

Testing a Selection of BPMN Tools for Their Support of Modelling Guidelines

♦M. Snoeck, ♦I. Moreno-Montes de Oca, ♦T. Haegemans, ♦B. Scheldeman, ♦T. Hoste

♦Research Center for Management Informatics, KU Leuven, Naamsestraat 69, 3000 Leuven, Belgium

♦Department of Computer Science, UCLV, Santa Clara, Cuba
monique.snoeck@kuleuven.be; lssel@uclv.edu.cu; tom.haegemans@kuleuven.be;
bjornscheldeman@gmail.com; hoste.tom.92@gmail.com

This is the camera ready version of this paper. The published version is available at http://link.springer.com/chapter/10.1007/978-3-319-25897-3_8

Provided for personal and non-commercial use only.

Please cite as follows:

Snoeck M, Moreno-Montes de Oca I, Haegemans T, Scheldeman B, Hoste T, Testing a selection of BPMN tools for their support of modelling guidelines, 2015, Springer LNBIP series, vol. 235, pp. 111 - 125, IFIP WG 8.1 working conference on the Practice of Enterprise Modelling (POEM 2015) (Valencia (Spain))

Testing a Selection of BPMN Tools for Their Support of Modelling Guidelines

*M. Snoeck, *I. Moreno-Montes de Oca, *T. Haegemans, *B. Scheldeman, *T. Hoste

*Research Center for Management Informatics, KU Leuven, Naamssestraat 69, 3000 Leuven, Belgium

*Department of Computer Science, UCLV, Santa Clara, Cuba
monique.snoeck@kuleuven.be; lssel@uclv.edu.cu; tom.haegemans@kuleuven.be;
bjornscheldeman@gmail.com; hoste.tom.92@gmail.com

Abstract. BPMN has become the de facto standard notation for process modelling. Past research has demonstrated the need for modelling guidelines to improve the quality of process models. In previous research we collected a set of practical guidelines through a systematic literature survey and classified those in different categories. In this paper we test a selection of BPMN tools for their support for these guidelines, and report on existing support per category of guideline and the kinds of support used by the tool to support the different guidelines. The results give insight into which domains of guidelines are well supported and which lack support from BPMN tools. Further, different preferences of the vendors are observed regarding the methods of support they implement in their tools.

1 Introduction

The Business Process Modelling Notation (BPMN) provides a way to conceptualize business processes into a graphical representation using constructs such as events, gateways, activities, processes, sub-processes, and control-flow dependencies [1, 2]. BPMN is a means to document, model and analyse the increasingly changing business processes in a structured, logically and systematic way [3]. The growing interest in BPMN turned it into an important standard with regards to process modelling within organizations and across organizations [4, 5].

One of the main objectives of BPMN is to provide an easily understandable notation. Because there is an increasing amount of stakeholders (e.g. business users, business analysts, technical developers) using these models, it is important that other people can easily interpret the BPMN models [6]. For people who were not involved in the modelling process, understanding complex and low quality models can be problematic. Also, misinterpretations of models can be the cause of a wrong implementation. A lot of the BPMN modelling is done as part of requirements engineering in the early phase of system development. When a model with a significant amount of errors

is used as a basis for implementation, this generally means a high modification cost and a lot of effort to correct these mistakes at a later point in time [1].

To avoid misinterpretations and high correction costs and get the full potential out of BPMN we have to make sure the quality of the BPMN models is high and the models do not contain purposeless complexity. The problem with providing a certain level of quality is that there are no unified standards regarding BPMN model quality. So the question that arises here is: How can we ensure the quality of BPMN models? Since relying on best practices has a positive impact on the model quality [7], following guidelines provided in literature can significantly enhance the quality of BPMN models.

However, two complications arise with regard to applying the guidelines. First, there is a vast amount of guidelines published in a variety of sources, making it difficult for a modeller to study and apply them all. This problem has been addressed in previous research by the authors. A systematic literature review (SLR) on pragmatic guidelines for business process modelling quality spanning the period of 2000-2014 [8] collected 72 papers addressing different aspects of modelling quality. All the guidelines and best practices identified in these papers were extracted, classified into different categories, and for each set of similar guidelines a unified guideline was proposed. This resulted in a set of 27 problems and unified guidelines, documented in a technical report [9]. Second, the fact that guidelines exist does not automatically mean all of these guidelines are always applied by every modeller. However, practically every user of BPMN uses the language in combination with a BPMN tool [3], and it was already proven that people using these tools and receiving support from these tools are less likely to produce low quality BPMN models [3]. This means that it is possible to provide BPMN models of decent quality, but most likely only when the modeller gets some kind of support provided by a modelling tool. This raises the following question: to what extent are the unified guidelines supported by BPMN modelling tools and how do they do this?

The remainder of this paper is structured as follows: Section 2 introduces the research questions; Section 3 presents related work; Section 4 introduces the approach for testing the tool support for a given collection of guidelines, while Section 5 reports on findings; Section 6 presents a discussion of the results. Finally, Section 7 presents conclusions and future work.

2 Research Questions

In this paper, we investigate which business process modelling guidelines are supported by current BPMN tools. In particular, we answer the following main question:

RQ: How extensive is the support for the identified set of unified guidelines in a representative set of popular BPMN tools?

The extent of the support can be assessed in two different ways: by looking at the number of guidelines that are supported, and by looking at the kind of support that is given (like e.g. supporting better layout through warnings versus automatic position-

ing of objects). Researching these different types of support will help answering our main research question. Hence the first sub-question is formulated as:

SRQ1: Which types of support exist for best practices in the considered set of tools?

The second way to evaluate the extent of the support is to look at which guidelines have some sort of support or not. So, the second research sub-question is formulated as follows:

SRQ2: To what extent are the guidelines supported by the considered set of tools?

The guidelines that were tested for support, have been classified into different categories [9]. By analysing the support per category, we gain useful insights regarding implementation problems or shortage of support in certain domain. Therefore, the third sub-question is:

SRQ3: To what extent is each category of best practices supported by the considered set of tools?

3 Related Research

While, to the best of our knowledge, BPMN tool support for guidelines, has not yet been researched, there are many other studies that provide a framework to evaluate a tool against certain criteria (e.g. [10-15]). The most relevant paper for our research [10] proposes an approach for the selection of a modelling tool and uses best practices as its evaluation criteria. First, the author develops a methodology to evaluate the candidate tools. The criteria for evaluation are build based on the definition of a core benchmark set defined in terms of a general benchmark goal. Moreover, the methodology requires the selection of a preferential tool, which reflects the company's perception of a "best-in-class" modelling tool [10]. Second, the researcher constructs metrics that indicate the importance that a company assigns to each criterion in the benchmark set. Third, the study provides an example by applying the framework to a set of 16 tools.

The main difference with the approaches of [10-15], is that we aim at comparing BPMN tools without adopting the perspective of a particular organisation. Consequently, we will use the framework of [10] as a starting point, but, to avoid bias, we will adhere to a more objective methodology and use context-independent measures. The guidelines identified through the SLR will serve as the core benchmark set.

4 Methodology

4.1 Selection of the Set of Guidelines

As explained above, in previous work, the authors of this paper collected the available guidelines from the scientific literature through a systematic literature review [8] and made an overview available as a technical report [9]. This technical report provides extensive information per guideline such as its sources, existing variants, metrics, examples and available scientific evidence. The technical report identifies 27

unified guidelines, classified into three large categories: guidelines that count elements, morphology guidelines and presentation guidelines, which each have a number of subcategories. For example, presentation is further subdivided in layout and label guidelines. For this paper, tool support for all guidelines was investigated, irrespective of available scientific evidence for their impact on modelling quality. This is motivated by the fact that future research may still prove certain guidelines to be useful, even though experimental evaluation of the impact on model quality may currently be lacking.

4.2 Selection of the Set of Tools

A representative set of tools was selected in 4 steps. The first step constituted of defining inclusion/exclusion criteria. Due to financial resource limitations, only free tools or tools with a free trial version were considered. To ensure a fair evaluation, only full functional tools with a recent update (2013 or later) were selected. Tools that seem to be outdated or only provide a limited functionality during the trial period should be excluded to avoid wrong conclusions or distorted results. Finally, to prevent checking non-relevant tools (e.g. purely graphical tools such as yEd or Microsoft Visio), only modelling tools, or suites with a modelling component were considered. In step 2, we tried to compose a set of available tools that is as complete as possible: we consolidated market overviews and existing lists [16, 17] into a list of 117 tools, which we reduced to a list of 20 tools by applying the exclusion and inclusion criteria (step 3). In step 4, due to time restrictions, six tools were selected while ensuring the sample to be as representative as possible. First, Signavio Process Editor was selected for being known for its support of best practices: a complete overview of the numerous guidelines that are incorporated in the tool is available on their website [18]. Next, two tools stand out because of their high degree of attention for best practices as evidenced by the availability of extensive documentation about best practices on the website of the vendors: Bizagi Process Modeller and Camunda Modeller [19, 20]. The fourth included modelling tool is Bonita [21] because it was ranked first among several open-source applications for modelling and publishing BPMN 2.0 processes [6]. Finally, the selection was expanded with two popular tools [22]: Visual Paradigm and ARIS Express.

4.3 Research Method per Subquestion

First, to identify the types of support, we followed an explorative approach, similar to grounded theory. To start, we explored the tools to get familiar with them and to discover the functionalities they offer (e.g. automatic formatting of the layout of the model, a quality check to validate the model...). Next, BPMN test models were drawn, without observing the guidelines. During this phase we identified the different ways of how the tool tries to support the user.

Second, to examine the extent to which a guideline is supported by the selected tools and to investigate to what extent each category of best practices is supported, the guidelines were tested by drawing diagrams with the tools. While each unified guide-

line corresponds to one problem, some guidelines could not be matched to a single test. As an example, unified guideline 5, dealing with the problem of multiple start and/or end events, has several sub-guidelines: 1) use no more than two start/end events in the top process level; 2) use one start event in subprocesses and 3) use two end events to distinguish success and fail states in subprocesses. This guideline was therefore split into atomic guidelines such as to ensure full test coverage (see items b-e for subcategory 1.2 Number of events in Table 1). Overall this resulted in 56 atomic guidelines to test.

For each atomic guideline, we constructed a BPMN test model that mimics its corresponding problem. For example, the guideline “use no more than two start/end events in the top process level” was tested by constructing a BPMN model that has three start events (see [9 p. 10]). Thus, each atomic guideline was assessed by one test and each test corresponds to exactly one atomic guideline. To reduce the possibility of mistakes, each test was performed two times per guideline, each time by a different researcher. In addition, whenever possible, the test models were exported from one tool to another by means of XML to avoid errors caused by manually redrawing the test models.

In total 56 models that evidence the violation of the unified guidelines were tested. In addition, each guideline was tested by two researchers in each of the six tools. This means 672 test were conducted.

5 Results

5.1 Identification of Types of Support

From the exploratory study it was possible to detect a certain consistency regarding the way the different tools provide support. In general, the same five different ways of support are used across the different tools:

Forced Support. In this case, the tool forces the user to follow the corresponding guideline, resulting in the inability to avoid the guideline (e.g. explicitly labelling the model as erroneous, denying the user to drag an element to a certain place, making it unable to save an invalid model, refusing unlabelled elements...).

Warning. In case of a warning, the tool gives the user a warning message after validating the model (e.g. when a user draws a model that is too large according to the tools' guidelines, when pressing the validate or save button, a message will be shown saying that the model should be smaller). This does however not restrict the user in any way to continue working on or save the model.

Suggestive Support. A suggestion means that the tool tries to direct the user in the right way (e.g. automatically suggesting a certain size for an element, drawing guides to position the element in a symmetric way...).

Documentation. The documentation of the tools contains some best practices referring to modelling guidelines which can help to enhance the quality of the model. This type of support can be combined with another type of support (e.g. a documented guideline can also be enforced by the tool). However, most of the guidelines stated in

the documentation are not supported by the tool itself, resulting in only written guidance for the modeller.

Related Support. Sometimes it happens that a tool does not directly support a particular guideline but does this indirectly by supporting a related guideline. For example, a tool may not provide direct support for the guideline "Avoid models with more than 7 events." but may issue a warning concerning the actual size of the model. This is obviously related to each other, and therefore classified as related support.

No Support. When a tool provides no support at all, this means there is no forced support, no warning, no suggestions and nothing within the provided documentation concerning a certain guideline.

5.2 Support for Individual Guidelines

The test results are summarized in Table 1, which gives an overview of which types of support the different tools have to offer to every tested guideline. For some tools the test resulted in more than one possible type of support. In this case the strongest support was reported as result of the test. The ordering between types of support was defined as follows: forced support is stronger than a warning; a warning is 'explicit' and therefore stronger than a suggestion; since a tool actively intervenes when suggesting something, a suggestion is stronger than documentation; and the documentation of the actual guideline is stronger than some indirect support. So, when both a warning and a suggestion support a certain guideline, only the warning is reported as result in Table 1. As an example, Signavio supports the guideline, "Place elements as symmetric as possible", in two different ways. First, the tool offers a suggestion by snapping elements automatically to a symmetric position relative to the model, which is classified as a suggestion. Second, the tool gives the user a layout warning when lines are not orthogonally drawn, which is classified as a warning. Because the warning is a stronger type of support than the suggestion, the type of support was classified as a warning in Table 1. The weaker type of support is not visible in Table 1, but is still archived in the complete overview in Appendix D of [23]. Overall, on the total of 336 combinations of atomic guideline and tool, 118 (35,1%) cases revealed some type of support. Fig. 1 gives an overview of the overall degree of support for the six tools.

5.3 Support per Category

A "degree of support" was calculated for every tool per category of guidelines. This degree of support indicates what proportion of the tests resulted in a positive value. As an example, if 9 out of the 15 tests in the category of guidelines that count elements return a positive result (meaning there is some kind of support), this results in a score of 60% (= 9/15). Fig. 2 gives an overview of the degree of support per category for the six tools. Similarly, Fig. 3 gives an overview of the strongest types of support per category for the 118 cases where a tool offers some sort of support for a guidelines.

Table 1: Tool support per guideline

LEGEND:		RESULT					
f	The tool forbids the user to avoid the guideline.	B	V			C	
w	The tool gives the user a warning when validating the model.	I	I			A	
s	The tool suggests following the guideline when the user is drawing the model.	Z	S			M	
d	The documentation of the tool supports the guideline. (can be combined with other symbol)	A	U		A	U	
n	The tool does nothing related to the guideline.	A	A		R	N	
r(n)	The tool does something related to another guideline. The number n refers to the guideline listed below.	G	L		I	D	
		I	S		S	A	
		S	P				
		I	M		E	M	
		G	O		R	B	
		N	D		A	O	
		A	E		D	N	
		V	L		I	E	
		I	E		G	T	
		O	R		M	A	
					S	R	
GUIDELINE	TEST: Does the tool avoid ...						
1 Guidelines that count elements							
1.1 Number of elements	a. A high number of elements	r(1)	n	n	n	n	
	b. Duplicate elements	n	d	n	w	n	
	c. Unnecessary elements	w	d	n	n	n	
1.2 Number of events	a. A high number of events	n	n	n	n	n	
	b. A high number of start events	w	n	n	n	n	
	c. A high number of end events	n	n	n	n	n	
	d. Absence of start events	w	d	n	w	n	
	e. Absence of end events	w	d	n	w	n	
	f. High number of intermediate events	n	n	n	n	n	
1.3 Number of arcs	a. A high number of arcs	n	n	n	n	n	
1.4 Number of gateways	a. A high number of gateways	n	n	n	n	n	
1.5 Number of activities	a. A high number of activities	r(1)	n	n	n	n	
1.6 Element complexity	a. A high number of routing paths per gateways	n	n	n	n	n	
	b. Multiple inputs and outputs on the same gateway	w	d	n	n	n	
	c. A high number of outgoing sequence flows from an event	f	n	n	n	n	
2 Morphology							
2.1 Depth (nesting)	a. A high level of depth	n	n	r(2)	n	r(3)	
2.2 Cyclicity	a. Cycles in the model	n	n	n	n	n	
	b. Badly formed cycles	w	n	n	n	n	
	c. Multiple exit points per cycle	n	n	n	n	n	
2.3 Parallelism	a. A high level of parallelism	n	n	r(2)	n	r(4)	
	b. Bad parallelism in the model	w	n	n	n	n	
2.4 Structuredness	a. Unstructuredness (AND-Gate/xOR-Gate mismatch)	w	d	n	n	n	
	b. Unstructuredness (xOR-Gate/AND-Gate mismatch)	w	d	n	n	n	
	c. Unstructuredness (AND-Gate/iOR-Gate mismatch)	n	d	n	n	n	
	d. Unstructuredness (iOR-Gate/AND-Gate mismatch)	n	d	n	n	n	
	e. Unstructuredness (xOR-Gate/iOR-Gate mismatch)	n	d	n	n	n	
	f. Unstructuredness (iOR-Gate/xOR-Gate mismatch)	n	d	n	n	n	
2.5 Diameter (longest path)	a. A large model in terms of diameter	w	n	n	n	s	
2.6 iOR Gateways	a. A high level of gateway diversity	s	n	n	n	n	
	b. The use of iOR-Gates	s	n	n	n	n	
2.7 General complexity	a. A high level of complexity	s	n	r(2)	n	r(3,4)	
2.8 Modularity	a. A too large model	r(1)	d	n	n	n	

Table 1 (ctd): Tool support per guideline

3 Presentation									
3.1 Layout guidelines	a. A high number of crossing lines	n	d	s	n	s	d		
	b. Overlapping lines	w	s	n	n	n	d		
	c. Overlapping elements	w	n	s	f	s	f		
	d. A high number of bends in connecting elements.	w	f	s	n	s	n		
	e. Objects not drawn orthogonally / symmetric	w	s	s	s	s	s		
	f. A square model (and supports a long and thin model)	n	n	s	n	n	d		
	g. Elements drawn with different sizes	n	n	n	n	n	d		
	h. A large drawing area	w	n	n	n	n	n		
	i. The use of color highlights (not supporting the guideline for colors)	s	s	s	f	s	f		
	j. Absence of color highlights (supporting the guideline for colors)	r(8)	r(8)	r(8)	n	r(8)	n		
	k. Gateways used for both splitting and merging	w	d	n	n	n	d		
	l. Elements placed too close together (not overlapping)	w	n	n	f	s	n		
	m. Related elements placed too far from each other	n	n	s	n	s	n		
	n. Unspecified task types	n	n	n	n	s	n		
	o. Absence of pools and swimlanes	n	s	n	s	n	n		
	p. A model flow not from left to right	w	d	n	n	s	n		
3.2 Label style	a. Labels that do not follow verb-object style	n	d	r(5)	n	n	d r(7)		
	b. Long labels	n	n	n	w	n	n		
	c. Gateways without labels	w	d	s	n	n	d		
	d. Events without labels	w	d r(6)	s	n	n	d		
	e. Tasks without labels	w	s	s	w	s	w		
	f. Unlabeled outgoing sequence flows from x-OR gateways	w	d	n	n	n	d		
	g. Pools (processes) without labels	w	s	s	f	s	n		
	h. Short labels or abbreviations	n	d	n	n	n	n		
Related Guidelines									
Nr									
1.	SIGNAVIO gives a Layout warning: The diagram is too large in size.								
2.	When VISUAL PARADIGM automatically defines the layout, a less complex view can be achieved, but not always.								
3.	When ARIS EXPRESS automatically defines the layout, depth is displayed in a clear and organized way.								
4.	When ARIS EXPRESS automatically defines the layout, parallelism is displayed in a clear and organized way.								
5.	VISUAL PARADIGM gives a 'Fair' warning when the label does not contain glossary terms.								
6.	BIZAGI Best practices (documentation) suggest not to label start and end events; only other events.								
7.	CAMUNDA Best practices (documentation) suggest 'verb + object' for task labels, but events should be labeled 'object + past'.								
8.	The tool allows you to change the color of elements, but it is suggested to use the same color.								
9.	CAMUNDA Best practices (documentation) suggest only for xOR-Gateways to label outgoing paths with a question.								

6 Discussion

6.1 Support of the Guidelines

Overall, the large majority of guidelines (85,71%) seems to be known by tool builders. Nevertheless there are major differences between tools (see Fig. 1). That is, some of the selected tools offer substantially more support than the others. This was already noticed during the exploratory research phase and was later confirmed in the results. Signavio was found to be the tool with the highest degree of overall support (57,14%), followed by Bizagi (50%), ARIS Express (30,36%), Visual Paradigm (28,57%), Camunda (25%) and Bonita (19,64%).

Although Signavio offers the highest level of support, it only supports 57,14% of the tested guidelines, which is relatively low. Likewise, the support that the other tools under investigation offer for the tested guidelines is less than or equal to 50%. Hence, when considering the tools separately, it can be seen that individual tools clearly lack support for a significant part of the guidelines.

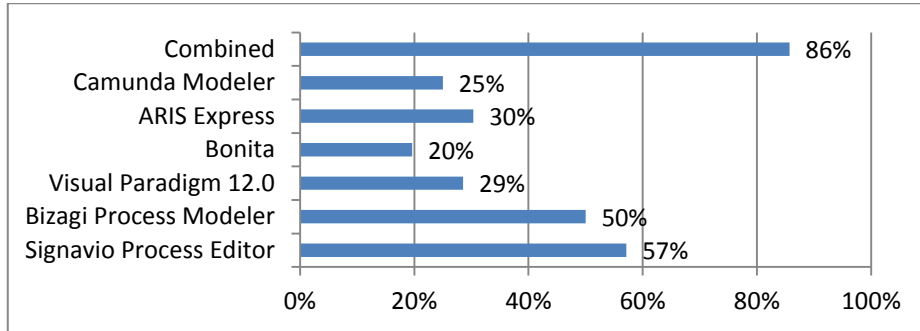


Fig. 1. Extent to which the guidelines are implemented by the set of tools

6.2 Support of the Categories of Guidelines

In general, the group of presentation guidelines is the best supported category of guidelines (see Fig. 2). However, when looking at the details in Table 1, we see large differences in how this support happens concretely. Visual Paradigm and ARIS Express have the highest degree of support for the guidelines regarding the layout of the model. Furthermore, Bizagi scored best regarding label style guidelines, despite having lower degrees of support for the other domains. Also, Camunda Modeller has a strong support for the label style guidelines. Finally, Bonita seems the tool with the weakest average support although certain individual categories score fairly. The different scores per category and subcategory in Fig. 2 indicate that tools vendors each have different preferences about which categories to support.

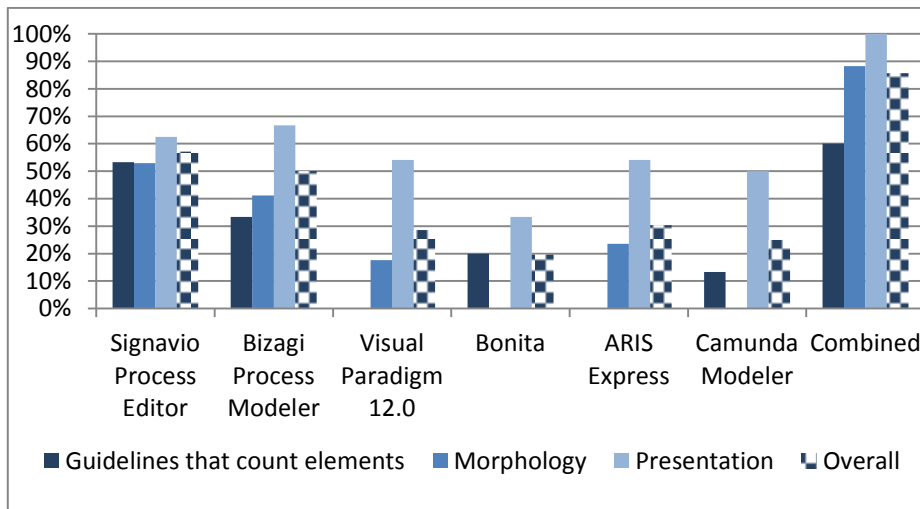


Fig. 2. Degree of support per category of guidelines

In a specific category of guidelines, the overall degree of support is always higher than the degree of support offered by the tool with the highest degree of support. For example, in Fig 2, it can be seen that the degree of support for guidelines that count elements is 60% for the combined set of tools, while, in the same category of guidelines, Signavio offers a degree of support of 53%, which is the highest score of a single tool within that category. This indicates that each single tool supports some guidelines that other vendors do not yet support. Thus, the different vendors of the tools can learn from each other regarding the different support offered by guidelines they have not yet implemented.

Surprisingly, the degree of support for guidelines that count elements is fairly low. Only 60% of the tested guidelines had a positive result in at least one tool (see Fig. 2) and, except for Signavio, all the tools have a degree of support lower than 35%. This low support is unexpected, because guidelines that count elements is the domain which should be the easiest to support (these guidelines have the easiest associated metrics [9]). Moreover, many of these guidelines have empirical support for their effect on model understandability [24, 25], which is a strong motive for providing tool support. A first explanation could be that tool builders fear that support in this area would restrict the user too much, e.g. when more complex models are needed, it would be irritating for the user to repeatedly receive warnings and suggestions in a situation where the high number of elements or arcs are unavoidable. A second possible explanation is that the perceived utility of automated support is perceived as low compared to e.g. the perceived utility of layout support: counting elements is fairly easy to perform manually without support, whereas the use of guides and snapping elements to a grid to optimize the layout of a diagram is considered as useful.

6.3 Types of Support

Overall, with only a use of 7% in the supporting tests, the forced approach is by far the lowest of all (Fig. 3). This comes as no surprise since a high number of error messages may be experienced as user-unfriendly. Forced support is only used for guidelines that count elements and for presentation.

Warnings appear in 26% of the cases. This type of support is convenient for the user as (s)he will not experience any limitations while still actively receiving feedback about model quality. Warnings are mostly used for guidelines that count elements, presumably because of the ease of implementation of the associated metrics.

Suggestions is the most frequently used type of support: it is used in 31% of the cases. While suggestions are less intrusive than warnings and error messages, they can be an effective way of providing support in a user friendly way. From Fig 3, it can be observed that this type of support is mainly used for presentation guidelines. In particular, Fig. 4 shows that when there is support for layout, in half of the cases (50%), this is done by means of suggestions (e.g. snapping an element to a grid, use standard formats for elements, use standard distances between lines...). For label style guidelines, suggestions only account for 30%.

Related support is the second least used type of support: they are used in 12% of the cases. In addition, we find this type of support in 30% of the cases where there is

support for morphology. A possible reason for this is that guidelines provided in this category are sometimes difficult to support in a straightforward way (e.g. "Keep the path from a start node to the end as short as possible").

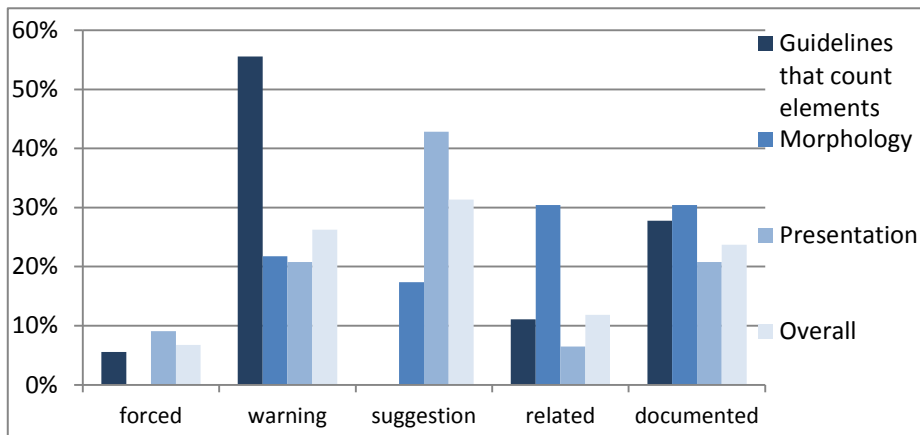


Fig. 3. Strongest types of support per category of guideline and overall

Documentation appeared as a type of support in 24% of the cases. In the category of presentation guidelines, documentation accounts for 33% of the type of support for label guidelines (see Fig. 4). Furthermore, documentation is the most used type of support for morphology guidelines, next to and presumably for the same reason as for the related type of support. Nevertheless, we need to mention that in some of the cases where a stronger type of support is offered, the guideline is documented as well. As an example, some guidelines that Signavio supported were also documented on their website. As explained in section 5.2, Table 1 only reports on the strongest type of support and, therefore, the additional support by means of documentation is not visible in the figures.

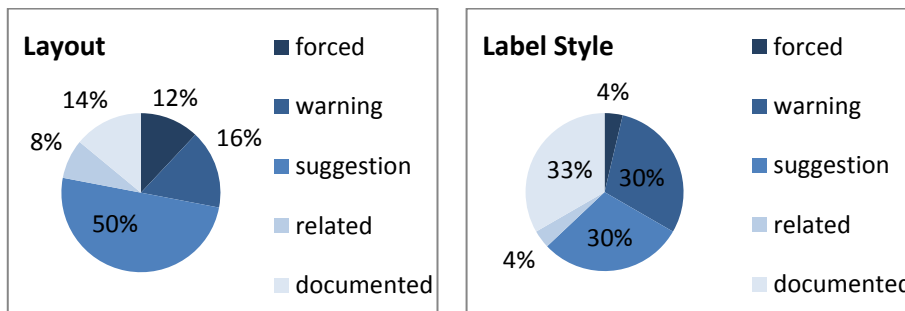


Fig. 4. Types of support per tool for layout and label style guidelines

When looking at individual tools, we can witness significant differences in types of support used by each of the tools (see Fig. 5). For presentation guidelines in particular, we can see that different tools handle support for layout and label guidelines in a very different way (see Fig. 6 and Fig. 7). While Signavio and Bizagi both have a high degree of overall support, Signavio mainly achieves this through warnings while Bizagi does this through the weaker form of support of documentation.

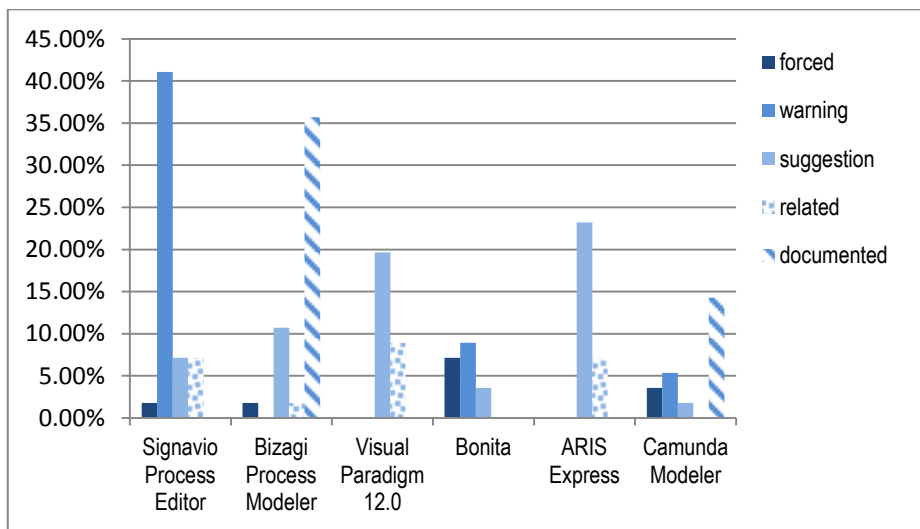


Fig. 5. Types of support per tool.

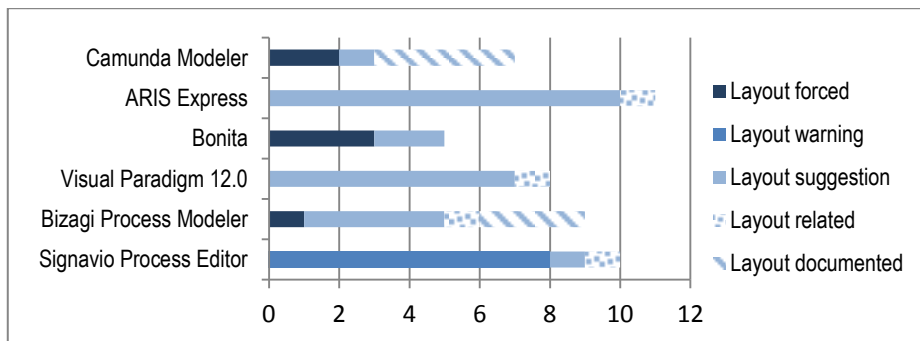


Fig. 6. Types of support per tool for layout guidelines

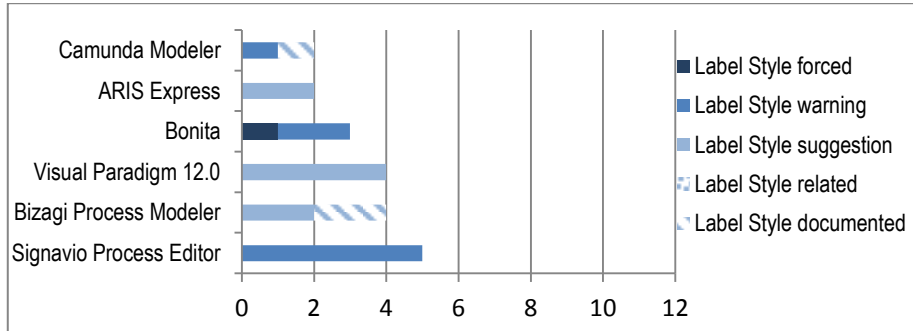


Fig. 7. Types of support per tool for label guidelines

6.4 Limitations

The most important limitation of this research is the limited set of tools that was investigated. However, during the tool selection phase a number of other tools were briefly investigated to test the available functionality in the free version of the tool (see tool selection criteria). None of these briefly explored tools seemed to offer substantially different or more support than the tools in the sample. Based on these experiences, it seems safe to conclude that the general conclusions hold for the wider range of BPMN tools.

A second limitation relates to the performed tests. Despite researching the tool extensively during the exploratory research phase in order to get familiar with the tool and its functionalities, we might have missed some of the intended support for guidelines. This was partly resolved by considering the "related guideline" as a type of support to take into account functionalities that are not a direct support for the tested guideline, but nevertheless offer relevant support related to the tested guideline.

7 Conclusion & Further Research

The goal of this research was to examine how extensive the support for the practical modelling guidelines is in a representative set of popular BPMN tools. First, we identified different types of support that indicate which level of support for a guideline is provided by a certain tool. These types of support were then used to indicate to which extent the BPMN modelling guidelines are implemented by a representative set of BPMN modelling tools. We found that almost every guideline was supported by at least one of the examined tools: only 8 out of the 56 atomic guidelines has no support at all. This leads to the conclusion that most guidelines are known by at least some of the tool builders. However, when considering the tools separately, each tool clearly lacks support for a significant part of the guidelines.

Considering the categories of guidelines, we found that the group of presentation guidelines is the best supported category. The second most supported category are the morphology guidelines, and the least supported guidelines are the guidelines that

count elements. However, the extent to which a certain category is supported, is highly dependent on the tool. In terms of types of support, warning and suggestions revealed to be the favourite types of support while enforcing guidelines is the least used type of support. The proportional use of different types of support is however dependent on the investigated tool and the category of guidelines.

As existing research has already demonstrated the positive impact on model quality (e.g. on model understanding and model correctness), following guidelines provided in literature can significantly enhance the quality of BPMN models. The presented results are therefore useful in the first place for anyone who needs to select a tool for modelling purposes. For example, when teaching BPMN to students, students should benefit from tools that offer a better support for model quality to make their exercises. Also business analysts who need to communicate about business processes with end-users will benefit from support for all guidelines that have a positive impact on model understanding and model correctness. Obviously, modelling being only one part of the full business process management cycle, other criteria than the support for modelling quality will come into play when selecting a full BPM suite.

Tool builders can use the results of this research to estimate the relative position of their tool in terms of relative support for business process model quality. The overview of existing guidelines and different levels of support that can be given, should be inspirational improve their tool's support.

While for a number of guidelines existing research has demonstrated the positive effect of those guidelines on e.g. model understanding or model correctness, further research should be performed to investigate the actual effect of a type of support on the actual application of a guideline by a modeller. Also, the relation between types of support and experienced utility, user-friendliness and guidelines acceptance may be an avenue of further research.

8 References

1. Dijkman, R.M., M. Dumas, and C. Ouyang, Semantics and analysis of business process models in BPMN. *Information and Software Technology*, 2008. 50(12): p. 1281-1294.
2. Born, M., J. Kirchner, and J.P. Müller. Context-driven Business Process Modeling. in *The 1st International Workshop on Managing Data with Mobile Devices (MDMD 2009)*. 2009. Milan, Italy.
3. Recker, J., et al. How Good is BPMN Really? Insights from Theory and Practice. in *14th European Conference on Information Systems*. 2006. Goeteborg, Sweden: Association for Information Systems.
4. Recker, J., BPMN Modeling - Who, Where, How and Why. *BPTrends*, 2008. 5: p. 1-8.
5. Zur Muehlen, M. and J. Recker, How Much Language is Enough? Theoretical and Practical Use of the Business Process Modeling Notation, in *Advanced Information Systems Engineering - CAiSE 2008*, M. Léonard and Z. Bellahsene, Editors. 2008, Springer, Heidelberg: Montpellier, France. p. 465-479.

6. Chinosi, M. and A. Trombetta, BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 2012. 34(1): p. 124-134.
7. Lübke, D., K. Schneider, and M. Weidlich. Visualizing use case sets as BPMN processes. in *3rd International Workshop on Requirements Engineering Visualization, REV'08*. 2008. IEEE.
8. Moreno-Montes de Oca, I., et al., A systematic literature review of studies on business process modeling quality. *Information and Software Technology*, 2015. 58(0): p. 187-205.
9. Moreno-Montes de Oca, I. and M. Snoeck, *Pragmatic guidelines for Business Process Modeling*. 2015: KU Leuven - FEB - Management Information Systems Group.
10. Daneva, M. A best practice based approach to CASE-tool selection. in *4th IEEE International Software Engineering Standards Symposium and Forum (ISESS'99), "Best Software Practices for the Internet Age"*. 1999. IEEE.
11. Rivas, L., et al., Tools selection criteria in software-developing Small and Medium Enterprises. *Journal of Computer Science & Technology*, 2010. 10(1).
12. Illes, T., et al. Criteria for software testing tool evaluation. a task oriented view. in *Proceedings of the 3rd World Congress for Software Quality*. 2005.
13. IEEE, *IEEE Std 1209-1992: IEEE Recommended Practice for the Evaluation and Selection of CASE Tools*. 1993.
14. Du Plessis, A.L., A method for CASE tool evaluation. *Information & Management*, 1993. 25(2): p. 93-102.
15. Le Blanc, L.A. and W.M. Korn. A structured approach to the evaluation and selection of CASE tools. in *Proceedings of the 1992 ACM/SIGAPP symposium on Applied computing: technological challenges of the 1990's*. 1992. ACM.
16. Harmon, P. and C. Wolf. *Business Process Trends; Business Process Modeling Survey*. 2011 [Retrieved: 2015 July]; Available from: http://www.bptrends.com/bpt/wp-content/surveys/Process_Modeling_Survey-Dec_11_FINAL.pdf.
17. *BPMN-Forum. BPMN tool related posts and discussions*. 2015 [Retrieved: 2015 July]; Available from: <http://bpmnforum.com/bpmn-tools>.
18. Signavio. *Guidelines by convention: Signavio Best Practice*. [Retrieved: 2015 July]; Available from: <http://academic.signavio.com>.
19. Camunda, *BPMN 2.0 Best Practices*. 2014.
20. Bizagi, *Bizagi Process Modeler User's Guide*. 2015.
21. Bonita. *Bonita*. [Retrieved: 2015 July]; Available from: <http://fr.bonitasoft.com>.
22. Ramakrishan, M., *Top Ten BPM tools you cannot ignore!*, in Wordpress. 2013.
23. Scheldeman, B. and T. Hoste, *The support of best practices by BPMN tools*, in Faculty of Economics and Business. 2015, KU Leuven: Belgium.
24. Mendling, J., H.A. Reijers, and W.M.P. van der Aalst, *Seven process modeling guidelines (7PMG)*. *Information and Software Technology*, 2010. 52(2): p. 127-136.
25. Mendling, J., et al., *Thresholds for error probability measures of business process models*. *Journal of Systems and Software*, 2012. 85(5): p. 1188-1197.