Modeling Design-Time Variability in Business Processes: Existing Support and Deficiencies

Inbal Mechrez and Iris Reinhartz-Berger

Information Systems Department,
University of Haifa, Haifa 31905, Israel
inbalma@gmail.com, iris@is.haifa.ac.il

Abstract. Recently the interest in managing families of business processes rather than individual processes has increased, mainly due to the need to maintain different variants of the same business process or similar business processes in the same organization. This led to the extension of different business process modeling languages (BPMLs) in order to support the representation of design-time variability, namely variability that is resolved when designing the particular business processes (the variants). However, the evaluation of these languages expressiveness is still in an inceptive stage. In particular, the abilities to express variable elements in different granularity levels and to guide variability in business process models have not been examined. To tackle this lack, we propose a two-dimensional framework which explicitly refers to granularity and guidance. We further examine how existing extensions of BPMLs support these dimensions, point on deficiencies in their expressiveness, and discuss the implications of those deficiencies through examples from a case study.

Keywords: Variability Modeling, Design Time Variability, Business Process Modeling, Configuration.

1 Introduction

Business processes have drawn much attention over the years [4]. They affect organization's performance, cost, and customer's satisfaction and are considered one of the key concepts to successful businesses. A common way to present the specification of business processes is through business process models which capture different aspects of business processes, such as their goals and constraints, their activities and flow, their events and resources, and the different organizational units or roles involved in their execution.

Various graphical languages have been proposed over the years to model business processes. These languages aim to bridge the gap between business process design and implementation, as well as to represent and communicate different aspects of business processes to various stakeholders. This is done in different ways: *imperative* business process modeling languages (BPMLs), for example, focus on *how* the process is executed (mainly, its activities and flow), while *declarative* BPMLs support the description of *what* should be done and not how it is done [25]. *Graph-based*

I. Bider et al. (Eds.): BPMDS 2014 and EMMSAD 2014, LNBIP 175, pp. 378–392, 2014.

BPMLs visually specify business processes as graphs, while *rule-based* BPMLs support abstracting the process logic into a set of rules [22]. There are BPMLs that mainly focus on *input/output flows*; others focus on *workflows* (namely, time ordering of activities); a third group concentrates on *agent* cooperation; and a fourth group is considered *state-based* [1].

Usually a single organization deals with a large number of business processes. The different business processes are not necessarily far apart from each other; they may be variants which are commonly considered as specializations of ”abstract" business use cases [31]. Sometimes the existence of such variants indicates on large differences in the instances of the business process. This kind of variability is commonly referred to as *runtime variability* [39]. In other cases the need to handle variable aspects is raised during the process design phase and requires designing and managing variants of the same business process or business process part for different organizational units, market segments, or involved items. In this case, which is commonly referred to as *design time variability* [39], variability is resolved at design-time and not at run-time, potentially making the variants more suitable to the specific business process needs, but less flexible.

Another interest in design-time variability is raised by software companies that aim to develop COTS products or process-aware information systems (PAIS) [8] for a market segment that includes organizations that have similar core business processes. These software companies may benefit from treating the different business processes as a family, monitoring and analyzing process commonality and variability. The results of such analysis can be incorporated into the products development, yielding flexible products that can be adapted to meet the specific needs of a particular organization in that market segment.

In order to support design-time variability in business processes, several modeling languages have been suggested in the last decade. Most of these languages extend existing languages, and especially BPMN and EPC, with variability aids, e.g., [26, 28, 31], or suggest aids to specify the variability orthogonally to the business process models, e.g., [13, 40]. However, the evaluation of these languages expressiveness is still in an inceptive stage. In this paper, we propose a two-dimensional framework that refers to *granularity*, namely, the variable elements, and *guidance*, i.e., the creation of variants at design-time. We use this framework for evaluating the expressiveness of 22 languages that support design-time variability modeling in business processes.

The contribution of this paper is two folded. First, it provides a useful input for practitioners by pointing and discussing the deficiencies of the different languages and assisting in language selection. Second, the deficiencies are also of interest to researchers who wish to know what languages should be worked on and in what directions.

The rest of the paper is structured as follows. Section 2 reviews the relevant literature. Section 3 elaborates on the suggested framework. Section 4 reviews existing BPMLs that support design-time variability modeling and discusses their expressiveness based on the suggested framework. Section 5 presents and exemplifies the found deficiencies, as well as discusses their implications. Finally, Section 6 concludes and refers to future research.

2 Related Work

A few studies have already examined variability modeling in business processes. Torres et al. [38] present two approaches targeted at the representation of process families: the *behavioral approach* which derives a process variant by hiding and blocking elements, and the *structural approach* which applies a set of change operations on a base model in order to derive a process variant. The authors further compare two specific languages: C-EPC [28], which is a behavioral approach, and Provop [13], which is a structural approach. The comparison is done in terms of understandability of the produced process model.

Ayora et al. [2] propose an approach that refers to variability concepts as first order elements in business process models. This approach is evaluated with respect to C-EPC [28], PESOA [26], and Provop [13], using a set of criteria. Some of these criteria refer to variability concepts, including variation points, process fragments, process fragment context, process fragment relationships, language support regarding variability, process context regarding variability, and variation point resolution time. The other criteria define quality factors, such as flexibility, scalability, and understanding.

Vervuurt [40] defines nine criteria that need to be considered when evaluating business process variability modeling languages, including: (1) the ability to mark variable elements, (2) the support of change patterns, (3) the configuration rules that adapt process model, (4) visualization of configuration rules that adapt process models, (5) domain visualization and process model configuration, (6) domain and process configuration rules, (7) selective display, (8) correctness, and (9) consistency. All these criteria focus on configuration as the main mechanism for creating variants. Vervuurt further uses the nine criteria for comparing and evaluating specific modeling languages, namely: C-EPC, BPMN, and Extended EPC (E-EPC). Based on the comparison findings, alternative solutions to business process variability modeling problems are suggested: combining C-EPC with feature diagrams (Feature-EPC), extending C-EPC with Change-Oriented Versioning (COV-EPC), and utilizing Proteus Configuration Language (PCL-EPC).

La Rose et al. [17] review three approaches to capture variability in business process models: (1) configurable nodes, e.g., C-EPC, (2) hiding & blocking, which aim to represent choices in configurable process models independently of the language, e.g., Configuration in SAP WebFlow [10], and (3) annotation-based process variability which aim to "improve the customization of process-oriented software systems", e.g., the study in [35]. They further claim that the existing languages do not provide sufficient support during the actual configuration of the generic or configurable process model, commonly termed the *reference model*. Thus they suggest independent representations of the variability that can be used to complement these approaches: questionnaires models, feature diagrams, and adaptive mechanisms.

Weidmann et al. [41] specifically refer to the variability scope, but only within BPMN 2.0, concluding that events, activities, gateways, sequence & message flows, and pools & lanes can have variable attributes. They further compare four approaches to variability modeling in business processes, namely Provop, PESOA, Process Configuration (ProCon) and Multi-Perspectives Variants (MultPers). The comparison, which is based on five criteria that focus on variability and dependency visualization in process models, leads to the conclusion that the interaction of the user with the process model is missing. Thus, an approach for Adaptive Business process modeling in the Internet of Services (ABIS) is presented. This approach enables business users to create their own process variants using process templates and process fragments.

Discussing techniques that deal with the management of process model variants, Dijkman et al. [7] distinguish between techniques that use a single consolidated model to capture the process variants and techniques that keep the process variants separate. The first group of techniques mainly utilizes variation points to distinguish between the common and variable parts. Representing variability in this kind of techniques may rely on configuring nodes (e.g., [11], [13], [18], [31]), attaching parameters to nodes or marking nodes with stereotypes (e.g., [35]), assigning cardinalities to arcs and nodes (e.g., [29]), or using aspect-oriented principles (e.g., [21]). The second group of techniques "leaves the various variants separate, but provides an infrastructure to identify and keep track of their commonalities in order to maintain consistency across variants when updating them" [7]. This can be done, for example, by utilizing the inheritance mechanism, using version control techniques, or identifying behavioural relations between process variants.

The above studies examine and compare a few extensions of BPMLs that support design-time variability modeling. However, these studies treat the business process models as a whole, without separately referring to different business process elements and to the way they vary. As variability may be present in different granularity levels, it is important to know the variability of which elements is supported by a certain BPML. In addition, the support that BPMLs provide to (re)use variable elements in specific business processes is not sufficiently analyzed in those studies, which mainly concentrate on a single mechanism – configuration, and do not examine the relationships between the process elements and the utilized mechanisms.

3 The Suggested Evaluation Framework

To tackle the aforementioned limitations, our framework refers to two dimensions for evaluating design-time variability in business process models: *granularity*, which refers to the variable business process elements, and *guidance*, which refers to the mechanisms to create variable elements (i.e., variants).

3.1 Granularity Dimension

Curtis et al. [6] refer to four perspectives of business processes: functional, behavioral, organizational, and informational. These perspectives are also mentioned in List and Korherr's metamodel [21], which was inspired by ARIS [34]. These perspectives, which are briefly reviewed below, are of high relevance to many BPMLs, which are classified as imperative. Thus, we set them and their high-level elements as the values of the granularity dimension.

The ***functional perspective*** represents *what* process elements are being performed [6]. The main elements in this perspective are atomic tasks and composite activities. ***Atomic tasks*** (also termed process elements or process steps) are functional units of a process that have no externally visible substructure. ***Composite activities***, on the other hand, represent major units of work that need to be performed in order to achieve the objective of a process. Composite activities are commonly described as sets of partially ordered tasks.

The ***behavioral perspective*** represents *when* activities are performed as well as aspects of *how* they are performed [6]. In particular, the behavioral perspective describes the order in which the different activities are executed (***control flows***) and when process elements are performed (i.e., ***sequence flows***). Moreover, ***data flows*** are used to connect atomic tasks with information resources (such as data, artifacts, and products) [16].

The ***organizational perspective*** describes the organization structure and, in particular, *where* and *by whom* (which agents) process elements are performed. Three types of process participants are commonly mentioned [21]: (1) an ***organizational unit***, which is a group of people organized for some purpose; (2) a ***role***, which is a group of process elements exhibiting a set of specific skills or qualifications and assigned to an agent; and (3) ***software***, e.g., applications and services, which automatically performs process elements.

The ***informational perspective*** represents the information and data produced or manipulated by a process and their interrelationships [6]. The informational perspective describes *which* information is involved in the business process, *how* it is represented, and *how* it is propagated among different activities. The elements of the informational perspective are primarily divided into resources and events: an ***event*** may trigger an activity or a task, whereas a ***resource*** is an entity to be produced or consumed by an atomic task, e.g., data, products, and artifacts.

Table 1 summarizes the granularity dimension in terms of perspectives, relevant questions, and high-level elements.

Table 1. The granularity dimension

Perspective	Relevant Questions	High-Level Elements
Functional	- *What* process elements are being performed?	- Atomic tasks - Composite activities
Behavioral	- *When* are process elements performed? - *How* are process elements performed?	- Sequence flows - Control flows - Data flows
Organizational	- *What* is the organization structure? - *Where* and *by whom* are the process elements performed?	- Organization units - Roles - Software
Informational	- Which information is involved? - How is it represented? - How is it propagated among different process elements?	- Resources - Events

Note, however, that there are a few business process elements, mentioned in the literature, which cannot be naturally classified into one of the four perspectives. These elements are either more abstract than the elements in the four perspectives (e.g., goals, soft-goals, and domains), or are needed for evaluation or classification purposes (e.g., process types, measures, and dimensions). List and Korherr [21] classify these elements under a fifth perspective called business process context. Although these elements are important for the completeness of the process models, their number and nature made us currently leave them out of the framework scope.

3.2 Guidance Dimension

The second dimension refers to the ways variability of a business process family can be resolved in order to create specific business processes. These ways are commonly termed *variability mechanisms* or *reuse mechanisms* in the area of reference modeling and business process families [5, 35]. Table 2 lists four common mechanisms, which were established as the values of the guidance dimension, their descriptions, and related terms.

Table 2. The guidance dimension

Variability mechanism	Related terms	Description
Configuration	Inclusion, exclusion, selection, blocking, hiding, deletion	Enables selecting process elements for inclusion
Inheritance	Specialization, encapsulation, uses	Enables specializing process elements
Parameterization	Parameters, values	Enables customizing process elements by assigning values to parameters
Extension	Addition, insertion	Enables attaching several variants (process elements) at a certain point at the same time

Configuration and parameterization are classified in [35] as basic variability mechanisms, as they are standalone and do not require any other variability mechanisms or new model design. Inheritance and extension, on the other hand, are variability mechanisms derived by restriction. Nevertheless, all the four are common variability mechanisms in business process modeling.

4 BPMLs That Support Design-Time Variability Modeling

Searching for BPMLs that have been suggested to model design-time variability in business processes, we found 22 such languages published since 2005. All of them are graph-based languages and most of them are imperative. 7 languages are based on BPMN and 6 on EPC. A few languages are based on other BPMLs: YAWL (2), UML Activity Diagrams (AD) (2), UML State machines (1), EWF-nets (1), Petri-nets (1), goal models (1), and SAP WebFlow (1). Most of the languages (20 out of 22) extend the base notation and introduce a single (unique) model that captures both commonality and variability. This kind of languages is commonly called *annotation-based* as variability is annotated on the base model. Two languages distinguish and keep the base model separate from the variability model. This kind of languages is termed *composition* as it proposes ways to combine or compose the two separately handled models, the base and the variability models.

Table 3 details, for each extension of BPML, the variability of which perspectives and high level process elements is supported and how, namely with which variability mechanisms. The grayed rows are languages that follow a composition-based approach (all others are annotation-based languages). As can be seen, variability is not uniformly supported with respect to the granularity and guidance dimensions: there are neglected perspectives, neglected elements, neglected variability mechanisms, and neglected combinations. We next discuss these deficiencies and exemplify their implications with examples from a case study.

5 Deficiencies in Business Process Variability Modeling

Conducting the evaluation of the reviewed extensions of BPMLs according to the granularity and guidance dimensions, we can find several deficiencies with respect to design-time variability modeling in business processes. To examine whether these deficiencies indicate real limitations, we conducted a case study for examining the variability of procurement processes in two organizations: a university library and an industrial company dealing with defense electronics. We collected data on the procurement processes and their variability through interviews, observations, and existing documents. We qualitatively analyzed the data and classified each variability type according to the two suggested dimensions. Due to space limitations, we briefly discuss here each deficiency and exemplify the implications with examples from the case study.

5.1 Deficiencies with Respect to the Granularity Dimension

Neglected Perspectives. Business processes may differ in what they are doing (the functional perspective), how and when they are doing that (the behavioral perspective), where and by whom they are doing that (the organizational perspective), and which information is required and in what way (the informational perspective). The business processes in a certain organization may vary only in specific perspectives and not in all of them. For example, procurement processes may vary in when and how they are performed and not in what they are doing, where and by whom they are doing that, and which information is required and in what way. In this case the expressiveness of variability modeling in the behavioral perspective is important, requiring a BPML whose expressiveness in this category is high.

As can be seen in Table 3, the most handled perspectives are the functional and the behavioral ones. These perspectives are the most prominent in “regular” BPMLs and here we see that they remain prominent when dealing with design-time variability. Variability modeling in the informational perspective, on the other hand, is supported to some extent. Several studies refer to variability in different data-related resources, such as data storage, objects, inputs, and outputs. Variability of events is partially handled in four studies. A possible reason for this low support may be that events, as opposed to functional and behavioral units and data-related elements, are considered external and independent of the organization (i.e., often the organization cannot directly affect the events of its business processes) [28]. Thus, events are individually handled and their variability is not commonly modeled.

Table 3. Design-time variability modeling in existing extensions of BPML*

Category	**Study**	**Functional Perspective**		**Behavioral Perspective**			**Organizational Perspective**			**Info. Perspective**	
		atomic tasks	**composite activities**	**control flows**	**sequence flows**	**data flows**	**org. units**	**roles**	**software**	**events**	**resources**
BPMN-based	PESOA for BPMN [26]		c e i p		p	p					p
BPMN-based	DeCo [33]	c i						c i			c i
BPMN-based	ADOM-BPMN [29]	c i		c i		c i					c i
BPMN-based	BPFM [23]	c									
BPMN-based	AO4BPMN 2.0 [24]		c i							c i	
BPMN-based	ABIS [41]	c	c	c	c	c					
BPMN-based	Provop [13]	c e i	c e i	c e		c e					
EPC-based	C-EPC [28], [31]	c		c							
EPC-based	PCL-EPC [40]	c		c	c					c	
EPC-based	ADOM-EPC [30]	c i		c i						c i	
EPC-based	c-iEPC [18]			c				c i			c i
EPC-based	COV-EPC [40]	c e		c						c e	
	Feature-EPC [40]	c		c							
Others	C-EWF [11] (EWF nets)	c		c							
Others	Liaskon et al. [20] (goal models)		c	c			c				c
Others	Protos2CPN [9] (Petri nets)	c									c
Others	Conf. in SAP WebFlow [10]			c							
Others	PESOA for UML-AD [26]	c e i p		p		p					c p
Others	Razavian & Khosravi [27] (UML-AD)	c		c		c					
Others	PESOA for state machines [26]			p	p						c e i p
Others	C-YAWL [12] (YAWL)			c							
Others	CoSeNet [36] (YAWL)			c	c						
# studies supporting the variability type		14	5	17	5	6	1	2	0	5	7

* Support of: c – configuration, i – inheritance, e – extension, p – parameterization

Variability modeling in the organizational perspective is partially handled in only three studies, where variability of roles or organizational units is handled. The reason for this low support may be that the reviewed studies selected to extend languages that do not focus on the organizational perspective, such as EPC and BPMN, and did not adopt more holistic methods, such as ARIS [34].

As an example for the need to improve the expressiveness in the neglected perspectives, Table 4 lists three examples taken from our case study. In the first two examples the variability is in the organization perspective: the budget can be controlled by a role or by a software system, and received items can be inspected by different roles. This type of variability cannot be handled in existing BPMLs as configurable control flows connect functional elements and not organizational ones and inheritance cannot simply utilized when different elements types (e.g., a role and software in the first example) are involved. In the third example, the variability is in the informational perspective. One can claim that this kind of variability can be specified using parameterization. However, currently business rules need to be associated to the parameter in order to constrain the values it can receive at design-time (and not at run-time).

Neglected Elements. Analyzing the variability in the different perspectives, we observed that not all elements in the same perspective are similarly handled. The degree of support for the different elements is once again important as organizations may face variability in certain elements, e.g., business processes that involve many events and event handlers. In this case, using BPMLs that support variability in the informational perspective will not necessarily help, as those BPML may concentrate on resource variability (and not on events).

Table 4. Examples of variability related to neglected perspectives

Case title	Case description	Organization	Comments
Budget control	Budget can be controlled by the finance department that monitors and alerts on excess expenditures or by an automatic alert software system	Industrial company	Variability in the organizational perspective
Received Items Inspection	Received item inspection can be done by a warehouseman or by any worker qualified by the warehouseman	Industrial company	Variability in the organizational perspective
Delivery date overdue	The system warns on delivery date overdue; the number of "acceptable" overdue days varies, depending on the organization policies	Industrial company, university library	Variability in the informational perspective

As can be seen in Table 3, the most neglected perspective naturally also yields the most neglected elements. However, in the functional and behavioral perspectives, the variability of composite activities, sequence flows and data flows is neglected. We speculate that the reasons for this lack of support are that composite activities are perceived as aids to support scalability and thus their variability is not supported as much as the variability of the building blocks (the atomic tasks); sequence flows are mainly used to connect elements and their variability is hard to be grasped and modeled; and data flows are secondary elements in process models. Furthermore, variability in data flows may be percolated to variability in resources (i.e., in the informational perspective) and vice versa. In the informational perspective, events and resources are similarly neglected. Finally, in the organizational perspective, the variability of software elements is completely neglected, maybe since they are considered in business process modeling as "black boxes".

Table 5 lists three examples related to neglected elements found in our case study: variability in software, sequence flows, and resources.

Table 5. Examples of variability related to neglected elements

Case title	Case description	Organization	Comments
Purchase order generation	A purchase order can be generated automatically by a purchasing module of an ERP system or automatically by an autonomous purchasing system	Industrial company, University library	Variability in software (org. perspective)
Shipment order	Purchase order can be produced before shipment or after shipment (push supply)	University library	Variability in sequence flows (behavioral perspective)
Types of invoices	A supplier invoices can be hard-copy or electronic	Industrial company	Variability in resources (inf. perspective)

The above types of variability can be handled in existing BPMLs by utilizing inheritance, but such a treatment introduces abstract elements to the model – the "super" elements in the inheritance, which complicate the models and may negatively affect comprehension. Furthermore, inheritance of behavioral elements is not well supported in existing BPMLs and sometimes requires splitting models or percolating the variability to connectors.

Neglecting Cross-perspective or Cross-element Variability. Most BPMLs support variability within the same kind of elements. Only a few BPMLs refer to variability that goes beyond the boundaries of a single element type or perspective. These BPMLs commonly define placeholder elements that can be replaced by different elements from the same perspective or from different perspectives. This possibility is mainly utilized for replacing control flows and sequence flows, atomic tasks and composite activities, and atomic tasks and sequence flows. The other combinations are (almost) completely neglected. As an example to the need to represent variability of different elements, which potentially belong to different perspectives, consider a case of inventory assessment. Assessing the inventory may be a complicated function in a certain organization, justifying its representation as a composite activity that includes tasks for counting the actual amounts, writing them down, comparing them to the expected amounts, resolving differences, and so on. The same process may be very simple in an organization which checks its inventory continuously by means of a cycle count, requiring only generation of a printed report. Moreover, inventory assessment in one organization may be an internal function, calling for its representation in the functional perspective. A different organization may use JIT (Just In Time) method in which the supplier manages the inventory and supplies the products whenever they are needed. In this case the supplier is an external entity that triggers events that may cause the activation of different functional units when occur. In these cases it is important whether the BPML supports variability that goes beyond the boundaries of a specific high-level element or a specific perspective.

5.2 Deficiencies with Respect to the Guidance Dimension

Neglected Variability Mechanisms. The nature of the mechanisms makes them suitable to different types of variability. Configuration, for example, is suitable to situations where all the variants are explicitly modeled and the selection of the appropriate variant needs to be guided. Extension and inheritance, on the other hand, enable additional design of the variants. Finally, parameterization requires generalization of the variants and properly using parameters when necessary.

As can be seen in Table 3, variability in business processes is mainly supported in terms of configuration. All the 22 reviewed BPMLs support configuration, which is relatively easy to utilize. Furthermore, in 12 of the languages configuration is the only utilized mechanism. To support configuration, the languages usually supply means for specifying optional elements and selection conditions. Less than half of the reviewed BPMLs support inheritance, while parameterization and extension are far away neglected. A possible reason for this may be that parameterization requires extra generalization effort (done only in the PESOA project) and extension is too lenient and less guided. The case entitled “delivery date overdue” in Table 4 exemplifies the need for parameterization at design-time, while the case entitled “received items inspection” in that table exemplifies the need for extension.

Neglected Granularity-Guidance Combinations. Examining the granularity-guidance combinations, we found that the most commonly used mechanism in all perspectives is configuration, while inheritance is commonly used, in addition to configuration, in the functional and informational perspectives. Extension and parameterization are lowly used in the functional, behavioral, and informational perspectives. These findings may be attributed to the nature of the mechanisms: inheritance of tasks and resources is known in other modeling areas, such as object-oriented modeling; extension is found with respect to functionality (e.g., extension points in use case diagrams); and parameterization is mainly known with respect to data and information. We found evidence to the need of the different variability mechanisms in the various perspectives. The case entitled “purchase order generation” in Table 5, for instance, exemplifies the need for inheritance in the organizational perspective.

6 Conclusions and Future Directions

Analyzing variability of business processes is important within an organization and between similar organizations. The main way to present the outcome of such analysis is through variability models which can be incorporated into or presented orthogonally to the business process models. We examined the expressiveness of different BPMLs that support design-time variability modeling with respect to two dimensions: *granularity*, which refers to four perspectives and their elements, and *guidance*, including four variability mechanisms. We found that variability in the functional and behavioral perspectives is extensively handled in the modeling level, although variability of composite activities, sequence flows and data flows is quite neglected. Variability in the informational perspective is supported to some extent, leaving aside important elements, such as events. Variability modeling in the organizational perspective is far away neglected. We further found that configuration is the most utilized variability mechanism in business processes, but some languages support extension, inheritance, and parameterization for creating process variants mainly in the functional and information perspectives. All languages concentrate on variability within the same element kinds, neglecting possible variability between different types of elements that may belong to the same or different perspectives.

It is important to consider the current study under the following limitations. First, we reviewed modeling languages in the field of business processes. We could extend the scope of review to studies that deal with variability in databases and organizations. This way we could increase the expressiveness in the informational and organizational perspectives. However, incorporating such languages into BPMLs is not trivial and may increase complexity (potentially decreasing comprehension). Second, most modeling languages reviewed in the current study are workflow-oriented. This is because most "regular" BPMLs are workflow-oriented [19]. However, business process modeling approaches that capture and refine business goals also exist. The study in [20], which is included in our review, is a goal-oriented language that explicitly refers to design-time variability in business processes. Third, we included in our study only graphical languages that extend existing BPML. In particular, textual and formal languages as well as proprietary languages were not included.

In the future, we plan to provide concrete suggestions for improving the expressiveness of variability modeling in BPMLs. In particular, we will provide suggestions for supporting neglected perspectives, elements, mechanisms, and combinations. We further plan to empirically evaluate the influence of these suggestions on the usability of different BPML extensions for variability modeling and their comprehensibility. Finally, we intend to explore additional dimensions and refine the current dimensions, e.g., by examining additional variability mechanisms and referring to low-level process elements.

References

1. Andersson, B., Bider, I., Johannesson, P., Perjons, E.: Towards a formal definition of goal-oriented business process patterns. Business Process Management Journal 11(6), 650–662 (2005)
2. Ayora, C., Torres, V., Pelechano, V.: Variability management in business process models. Technical Report 17, PROS – UPV (2012)
3. Bayer, J., Flege, O., Gacek, C.: Creating product line architectures. In: van der Linden, F.J. (ed.) IW-SAPF 2000. LNCS, vol. 1951, pp. 210–216. Springer, Heidelberg (2000)
4. Becker, J., Rosemann, M., von Uthmann, C.: Guidelines of Business Process Modeling. In: van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.) Business Process Management. LNCS, vol. 1806, pp. 30–49. Springer, Heidelberg (2000)

5. Brocke, J.: Design Principles for Reference Modelling: Reusing Information Models by Means of Aggregation, Specialisation, Instantiation, and Analogy. Reference Modeling for Business Systems Analysis, 47–75 (2007)
6. Curtis, B., Kellner, M., Over, J.: Process Modeling. Communication of the ACM 35(9), 75–90 (1992)
7. Dijkman, R., La Rosa, M., Reijers, H.: Managing large collections of business process models — current techniques and challenges. Computers in Industry 63(2), 91–97 (2012)
8. Dumas, M., Van der Aalst, W.M.P., Ter Hofstede, A.H.M.: Process-Aware Information Systems: Bridging People and Software through Process Technology. Wiley & Sons (2005)
9. Gottschalk, F., van der Aalst, W.M.P., Verbeek, H.M.W.: Protos2cpn: Using colored Petri Nets for configuring and testing business processes. International Journal on Software Tools for Technology Transfer 10(1), 95–110 (2007)
10. Gottschalk, F., van der Aalst, W.M.P., Jansen-Vullers, M.H.: SAP WebFlow Made Configurable: Unifying Workflow Templates into a Configurable Model. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 262–270. Springer, Heidelberg (2007)
11. Gottschalk, F., Van Der Aalst, W.M.P., Jansen-Vullers, M.H., Rosa, M.L.: Configurable workflow models. International Journal on Cooperative Information Systems 17(2), 177–221 (2008)
12. Gottschalk, F., Wagemakers, T.A.C., Jansen-Vullers, M.H., van der Aalst, W.M.P., La Rosa, M.: Configurable process models: Experiences from a municipality case study. In: van Eck, P., Gordijn, J., Wieringa, R. (eds.) CAiSE 2009. LNCS, vol. 5565, pp. 486–500. Springer, Heidelberg (2009)
13. Hallerbach, A., Bauer, T., Reichert, M.: Capturing variability in business process models: The Provop approach. Journal of Software Maintenance and Evolution: Research and Practice 22(6/7), 519–546 (2010)
14. Istoan, P., Klein, J., Perouin, G., Jézéquel., J.M.: A metamodel-based classification of variability modeling approaches. In: VARY International Workshop Affiliated with MODELS 2011, pp. 23–32 (2011)
15. Korherr, B.: Business Process Modelling Languages, Goals, and Variabilities. PhD Thesis, Vienna University of Technology (2008)
16. Korherr, B., List, B.: Extending the EPC and the BPMN with Business Process Goals and Performance Measures. In: The 9th International Conference on Enterprise Information Systems (ICEIS 2007), pp. 287–294. ACM/SIGMIS and AAAI (2007)
17. La Rosa, M., Dumas, M., ter Hofstede, A.: Modelling Business Process Variability for Design-Time Configuration. In: Cardoso, J., van der Aalst, W.M.P. (eds.) Handbook of Research on Business Process Modeling. Information Science Reference, pp. 204–228 (2008)
18. La Rosa, M., Dumas, M., ter Hofstede, A., Mendling, J.: Configurable Multi-Perspective Business Process Models. Information Systems 36(2), 313–340 (2011)
19. Lapouchnian, A., Yu, Y., Mylopoulos, J.: Requirements-driven design and configuration management of business processes. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 246–261. Springer, Heidelberg (2007)
20. Liaskon, S., Lapouchnian, A., Yu, Y., Yu, E., Mylopoulos, J.: On Goal-based Variability Acquisition and Analysis. In: The 14th IEEE International Conference Requirements Engineering (RE 2006), pp. 92–96 (2006)

21. List, B., Korherr, B.: An Evaluation of Conceptual Business Process Modelling Languages. In: The 21st ACM Symposium on Applied Computing (SAC 2006), pp. 1532–1539 (2006)
22. Lu, R., Sadiq, W.: A survey of comparative business process modeling approaches. In: Abramowicz, W. (ed.) BIS 2007. LNCS, vol. 4439, pp. 82–94. Springer, Heidelberg (2007)
23. Park, J.: A Modeling Approach for Business Processes Based on Variability. In: 9th International Conference on Software Engineering Research, Management and Applications (SERA), pp. 211–218 (2011)
24. Patiniotakis, I., Papageorgiou, N., Verginadis, Y., Apostolou, D., Mentzas, G.: An Aspect Oriented Approach for Implementing Situational Driven Adaptation of BPMN2.0 Workflows. In: La Rosa, M., Soffer, P. (eds.) BPM Workshops 2012. LNBIP, vol. 132, pp. 414–425. Springer, Heidelberg (2013)
25. Pichler, P., Weber, B., Zugal, S., Pinggera, J., Mendling, J., Reijers, H.A.: Imperative versus Declarative Process Modeling Languages: An Empirical Investigation. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM Workshops 2011, Part I. LNBIP, vol. 99, pp. 383–394. Springer, Heidelberg (2012)
26. Puhlmann, F., Schnieders, A., Weiland, J., Weske, M.: Variability Mechanisms for Process Models. PESOA-Report TR 17/2005, Process Family Engineering in Service-Oriented Applications, PESOA (2005)
27. Razavian, M., Khosravi, R.: Modeling Variability in Business Process Models using UML. In: 5th International Conference on Information Technology: New Generations (ITNG 2008), pp. 82–87 (2008)
28. Recker, J., Rosemann, M., van der Aalst, W.M.P., Mendling, J.: On the Syntax of Reference Model Configuration. In: Bussler, C.J., Haller, A. (eds.) BPM 2005. LNCS, vol. 3812, pp. 497–511. Springer, Heidelberg (2006)
29. Reinhartz-Berger, I., Soffer, P., Sturm, A.: Organizational reference models: Supporting an adequate design of local business processes. International Journal of Business Process Integration and Management 4(2), 134–149 (2009)
30. Reinhartz-Berger, I., Soffer, P., Sturm, A.: Extending the adaptability of reference models. IEEE Transactions on Systems, Man, and Cybernetics, Part A 40(5), 1045–1056 (2010)
31. Rosemann, M., van der Aalst, W.: A Configurable Reference Modelling Language. Information Systems 32(1), 1–23 (2007)
32. Russell, N., ter Hofstede, A.H.M., Edmond, D.: Workflow Data Patterns. QUT Technical report, FIT-TR-2004-01, Queensland University of Technology, Brisbane (2004), http://is.tm.tue.nl/research/patterns/documentation.htm
33. Rychkova, I., Nurcan, S.: Towards Adaptability and Control for Knowledge-Intensive Business Processes: Declarative Configurable Process Specifications. In: The 44th Annual Hawaii International Conference on System Sciences, pp. 1–10 (2011)
34. Scheer, A.W.: ARIS – Business Process Modeling, 3rd edn. Springer (2000)
35. Schnieders, A., Puhlmann, F.: Variability Mechanisms in E-Business Process Families. In: The 9th International Conference on Business Information Systems (BIS 2006), pp. 583–601 (2006)
36. Schunselaar, D.M.M., Verbeek, E., van der Aalst, W.M.P., Raijers, H.A.: Creating Sound and Reversible Configurable Process Models Using CoSeNets. In: Abramowicz, W., Kriksciuniene, D., Sakalauskas, V. (eds.) BIS 2012. LNBIP, vol. 117, pp. 24–35. Springer, Heidelberg (2012)

37. Shahzad, K., Johannesson, P.: An Evaluation of Process Warehousing Approaches for Business Process Analysis. In: The 5th ACM-EOMAS in Conjunction with CAiSE 2009, pp. 1–14 (2009)
38. Torres, V., Zugal, S., Weber, B., Reichert, M., Ayora, C., Pelechano, V.: A qualitative comparison of approaches supporting business process variability. In: La Rosa, M., Soffer, P. (eds.) BPM Workshops 2012. LNBIP, vol. 132, pp. 560–572. Springer, Heidelberg (2013)
39. Van Gurp, J., Bosch, J., Svahnberg, M.: On the notion of variability in software product lines. In: Proceedings of the Working IEEE/IFIP Conference on Software Architecture (WICSA 2001), pp. 45–54 (2001)
40. Vervuurt, M.: Modeling Business Process Variability. Master Thesis. University of Twent (2007)
41. Weidmann, M., Koetter, F., Kintz, M., Schleicher, D., Mietzner, R.: Adaptive Business Process Modeling in the Internet of Services (ABIS). In: ICIW 2011, pp. 29–34 (2011)