2008

Representational Deficiency of Process Modelling Languages: Measures and Implications

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

Jan Recker
Information Systems Discipline Queensland University of Technology, j.recker@qut.edu.au

Michael
Queensland University of Technology, m.rosemann@qut.edu.au

Peter Green
The University of Queensland, p.green@business.uq.edu.au

REPRESENTATIONAL DEFICIENCY OF PROCESS MODELLING LANGUAGES: MEASURES AND IMPLICATIONS

Indulska, Marta, The University of Queensland, St Lucia, QLD 4072, Australia, m.indulska@business.uq.edu.au

Recker, Jan, Queensland University of Technology, 126 Margaret Street, Brisbane, QLD 4000, Australia, j.recker@qut.edu.au

Rosemann, Michael, Queensland University of Technology, 126 Margaret Street, Brisbane, QLD 4000, Australia, m.rosemann@qut.edu.au

Green, Peter, The University of Queensland, St Lucia, QLD 4072, Australia, p.green@business.uq.edu.au

Abstract

The large number of available process modelling languages has given rise to the need for evaluation and comparison of their representational capabilities. Over the last few years, the research community has risen to the challenge by carrying out a significant amount of work in the area of such analysis. Much of this effort is based on the Bunge-Wand-Weber representation model, a common benchmark used for the analysis of grammars that purport to model the real world and the interactions within it. However, the carried out BWW analyses of various process modelling languages exist largely separately from each other, with no comprehensive effort at the comparative measurement of their representational capability. This paper introduces four measures that, together, reflect the representational capacity and clarity of a process modelling language. These four measures are used in this paper to compare seven popular process modelling languages. The work provides insights into the representational deficiency similarities and differences between process modelling languages and also predicts some of their implications for practice.

Keywords: process modelling, representation theory, ontological analysis, business process management

1 INTRODUCTION

While the general objectives and methods of Business Process Management (BPM) are not new, BPM has only recently received a significant amount of attention and is now perceived to be a main business priority (Gartner Group 2007). However, the actual modelling of business processes still presents major challenges for organizations (Indulska *et al.* 2006). In simple terms, process modelling is the task of creating so-called process models through the use of a (mostly graphical) modelling language, i.e., a set of grammar constructs and rules to combine those constructs. As graphical presentations of current or future business processes, business process models serve two main purposes. First, intuitive business process models are used for scoping the project, and capturing and discussing business requirements and process improvement initiatives with subject matter experts. A prominent example of such a business modelling language is the Event-driven Process Chain (EPC) (Scheer 2000). Second, business process models are used for process automation, which requires their conversion into executable languages. These automated languages have higher requirements in terms of expressive power. Examples include Petri nets (Petri 1962) or the Business Process Modelling Notation (BPMN) (BPMI.org & OMG 2006).

Overall, a large number of process modelling languages has been proposed since Carl Petri published his initial ideas on Petri nets in 1962 (Petri 1962), and process modelling has become one of the most popular reasons for conceptual modelling (Davies *et al.* 2006). In turn, this situation has created problems for organizations in terms of choice of language for their process modelling initiatives. The lack of systematic evaluation and comparison of representational capabilities of modelling techniques leaves organizations less capable of choosing a technique that fits their modelling needs, hence increasing the complexity of documenting and managing their processes. The research community has responded to this problem by investigating theoretical bases on which such languages can be systematically compared, and has also provided analyses of various process modelling languages. The basis for the majority of the analyses has been the Bunge-Wand-Weber (BWW) representation model (Wand & Weber 1990, 1993, 1995, Weber 1997), which uses the principles of *representational analysis* for the investigation of a modelling language's strength and weaknesses. The BWW representation model denotes a widespread means for evaluating conceptual modelling languages for information systems analysis and design. However, despite the availability of representational analyses of various languages, there is a lack of consistent measurement through which the languages' representational capabilities might be compared. Also, most studies to date have focussed on one specific representational measurement only, *viz.*, ontological completeness, instead of exploiting the full spectrum of possible deficiency measurement that the representation model has to offer (Rosemann *et al.* forthcoming). Thus, our research is motivated in several ways:

- to provide, for the BWW analyses, measurements that reflect a process modelling language's capability to represent a domain completely and without ambiguity;
- to visualise and compare, based on the introduced measures, the representational deficiencies of seven popular process modelling languages; and
- to examine any evident patterns of similarity between the various languages, based on the introduced measures, and discuss their implications for process modelling practice.

The *aim of this paper* then is to enhance existing BWW analyses of completeness and clarity of process modelling languages by providing finer granularity of comparison through the use of standardised measures. We are very much aware that ontological completeness and clarity are not the only relevant criteria for the evaluation of capabilities of a process modelling language. Thus, the focus on the set of BWW constructs in our comparative analysis leads to a specific scope. The study is based on a review of seven previously published BWW analyses of process modelling languages, *viz.* Petri nets (Petri 1962), EPC (Scheer 2000), Electronic Business using eXtensible Markup Language Business Process Specification Schema (ebXML BPSS) v1.01 (OASIS 2001), Business Process

Modelling Language v1.0 (BPML) (Arkin 2002), Business Process Execution Language for Web Services (BPEL4WS) v1.1 (Andrews *et al.* 2003), Web Service Choreography Interface v1.0 (WSCI) (Arkin *et al.* 2002), and BPMN (BPMI.org & OMG 2006), and introduces four measures that reflect these languages' complexity through the degrees of construct deficit, overload, excess and redundancy. The comparability afforded by the introduced measures allows the identification of commonalities between the languages and also points to differences in their design considerations.

This paper is structured as follows. The next section provides an overview of the benchmark on which the measures are to be defined – the Bunge-Wand-Weber representation model – and its previous applications in the area of process modelling. Section 3 reviews the BWW analyses of the seven chosen process modelling languages and introduces the four measures on which they can be compared. Section 4 discusses the results of the application of the measures to our chosen set of process modelling languages, and also discusses some implications for practice. The paper concludes in section 5 with a brief discussion of results, limitations, and an outlook to future research.

2 BACKGROUND & RELATED WORK

Over the last few decades many conceptual modelling languages have emerged with limited theoretical foundation underlying their conception or development (Floyd 1986). Concerned over this trend, Wand and Weber (1990, 1993, 1995) developed and refined a set of models for the evaluation of the representational capability of the modelling languages. These models are based on an ontology defined by Bunge (1977) and are referred to as the Bunge-Wand-Weber (BWW) models. Generally, *ontology* studies the nature of the world and attempts to organize and describe what exists in reality, in terms of the properties of, the structure of, and the interactions between real-world things (Shanks & Tansley & Weber 2003). As computerized information systems are representations of real world systems, Wand and Weber suggest that a theory of representation based on ontology can be used to help define and build information systems that contain the necessary representations of real world constructs including their properties and interactions. The *BWW representation model* is one of three theoretical models defined by Wand and Weber (Wand & Weber 1995) that make up the Representation Theory. The model specifies the constructs that are deemed necessary to provide faithful representations of Information Systems, and which therefore should be included in any conceptual modelling language. These constructs can be represented in a meta model that shows four clusters (Rosemann & Green 2002): things including properties and types of things; states assumed by things; events and transformations occurring on things; and systems structured around things. For more information on the BWW representation model refer to (Weber 1997).

Weber (1997) suggests that the BWW representation model can be used to analyze a particular modelling language to make predictions on the modelling strengths and weaknesses of the language, in particular its capabilities to provide *complete* and *clear* representations of the domain being modelled. Such analysis is done by relating, or mapping, the representational model and the language constructs to each other. The basic assumption is that any deviation from a 1:1 relationship between the corresponding constructs in the representation model and the modelling language leads to representational deficiency in the use of the language, which potentially causes confusion to its users. He clarifies two main evaluation criteria that may be studied according to the BWW model: *Ontological Completeness* and *Ontological Clarity*. These criteria can be further categorized into four sub-types (see Figure 1).

Ontological completeness is indicated by the inverse degree of construct deficit (1:0), *i.e.*, the extent to which a process modelling language covers completely the constructs proposed in the BWW representation model. On the other hand, *ontological clarity* is constituted by the degrees of construct overload (m:1), being the extent to which single language constructs cover several BWW constructs, construct redundancy (1:m), being the extent to which a single BWW construct maps to several language constructs, and construct excess (0:1), being the extent of language constructs that do not map to any BWW construct.

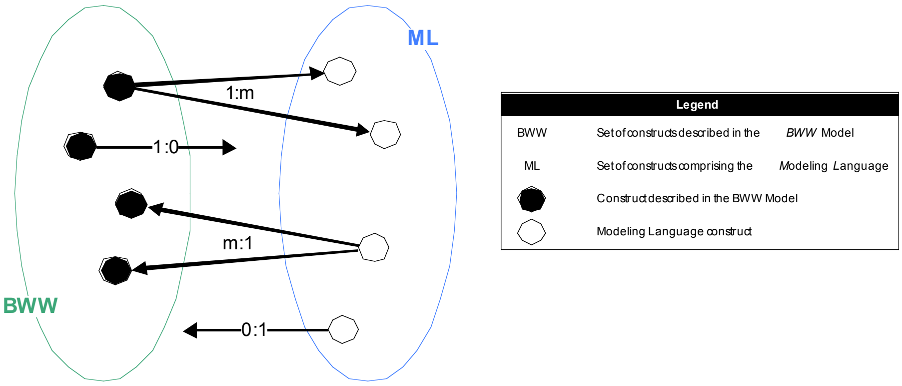

Figure 1. Representational deficiencies of a modelling language. Adapted from (Weber 1997)

Based on these four types of mappings, representation theory advocates the principle that a 'good' modelling language should be ontologically complete, i.e., it should not exhibit construct deficit. Ontological completeness implies that language users can describe all real-world phenomena that they seek to have represented by the information system they model. A 'good' modelling language should furthermore be ontologically clear, i.e., it should not exhibit construct overload, redundancy or excess. Ontological clarity implies that users can unambiguously describe the real-world phenomena that they seek to have represented without causing confusion to the end users.

The BWW representation model has been successfully used in many research projects for the evaluation of different modelling languages (see (Green & Rosemann 2004, Rosemann *et al.* forthcoming) for overviews), including data models, object-oriented models and reference models. It also has a track record in the area of process modelling, with contributions coming from various researchers. Keen and Lakos (1996), for example, determined essential features for a process modelling scheme by using the BWW representation model to evaluate the degree of ontological completeness of six process modelling languages in a historical sequence. The process modelling languages examined include the ANSI flowchart notation, the ISO Conceptual Schema Model (ISO/TC97), the Méthode d'Etude et de Réalisation Informatique pour les Systèmes d'Entreprise (MERISE), the Data Flow Diagram (DFD) notation, the Integrated Definition Method 3 Process Description Capture Method (IDEF3), and the Language for Object-Oriented Petri nets (LOOPN++). From their analysis, Keen and Lakos concluded that, in general, the BWW representation model facilitates the interpretation and comparison of process modelling languages. They propose the BWW constructs of *system*, *system composition*, *system structure*, *system environment*, *transformation*, and *coupling* to be essential process modelling language requirements.

Green and Rosemann (2000) used the BWW representation model to analyze the Event-Driven Process Chain (EPC) notation, focusing on both ontological completeness and clarity. Empirically confirmed shortcomings were found in the EPC notation with regard to the representation of real world objects, in the definition of business rules, and in the thorough demarcation of the analyzed system (Green & Rosemann 2001).

Green *et al.* (2005) also examined ebXML BPSS v1.01 in terms of ontological completeness and clarity. This study was then extended by Green *et al.* (2007) to include the examination of three other leading standards for enterprise system interoperability, including BPEL4WS v1.1, BPML v1.0, and WSCI v1.0. In addition, a minimal ontological overlap (MOO) analysis (Weber 1997) was also conducted in order to determine the set of modelling standards with a minimum number of overlapping constructs but with maximal ontological completeness (MOC), *i.e.*, maximum expressiveness, between the selected standards. The study identified two sets of standards that, when used together, allow for the most expressive power with the least overlap of constructs, *viz.,* ebXML BPSS and BPEL4WS, and, ebXML and WSCI.

While some criticisms have been leveled over the years at the use of representation theory, *viz*., limited empirical testing (Wyssusek 2006), lack of coverage and lack of understandability of the constructs (Rosemann *et al.* 2004), the work to date has attempted to mitigate such criticisms. For instance, a number of authors have undertaken empirical tests of the validity of the predictions stemming from representation theory (e.g., (Green 1997, Bodart *et al.* 2001, Green and Rosemann 2001, Shanks *et al.* 2002, Parsons and Cole 2004, Gemino and Wand 2005, Burton-Jones and Meso 2006, Recker *et al.* 2006)). These studies found that the premises offered by representation theory indeed affect conceptual modeling activities, outcomes and success, and, moreover, leverage "better" conceptual modeling. Other researchers have furthermore undertaken efforts to provide procedural guidelines for the application of the theory (Green *et al.*, 2006).

Following these procedural guidelines, Recker *et al.* (2006) employed the BWW representation model to analyse BPMN and derive a number of propositions in regards to its deficiencies. These propositions were empirically tested and majority were supported by BPMN users. Rosemann *et al.* (2006) then consolidated various BWW analyses of process modelling languages and provided a snapshot of the evolution of such languages based on the levels of their ontological completeness. We build on this work here by providing a finer granularity of the BWW analysis results – one that includes the frequency of BWW construct representation – and introducing measures *for each of the four* possible representational deficiencies, i.e., we extend the study by Rosemann *et al.* (2006) with the aspect of ontological clarity. Hence, we draw on the original analyses of the languages to also further consider their construct redundancy, overlap and excess levels, thus allowing us to compare the seven popular modelling languages based on all four ontological deficiencies. Accordingly, the next section recaps the consolidated BWW analyses of our chosen process modelling languages, provides a more detailed BWW representation analysis result and introduces the measures.

3 ANALYSIS AND MEASURES OF DEFFICIENCY

Our work focuses on already carried out BWW analyses of process modelling languages. The languages chosen for this study are: EPC, Petri nets, ebXML, BPML, WSCI, BPEL, and BPMN (refer to Figure 2). While additional process modelling languages have been analysed with the BWW representation model as the benchmark (Rosemann *et al.* 2006), these analyses only focused on the ontological completeness of the language, hence not providing us with all necessary data for a full comparison across all ontological deficiencies. The omitted languages, for example MERISE or IDEF3, also are not frequently used in practice, hence further justifying the decision to eliminate them from our study. Moreover, there is also work that concentrates on the BWW representational analysis of dynamic modelling languages (see, for example, (Opdahl & Henderson-Sellers 2002, Soffer *et al.* 2001)), however, these particular languages are not considered in our research. For example, modelling languages relying on an object-oriented paradigm (like UML, OML, OPM, or LOOPN++) have not been included in this study. These languages, applied in software engineering rather than process management contexts, have different or extended requirements in terms of representation capabilities and are, therefore, limited in comparability to 'pure' process modelling notations.

The consolidated analysis of the seven considered modelling languages is shown in Figure 2. This work is based on, and extends the work described in (Rosemann *et al.* 2006). While the focus of the work by Rosemann *et al.* (2006) studied the occurrences of BWW construct mappings only, we now consider the frequency with which BWW representation constructs are supported in the process modelling languages. Accordingly, in Figure 2, a number in a cell indicates that the process modelling language represented in the specific column provides that number of representations for the BWW construct represented in the specific row. An 'x' in a table cell represents that a BWW construct can be represented in the modelling languages, but not necessarily all variations of the construct are supported. This situation is only relevant to the BWW property construct, which is further specified as different types of properties in the BWW model *e.g.* emergent property, mutual binding property etc.

The consideration of the BWW property construct at the higher level is necessary because not all previous studies considered the various property specialisations.

BWW Construct / Language / Version / Year	Petri net	EPC	ebXML 1.01	BPML 1.0	WSCI 1.0	BPEL4WS 1.1	BPMN 1.0
	1962	1992	2001	2002	2002	2003	2004
Things including Properties and Types of Things							
THING	1						2
CLASS	1		3	1	1	1	2
KIND							1
PROPERTY		x	x	x	x	x	x
States assumed by Things							
STATE	3	1	5	1	1	1	
CONCEIVABLE STATE SPACE			1				
STATE LAW	1	1	1				
LAWFUL STATE SPACE	1		1				
STABLE STATE		1	4				
UNSTABLE STATE	3		1				
HISTORY			1				
Events and Transformations occurring on Things							
EVENT	1	1	3	3	6	4	9
CONCEIVABLE EVENT SPACE			1				
LAWFUL EVENT SPACE			3				
EXTERNAL EVENT		1	3	2	3	1	8
INTERNAL EVENT	1	1	4	1	3	3	8
WELL-DEFINED EVENT	1	1	1	2	2	1	2
POORLY DEFINED EVENT			2	1	1	2	7
TRANSFORMATION	1	1	1	10	8	11	6
LAWFUL TRANSFORMATION	1	1	1	4	4	3	7
ACTS ON	1					1	1
COUPLING			2		1	1	1
Systems structured around Things							
SYSTEM			2		1	1	2
SYSTEM COMPOSITION					1	1	2
SYSTEM ENVIRONMENT							2
SYSTEM STRUCTURE					1	1	
SUBSYSTEM			1				2
SYSTEM DECOMPOSITION							2
LEVEL STRUCTURE		1					2

Figure 2. Consolidated BWW representational analysis of the chosen modelling languages

3.1 Measures of representational deficiency

In order to enable comparison between the languages, we now introduce four measures of their potential representational deficiencies. That is, we introduce a measure for ontological completeness and three measures that, together, can be used to approximate, and assess, ontological clarity.

For the purpose of objectively comparing the ontological completeness of a process modelling language, we propose the *Degree of Deficit (DoD)* measure. DoD can be measured relatively as the number of BWW constructs found *not* to have a mapping to language constructs divided by the *total* number of constructs defined in the BWW representation model.

For the purpose of objectively comparing the ontological clarity of a process modelling language, we propose three measures *viz. Degree of Redundancy (DoR), Degree of Overlap (DoO) and Degree of Excess (DoE),* for the earlier discussed construct redundancy, construct overlap and construct excess situations, respectively. DoR can be measured relatively as the number of language constructs found to have a mapping to the *same* BWW construct divided by the *total* number of constructs in the modelling language. DoE can be measured relatively as the number of language constructs found *not* to have a mapping to any BWW construct divided by the *total* number of constructs in the modelling language. Last, DoO can be measured relatively as the number of language constructs found to have a mapping to *more than one* BWW construct divided by the *total* number of constructs in the modelling language.

On the basis of these four measures, our forthcoming analysis rests on a very simple observation: A high degree of ontological completeness (i.e., a low degree of construct deficit) of a process modelling language would mean that users are able to depict all relevant aspects of real-world domains they seek to have articulated in a process model. As such, the degree of deficit is indicative of the coverage capacity of a process modelling language. This coverage capacity, however, can be negatively affected by a lack of ontological clarity. The clarity of a language describes how unambiguously the meaning of its constructs is specified and, thus, how much effort is needed to ascribe desired real-world meaning to the constructs.

In other words, while a certain language may provide sufficient representation (indicated by a low degree of construct deficit), it may be complex to use because this scope of coverage may come at the expense of redundant, overloaded or excessive constructs (indicated by high degrees of construct redundancy, overload and excess). Hence, representation theory offers the theoretical foundation for measuring the complexity of a process modelling language: given a certain scope of coverage, as defined by the degree of deficit in a language, one is able to gauge, and compare, the complexity of the language by establishing its degrees of redundancy, overload and excess.

In the next section, we present the results of the comparison of the four measures for the chosen languages. Due to space considerations we do not present here the full analyses of construct redundancy, construct overlap and construct excess. Instead, we focus on the comparative results and reserve this finer detail for an extension of the paper and for presentation at ECIS.

4 RESULTS

Having applied the introduced measures to the BWW analyses of the chosen process modelling languages (see summarised results in Table 1), we are able to visualise the results in a series of radar charts that enable easy comparison of the languages' representational deficiencies. The radar charts are shown in Figure 3, with the degrees of deficit and excess represented on the Y-axis and the degrees of overlap and redundancy represented on the X-axis. All measures are represented in percentage terms.

Technique	DoD	DoR	DoE	DoO
Petri Net	58.62%	28.57%	0.00%	42.86%
EPC	62.07%	0.00%	42.86%	28.57%
ebXML 1.01	27.59%	15.69%	13.73%	1.96%
BPML 1.0	65.52%	30.43%	28.26%	0.00%
WSCI 1.0	48.28%	30.61%	18.37%	4.08%
BPEL4WS 1.1	48.28%	31.91%	12.77%	2.13%
BPMN 1.0	34.48%	51.28%	38.46%	25.64%

Table 1. Ontological completeness and clarity of process modelling techniques

From Figure 3, some interesting conclusions can be drawn in regards to the representational deficiencies of process modelling languages. Clearly, the capability of ebXML is closest to the general principles of representation theory, as its relatively low degree of deficit (28%) is complemented by low degrees of redundancy (16%), excess (14%) and overload (2%). It can thus be assumed that the use of ebXML not only enables modellers to create reasonably complete descriptions of real-world domains but also relatively clear descriptions that bear little complexity and can unambiguously be interpreted. On the other hand, EPC, unlike ebXML, exhibits no construct redundancy (0%) but a very high deficit of constructs (62%) as well as comparatively medium and low degrees of excess (43%) and overload (29%) respectively. Accordingly, based on the BWW representation model, the use of the EPC language for modelling initiatives can be problematic due to its high deficit of BWW constructs, and significant levels of excess constructs. Despite being the only considered language to exhibit no construct redundancy, the situation implies that users will encounter ambiguity in their choice of modelling constructs as they struggle to ascribe meaning to EPC constructs that do not appear to clearly represent real world objects and struggle to find representations for such real world objects.

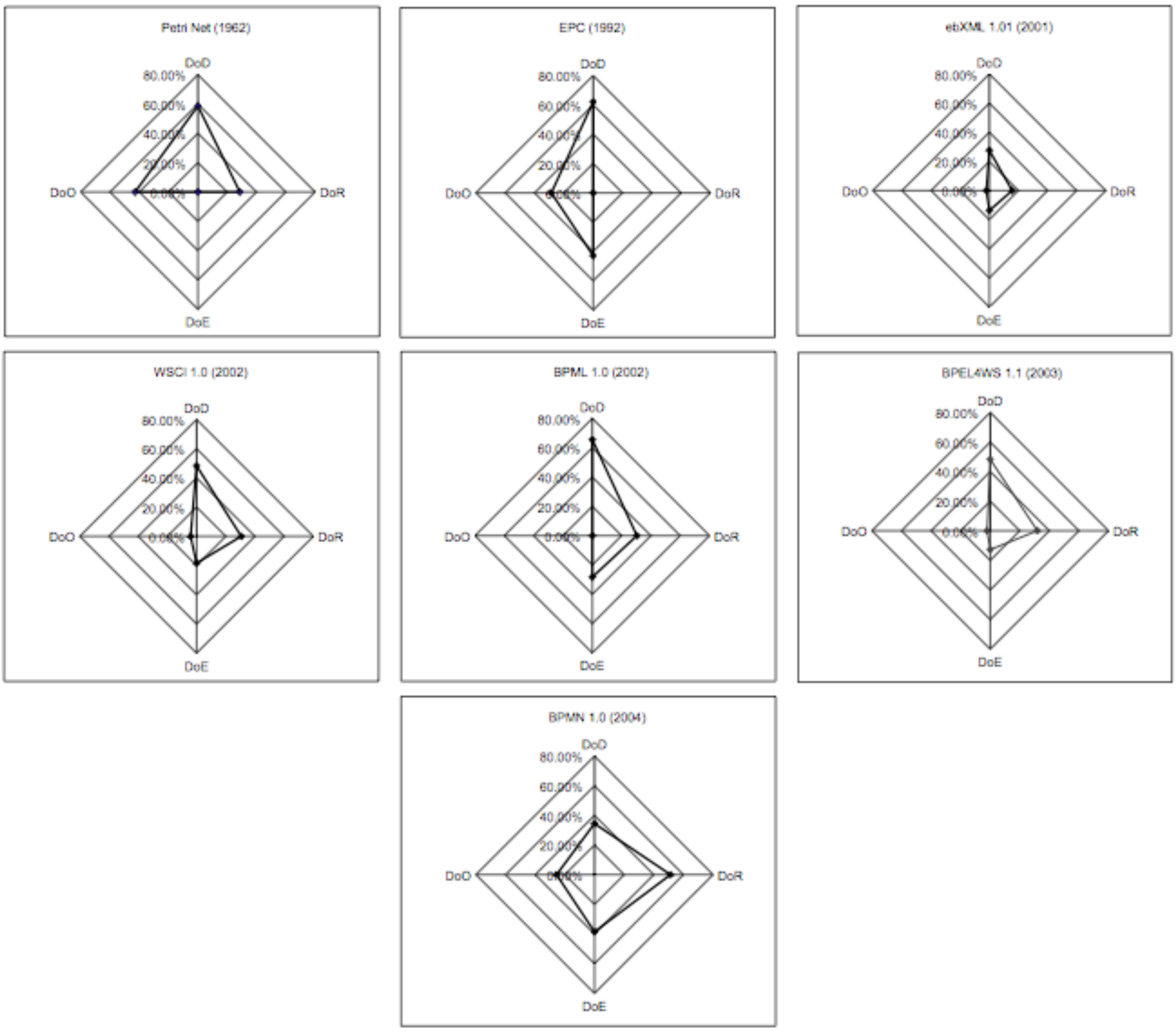

Figure 3. Radar charts of ontological completeness and clarity based on introduced measures

A scan of Figure 3 also shows that Petri nets is the only language to not exhibit any degree of construct excess (0%). This is a desirable situation as it prevents user confusion due to the need to ascribe meaning to constructs that do not appear to have a real world meaning. This situation is also helped by the relatively low levels of construct redundancy in Petri nets. However, the language does exhibit a high degree of deficit, which appears to lead to a moderate degree of overload as real world phenomena are represented by the same BWW construct due to the lack of its representation. This situation in itself leads to the development of models that are ambiguous in their interpretation.

The visualisation of the measures also shows that BPML, WSCI and BPEL4WS share commonalities in the degrees of their representational deficiencies. Each of the three languages shares a low degree of overload (0%, 4% and 2% respectively), which means that the language constructs tend to represent uniquely a specific real world construct, thus avoiding confusion when interpreting the resulting models. Each of the three languages also exhibits relatively low degrees of excess (28%, 18% and 13% respectively), implying that unlike in the case of the EPC language, users are not confronted with many constructs of which the their real world meaning is not clear. Further, the languages have relatively low degrees of construct redundancy (30%, 31% and 32% respectively), whereby multiple BPML, WSCI and BPEL4WS constructs can represent the same real world concept, of the real world concepts that these languages are actually able to model. However, these mostly positive aspects of relatively low degrees of overlap, excess and redundancy are overshadowed by high (BPML) and medium (WSCI and BPEL4WS) degrees of deficit (66%, 48% and 48% respectively), which implies a lack of representation for a significant amount of real world objects.

Last, BPMN, while obtaining a considerably low degree of deficit (34%), exhibits high degrees of deficiency across all language clarity aspects (DoR: 51%; DoE: 38%; DoO: 26%). The use of BPMN can thus be expected to lead to quite complete but also unclear and potentially ambiguous representations of real-world domains – a problematic finding given the popularity of the BPMN language in industry (zur Muehlen 2007).

Further to the above comparison of representational deficiencies, two interesting, more general, patterns can be observed from Figure 3. First, some languages, such as Petri nets, achieve low degrees of redundancy and excess with high degrees of overload. The scope of coverage of these languages is thus obtained through a rather restricted set of language constructs, which in turn are subject to overload. From this observation a language design principle emerges that advocates a process modelling language specification with a minimal set of language constructs that is at the same time very flexible in meaning and purpose. The use of such a language would thus not bear complexity due to a surplus of equivalent or excessive language constructs. However, the resulting models may still be prone to understandability concerns as the used language constructs have, *prima facie*, multiple meanings in the model.

As opposed to this, a second set of languages, such as BPML or WSCI, achieve a low degree of overload with high degrees of redundancy and excess. Their graphs in Figure 3 thus correspond more to a triangle between the dimensions of deficit, redundancy and excess. The observable underlying language design principle is coined by a language specification that offers an extensive set of language constructs for modelling that, while being clear in specification (indicated by a low degree of overload), are potentially redundant and/or excessive. In consequence, such languages achieve a certain scope of coverage through a multitude of constructs, which in turn, *prima facie*, offer too many choices for representing the real-world phenomena the user seeks to describe. Such design principle appears to be based on language extension rather than revision and clarification.

The consolidated overview of the representational capabilities of process modelling languages in Figure 2 can be used to guide relevant stakeholders in the selection of an appropriate process modelling language. Based on preferences that stem from factors such as modelling role or modelling purpose, a language that is potentially redundant in its use may or may not be favourable in contrast to a language that is neither excessive nor redundant but overloaded in its constructs. While the overall objective of providing complete representations of real-world domains can be regarded as given, certain trade-offs can be made with respect to the 'costs of clarity' with which the desired scope of coverage can be achieved. The investigation of such preferences and trade-offs, however, is outside the scope of this paper and is designated as future work.

5 CONCLUSIONS AND FUTURE WORK

This paper strengthened previous contributions of representational analyses of process modelling languages by presenting a finer granularity of analysis and introducing four measures. The introduced measures are those of degree of deficit (DoD), degree of redundancy (DoR), degree of overload (DoO) and degree of excess (DoE). The application of the measures to previous BWW analyses of process modelling languages leads to a clearer comparison of representational capability of the languages, and a better understanding of the levels and types of potential confusion that may be introduced by the use of these languages. The work uncovered a pattern within the degrees of representational capability and clarity of languages such as BPML, WSCI, BPEL4WS, and to a much lesser extent ebXML, as well as identifying fairly distinct design principles behind languages such as EPC, Petri nets and BPMN.

While the work provides a useful visual comparison of representational deficiencies of the chosen languages, there exist some limitations. First, the original BWW analyses were in some cases performed by different research teams and only consolidated for the purposes of this work. This situation might introduce some inconsistency in the analysis of representation capabilities. Second, not all considered analyses have been empirically validated with the language users, hence at this stage (with the exception of EPC and BPMN) the results indicate *potential* degrees of deficiency as suggested by the BWW representation model. However, our analytical study has led to a number of interesting propositions that can now be tested by means of empirical research strategies. Third, while the four introduced measures offer a quick and objective comparison of the degrees of deficiency, they only indicate the criticality of the deficiency and do not indicate whether any two languages are able to represent the same set of real world concepts. For example, two languages with the same degree of deficit may be able to represent different sets of real world concepts. This situation, in turn, implies that the measures are a useful first step in analysis of a language for its suitability to a given modelling role or purpose, but should be followed by investigating suitability to the modelling task by looking at the full results of the BWW representation mapping.

Future work in this area is continuing with efforts to empirically validate the deficiencies predicted by the application of the BWW representation model for the purpose of analysis of the chosen process modelling languages. We are also investigating complementing the BWW analysis of selected languages with other frameworks that might widen the scope of the comparison.

References

Andrews, T., F. Curbera, H. Dholakia, Y. Goland, J. Klein, F. Leymann, et al. (2003). Business Process Execution Language for Web Services. Version 1.1. Retrieved February 10, 2006, from http://xml.coverpages.org/BPELv11-May052003Final.pdf.

Arkin, A. (2002). Business Process Modeling Language. Retrieved January 16, 2006, from http://www.bpmi.org/.

Arkin, A., S. Askary, S. Fordin, W. Jekeli, K. Kawaguchi, D. Orchard, et al. (2002). Web Service Choreography Interface (WSCI) 1.0. Retrieved January 16, 2006, from http://www.w3.org/TR/wsci/.

Bodart, F., A. Patel, M. Sim, and R. Weber (2001) "Should Optional Properties Be Used in Conceptual Modelling? A Theory and Three Empirical Tests," Information Systems Research (12) 4, pp. 384-405.

BPMI.org and OMG (2006). Business Process Modeling Notation Specification. Final Adopted Specification. Retrieved February 20, 2006, from http://www.bpmn.org.

Bunge, M.A. (1977). Treatise on Basic Philosophy Volume 3: Ontology I - The Furniture of the World. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Burton-Jones, A. and P. Meso (2006) "Conceptualizing Systems for Understanding: An Empirical Test of Decomposition Principles in Object-Oriented Analysis," Information Systems Research (17) 1, pp. 38-60.

Davies, I., P. Green, M. Rosemann, M. Indulska and S. Gallo (2006). How do Practitioners Use Conceptual Modeling in Practice? Data & Knowledge Engineering, 58 (3), 358-380.

Floyd, C. (1986). A Comparative Evaluation of System Development Methods. In Information System Design Methodologies: Improving the Practice (Olle, T.W., H.G. Sol and A.A. Verrijn-Stuart, Eds.), pp. 19-54, North-Holland, Amsterdam, The Netherlands.

Gartner Group (2007). Creating Enterprise Leverage: The 2007 CIO Agenda. In EXP Premier Report No. January2007. Gartner, Inc, Stamford, Connecticut.

Gemino, A. and Y. Wand (2005) "Complexity and Clarity in Conceptual Modeling: Comparison of Mandatory and Optional Properties," Data & Knowledge Engineering (55) 3, pp. 301-326.

Green, P. (1997) "Use of Information Systems Analysis and Design (ISAD) Grammars in Combination in Upper CASE Tools - An Ontological Evaluation." 2nd CAiSE/ IFIP8.1 International Workshop on Evaluation of Modeling Methods in Systems Analysis and Design, Barcelona, Spain, 1997, pp. 1-12.

Green, P. and M. Rosemann (2000). Integrated Process Modeling. An Ontological Evaluation. Information Systems, 25 (2), 73-87.

Green, P. and M. Rosemann (2001). Ontological Analysis of Integrated Process Models: Testing Hypotheses. Australasian Journal of Information Systems, 9 (1), 30-38.

Green, P. and M. Rosemann (2004). Applying Ontologies to Business and Systems Modeling Techniques and Perspectives: Lessons Learned. Journal of Database Management, 15 (2), 105-117.

Green, P., M. Rosemann and M. Indulska (2005). Ontological Evaluation of Enterprise Systems Interoperability Using ebXML. IEEE Transactions on Knowledge and Data Engineering, 17 (5), 713-725.

Green, P., M. Rosemann, M. Indulska and C. Manning (2007). Candidate Interoperability Standards: An Ontological Overlap Analysis. Data & Knowledge Engineering, 62 (2), 274-291.

Indulska, M., S. Chong, W. Bandara, S. Sadiq and M. Rosemann (2006). Major Issues in Business Process Management: An Australian Perspective. In Proceedings of the 17th Australasian Conference on Information Systems (Spencer, S. and A. Jenkins, Eds.), Australasian Association for Information Systems, Adelaide, Australia.

Keen, C.D. and C. Lakos (1996). Analysis of the Design Constructs Required in Process Modelling. In Proceedings of the International Conference on Software Engineering: Education and Practice (Purvis, M., Ed.), pp. 434-441, IEEE Computer Society, Dunedin, Ireland.

OASIS (2001). ebXML Business Process Specification Schema Version 1.01. Retrieved March 12, 2005, from http://www.ebxml.org/specs/ebBPSS.pdf.

Opdahl, A.L. and B. Henderson-Sellers (2002). Ontological Evaluation of the UML Using the Bunge-Wand-Weber Model. Software and Systems Modeling, 1 (1), 43-67.

Parsons, J. and L. Cole. (2004) "An Experimental Examination of Property Precedence in Conceptual Modelling." 1st Asian-Pacific Conference on Conceptual Modelling, Dunedin, New Zealand, 2004, pp. 101-110. ACM International Conference Proceeding Series 59.

Petri, C.A. (1962). Fundamentals of a Theory of Asynchronous Information Flow. In IFIP Congress 62: Information Processing (Popplewell, C.M., Ed.), pp. 386-390, North-Holland, Munich, Germany.

Recker, J., M. Indulska, M. Rosemann and P. Green (2006). How Good is BPMN Really? Insights from Theory and Practice. In Proceedings of the 14th European Conference on Information Systems

(Ljungberg, J. and M. Andersson, Eds.), pp. 1582-1593, Association for Information Systems, Goeteborg, Sweden.

Rosemann, M. and P. Green (2002). Developing a Meta Model for the Bunge-Wand-Weber Ontological Constructs. Information Systems, 27 (2), 75-91.

Rosemann, M., P. Green, and M. Indulska (2004) “A Reference Methodology for Conducting Ontological Analyses,” in, vol. 3288 H. Lu, W. Chu, P. Atzeni, S. Zhou et al. (Eds.) Conceptual Modeling – ER 2004, Shanghai, China: Springer, pp. 110-121.

Rosemann, M., P. Green, M. Indulska and J. Recker (forthcoming). Using Ontology for the Representational Analysis of Process Modeling Techniques. International Journal of Business Process Integration and Management, in press.

Rosemann, M., J. Recker, M. Indulska and P. Green (2006). A Study of the Evolution of the Representational Capabilities of Process Modeling Grammars. In Advanced Information Systems Engineering - CAiSE 2006 (Dubois, E. and K. Pohl, Eds.), pp. 447-461, Lecture Notes in Computer Science 4001, Springer, Luxembourg, Grand-Duchy of Luxembourg.

Scheer, A.-W. (2000). ARIS - Business Process Modeling. 3rd Edition. Springer, Berlin, Germany.

Shanks, G., E. Tansley, J. Nuredini, D. Tobin et al. (2002) “Representing Part-Whole Relationships in Conceptual Modelling: An Empirical Evaluation.” 23rd International Conference on Information Systems, Barcelona, Spain, 2002, pp. 89-100.

Shanks, G., E. Tansley and R. Weber (2003). Using Ontology to Validate Conceptual Models. Communications of the ACM, 46 (10), 85-89.

Soffer, P., B. Golany, D. Dori and Y. Wand (2001). Modelling Off-the-Shelf Information System Requirements. An Ontological Approach. Requirements Engineering, 6 (3), 183-199.

Wand, Y. and R. Weber (1990). An Ontological Model of an Information System. IEEE Transactions on Software Engineering, 16 (11), 1282-1292.

Wand, Y. and R. Weber (1993). On the Ontological Expressiveness of Information Systems Analysis and Design Grammars. Journal of Information Systems, 3 (4), 217-237.

Wand, Y. and R. Weber (1995). On the Deep Structure of Information Systems. Information Systems Journal, 5 (3), 203-223.

Weber, R. (1997). Ontological Foundations of Information Systems. Coopers & Lybrand and the Accounting Association of Australia and New Zealand, Melbourne, Australia.

Wyssusek, B. (2006) "On Ontological Foundations of Conceptual Modelling," Scandinavian Journal of Information Systems (18) 1, pp. 63-80.

zur Muehlen, M. (2007). Research in Business Process Management. BPTrends, 4 (5), 1-7.