the mappings was always the same. The propositions were formulated based on the results of the mappings and on the interpretation of the BWW concepts based on the descriptions of the given domain found in the SWEBOK and in the BABOK. In addition, the questionnaire design followed well accepted guidelines found in the literature, and used a structure that has been already validated in similar studies. The formulation of the questions was based on the propositions to be tested. Moreover, the questionnaire was pre-tested and discussed three times with professionals that fit the target-audience profile.

Regarding internal validity, pairs of researchers required several iterations to reach a consensus on each final mapping. In addition, a great effort was made to ensure survey participants' experience both in BPM and software requirements elicitation. The data collected in the survey allow analyzing in future work the impact of the participants' demographics on the variations of their answers. In addition, the fact that the survey was semi-supervised allowed to clear doubts from the participants, obtaining more accurate responses. Moreover, the use of the questionnaire structure helped to confirm the perception of a modeling problem.

Regarding external validity, the main threat is the sample size used in this research. However, the results are strengthened in terms of generalization by the fact that the propositions being tested were derived from previous theoretical work: that is, for those propositions that have been supported, the results of the survey converge with the results of our analyses of the BWW representation model. In other words, the final results are concluded based on multiple sources of evidences (i.e. triangulation). Finally, to increase the reliability the recommendations proposed in [14] and [5] have been taken into consideration in the mapping processes. Regarding the survey, a protocol was elaborated, the questionnaire was retested with one of the professionals who volunteered for the pre-tests, and closed questions were preferred to reduce the bias of the researcher when coding the responses.

REFERENCES

[1] Y. Wand and R. Weber, "Research commentary: information systems and conceptual modeling -- a research agenda," Info. Sys. Research, vol. 13, pp. 363-376, 2002.

[2] A. Abran, J. Moore, P. Bourque, and R. Dupuis, SWEBOK: Guide to the Software Engineering Body of Knowledge 2004 Version. Los Alamitos, California: IEEE Computer Society, 2004.

[3] I. Davies, P. Green, M. Rosemann, M. Indulska, and S. Gallo, "How do practitioners use conceptual modeling in practice?," Data & Knowledge Engineering, vol. 58, pp. 358-380, 2006.

[4] M. Rosemann, P. Green, M. Indulska, and J. C. Recker, "Using ontology for the representational analysis of process modelling techniques," International Journal of Business Process Integration and Management, vol. 4, pp. 251-265, 2009.

[5] A. Gehlert and W. Esswein, "Toward a formal research framework for ontological analyses," Adv. Eng. Inform., vol. 21, pp. 119-131, 2007.

[6] Y. Wand and R. Weber, "On the deep structure of information systems," Information Systems Journal, vol. 5, pp. 203-223, 1995.

[7] M. Rosemann and P. Green, "Integrating multi-perspective views into ontological analysis," in Proceedings of the 21st international conference on Information systems, ICIS'00. Brisbane, Queensland, Australia: Association for Information Systems, 2000, pp. 618-627.

[8] IIBA, A Guide to the Business Analysis Body of Knowledge (BABOK Guide), Version 2.0 ed. Toronto: International Institute of Business Analysis (IIBA), 2009.

[9] OMG, "OMG Business Process Model and Notation (BPMN), version 1.2," Object Management Group, 2009.

[10] 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: ECIS' 06, A. M. e. Ljungberg J, Ed. Göteborg, Sweden, 2006, pp. 1-12.

[11] Y. Wand and R. Weber, "An ontological model of an information system," IEEE Trans. Softw. Eng., vol. 16, pp. 1282-1292, 1990.

[12] Y. Wand and R. Weber, "On the ontological expressiveness of information systems analysis and design grammars," Information Systems Journal, vol. 3, pp. 217-237, October 1993.

[13] P. Green and M. Rosemann, "Integrated process modeling: an ontological evaluation," Information Systems, vol. 25, pp. 73-87, 2000.

[14] P. Green, M. Rosemann, M. Indulska, and J. Recker, "Improving representational analysis: an example from the enterprise systems interoperability domain," in 17th Australasian Conference on Information Systems. Adelaide, Australia, 2006.

[15] C. Monsalve, A. April, and A. Abran, "Requirements elicitation using BPM notations: focusing on the strategic level representation," in Recent Researches in Applied Computer and Applied Computational Science: 10th WSEAS International Conference on Applied Computer and Applied Computational Science (ACACOS '11). Venice, Italy: WSEAS Press, 2011, pp. 235-241.

[16] J. Recker, M. Rosemann, M. Indulska, and P. Green, "Business process modeling: a comparative analysis," Journal of the Association for Information Systems, vol. 10, pp. 333-363, 2009.

[17] P. Green, M. Rosemann, and A. Ipswich, "Ontological analysis of integrated process models: testing hypotheses," AJIS, vol. 9, 2001.

[18] R. Meersman, Z. Tari, P. Herrero, S. Goedertier, and J. Vanthienen, "Declarative Process Modeling with Business Vocabulary and Business Rules," in On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops. vol. 4805: Springer Berlin / Heidelberg, 2007, pp. 603-612.

[19] M. zur Muehlen and M. Indulska, "Modeling languages for business processes and business rules: A representational analysis," Information Systems, vol. 35, pp. 379-390, June 2010.

[20] M. Milanovic, D. Gasevic, and L. Rocha, "Modeling Flexible Business Processes with Business Rule Patterns," in The 15th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2011), Helsinki, Finland, 2011, pp. 65-74.

[21] B. A. Kitchenham and S. L. Pfleeger, "Personal opinion surveys," in Guide to Advanced Empirical Software Engineering, F. Shull, J. Singer, and D. I. K. Sjoberg, Eds.: Springer London, 2008, pp. 63-92.

[22] P. Salant and D. A. Dillman, How to conduct your own survey. New York: John Wiley & Sons Inc, 1994.

[23] P. F. Green and M. Rosemann, "Perceived ontological weaknesses of process modeling techniques: further evidence," in 10th European Conference on Information Systems: Information Systems and the Future of the Digital Economy, ECIS 2002, S. Wrycza, Ed. Gdansk, Poland: University of Gadalsk, 2002, pp. 312-320.

[24] I. G. Davies, M. Rosemann, and P. Green, "Exploring proposed ontological issues of ARIS with different categories of modellers," in Australasian Conference on Information Systems, (ACIS 2004). Hobart, Australia: Australasian Chapter of the Association for Information Systems 2004.

[25] 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: Australasian Chapter of the Association for Information Systems 2005.