

# The impact of confusion on syntax errors in simple sequence flow models in BPMN

Jan Claes<sup>1</sup> and Gilles Vandecaveye<sup>1</sup>

<sup>1</sup>Department of Business Informatics and Operations Management,  
Ghent University, Tweeckerkenstraat 2, 9000 Gent, Belgium  
{jan.claes, gilles.vandecaveye}@ugent.be

**Abstract.** With the growing investments in getting to know and controlling their business processes, organizations produce many business process models. These models have become crucial instruments in the process lifecycle and therefore it is important that they are correct and clear representations of reality. They should contain as few errors and confusions as possible. Because we assume a causal relation between confusion and errors, we investigated it empirically. For our observation group, the data shows a correlation and temporal ordering between both. More in detail, avoiding implicit and redundant events and gateways is related with making less errors.

**Keywords:** process of process modeling, process model quality, syntactic quality, syntax error, confusion

## 1 Introduction

Nowadays, organizations put a lot of effort in documenting and analyzing their processes with the aid of graphical representations, i.e., with business process models [1, 2]. The de facto standard language they use for these models is BPMN [3]. The models are typically constructed by someone who has gained expertise in modeling through training and practice (“a modeler”), and with the input of people who know the process very well (“the process owners”) [4]. Unfortunately, case studies show that the quality of the produced models is poor, because they are often ambiguous, incomplete, or wrong [5–7]. Causes for these issues are categorized in two types: *knowledge problems* (the modeler has imperfect knowledge of the goal, the end users, the process, or the modeling language) and *cognitive problems* (the modeler does not succeed in externalizing their knowledge about the process perfectly in the model) [8]. This paper aims to provide a contribution towards solving the latter kind of problems by analyzing the impact of confusion during modeling on making errors.

Because of the complexity of the study topic, the scope is limited in several ways in order to provide initial knowledge that may be of importance when working towards a solution.

- The language was reduced to its minimum by excluding all but the 6 fundamental constructs for modeling the control flow of processes [9] in order to focus first on essential aspects only.
- The modeling was reduced to an assignment of translating a pre-structured textual description of a process into a graphical business process model in order to focus first on cognitive problems and the task of the modeler only.
- The analysis concerns *syntax* issues only in order to focus first on the type of issues that can be determined most objectively.
- The study used observational data of master students as a proxy for *novice* modelers in order to reduce the variability of modeling experience, domain knowledge, problem solving skills, etc. in the dataset.

In our study of the impact of confusion on making errors during modeling, an existing dataset was explored. It contains tool operations of 146 modelers and it was enriched with the timing of adding/removing confusing constructs and of adding/removing syntactical errors in the model. Next, the relation between confusion and errors was studied in detail. The conclusion is not surprising: confusion correlates with making more and solving less errors. More in particular, the study confirms the importance of using explicit events and gateways in a well-structured way.

Yet, since completing the data and constructing the proof for this conclusion proved to be challenging (even under the complexity-reducing limitations mentioned above), we believe that it is important to publish this study. In fact, this is the first paper to provide such detailed explanatory and statistical material on the relation between confusion and errors during modeling. It answers to the criticism in literature about process modeling guidelines, some of which lack an empirical foundation. It provides input for teachers who may now be able to teach their students how to *avoid* certain errors instead of *detecting and correcting* them. It provides useful input for tool developers who can improve their syntax support during modeling. On the other hand, the external validity of the research is not investigated, and readers should be cautious when generalizing the results of this paper.

This paper is structured as follows. Section 2 presents related work. Section 3 describes the data collection and Section 4 discusses the data analysis. Section 5 provides a discussion. Section 6 provides a summarizing conclusion.

## 2 Background and related work

The research described in this article builds further on the work of De Bock & Claes [10], who studied the origin and evolution of syntax issues in simple BPMN models. They introduced a classification scheme, which is partially adopted here. The scheme makes a distinction between errors, irresolutions, and confusions (cf. Table 1). A syntax *error* is a clear fault against the specified syntax of the modeling language. Further, since some constructs or their meaning are not clearly or consistently defined in the BPMN syntax specification, the paper distinguishes these from errors and calls them *irresolutions*. Next, some constructs are clearly

syntactically correct, but they are still considered as constructions to avoid. These are called *confusions*. For this paper, because of their ambiguity, we decided to consider *irresolutions* also as confusing syntactical constructs.

Table 1. List of identified syntax issues and their occurrence in the dataset

	<b>Construction</b>	<b>Code</b>	<b>#made</b>	<b>#solved</b>
Confusion	Contains no start event	0s	0	0
	Contains no end event	0e	1	0
	Contains multiple start events	S	94	20
	Contains multiple end events	E	85	8
	Multiple parallel sequence flows from non-gateway	Sa	7	5
	Multiple optional sequence flows towards non-gateway	Jx	244	34
	No label for edge departing from XOR splits	Lx	358	269
Irresolution	One gateway combines a join and split feature	C	164	14
	Wrong type of join combined with a certain split	W	22	7
	No join gateways in case of optional iterations	I	0	0
	Wrong nesting of gateways	N	19	3
	AND and XOR are joined together in one join gateway	T	13	0
	No label for activity	La	2	2
	Sequence flow with arrow at each side	DS	22	11
Error	Implicit start event without implicit end event	0se	0	0
	Implicit end event without implicit start event	0es	1	0
	Not all of the paths are closed (missing end event?)	P	57	36
	Condition at sequence flow originating from start event	Ls	0	0
	Sequence flow to start event	Bs	9	0
	Sequence flow from end event	Be	0	0
	Sequence flow from start event missing	Ms	0	0
	Sequence flow to end event missing	Me	0	0
	Multiple optional sequence flows from non-gateway	Sx	99	31
	Multiple parallel sequence flows towards non-gateway	Ja	4	0
Gateway with only 1 incoming and 1 outgoing sequence flow	1e	373	359	

Second, the scheme recognizes the fact that at certain times during modeling one cannot be sure whether a *missing* construct is an issue, since its placement can be deliberately postponed or unintentionally forgotten. The proposed way of dealing with this, is to make a distinction between uncertain issues (certain *missing* constructs in incomplete parts of the model) and definite issues (*missing* constructs in completed parts of the model, and *wrong* constructs). For this paper, we consider both the making and solving of issues, further neglecting the difference between uncertain and definite. Only when an uncertain issue evolves into its definite issue equivalent, we disregarded the “solving of the uncertain issue” and the “making of the definite issue” in order to capture the making of the issues at the earliest time.

This paper contributes to the process model quality literature. Since there are ample papers about this topic, we refer to the recent, extensive literature reviews of Figl [11] and De Meyer & Claes [12]. Besides literature about *what* is a high-quality process model, there is also a growing body of literature about *how* process models of

high quality can be constructed. For example, Guidelines of Modeling provides high-level recommendations such as to optimize correctness, relevance, economic efficiency, clarity, comparability, and systematic design [13]. Nevertheless, it was criticized for its lack of concrete, operational support and the lack of empirical evidence. On the other extreme, the Seven Process Modeling Guidelines [14], Ten Process Modeling Guidelines [15], and Quality Indicators related to Gateway Complexity [16] take a dominantly empirical angle as to finding the optimal thresholds for certain process model metrics such as the size of the model, the number and nesting depth of gateways, etc. Further, Concrete [17] and Abstract Syntax modification Patterns [18] bundled detailed guidelines for process modeling from existing literature, software, and cases. What they all have in common, is that the focus is on *how* to improve process models, and less on *why* certain guidelines have certain effects on the quality of the produced models (i.e., the cognitive aspect). The current paper aims to address this gap by studying the relations between different factors of process model quality. More in particular, it relates the amount of syntactical confusion in a model with making and solving errors during modeling.

### 3 Data collection

As mentioned, this study builds on an existing dataset used in previous research [8, 10]. The dataset contains all the operations of 146 modelers in our modeling tool while constructing a graphical model from a textual process description<sup>1</sup>. For each operation, the dataset contains, the type (e.g. ‘create activity’, ‘move xor gateway’, ‘delete edge’), the time, the id of the model element to which it applies, and the code(s) of the issue(s) that is/are made or solved by the operation. For this study, we could use only 122 of the 146 modeling sessions, disregarding the models that contain no confusions nor errors, and Petri-Net-style models (using *event* symbols for *places*).

From these data, we derived for each modeling session the measures presented in Table 2. When considering *operations* in our study, the ‘*move*’ operations are disregarded. They are not relevant, because they have no impact on syntax issues.

Table 2. Measures per modeling session

Code	Formula	Definition
CI+		The number of confusions and irresolutions made during modeling
CI-		The number of confusions and irresolutions solved during modeling
CI	$CI+ - CI-$	The number of confusions and irresolutions remaining after modeling
E+		The number of errors made during modeling
E-		The number of errors solved during modeling
E	$E+ - E-$	The number of errors remaining after modeling
Op		The number of relevant operations used to create the model
T		The number of minutes used to create the model (T stands for <i>time</i> )

<sup>1</sup> For details, see the 2015 experiment at <https://www.janclaes.info/experiments>.

The above measures represent totals over the whole session. At each operation that created or removed a syntax issue, the measures in Table 3 were calculated as well.

Table 3. Measures per syntax issue influencing operation

Code	Formula	Definition
ci		The number of confusions and irresolutions currently in the partial model
[ci]	ci/max(ci)	The relative ci expressed as a percentage of the maximum ci in the model
e+		The number of errors made by the current operation
e-		The number of errors solved by the current operation
e		The number of errors currently in the partial model
[e-]	e-/e	The <i>relative</i> number of <i>existing</i> errors solved by the current operation
op		The number of operations since the previous syntax issue operation
t		The number of minutes since the previous syntax issue operation
pe+	e+/op	The current chance to make an error: the number of errors made divided by the number of operations since the previous issue operation
p[e-]	[e-]/op	The current chance to solve an error: the <i>relative</i> number of errors solved divided by the number of operations since the previous issue operation

## 4 Data Analysis

### 4.1 General relations between confusion and errors

First, we look at the relation between confusion (CI) and errors (E) at the model level. Correlations are calculated between the variables of Table 2. The results are displayed in Table 4.

Table 4. Correlations at model level  
(displaying Spearman's rho and the 2-tailed significance)

N=122	CI+	CI-	CI	E+	E-	E	Op	T
CI+	-	,731**	,759**	0,046	-,218*	,316**	-0,113	-0,05
	.	0,000	0,000	0,615	0,016	0,000	0,217	0,587
CI-	,731**	-	,187*	0,011	-0,042	0,044	0,085	,179*
	0,000	.	0,039	0,907	0,644	0,630	0,353	0,048
CI	,759**	,187*	-	0,142	-0,151	,431**	-,232*	-,216*
	0,000	0,039	.	0,118	0,096	0,000	0,010	0,017
E+	0,046	0,011	0,142	-	,776**	,472**	,208*	0,036
	0,615	0,907	0,118	.	0,000	0,000	0,021	0,696
E-	-,218*	-0,042	-0,151	,776**	-	-0,054	,262**	0,109
	0,016	0,644	0,096	0,000	.	0,555	0,003	0,231
E	,316**	0,044	,431**	,472**	-0,054	-	-0,086	-0,136
	0,000	0,630	0,000	0,000	0,555	.	0,347	0,136
Op	-0,113	0,085	-,232*	,208*	,262**	-0,086	-	,561**
	0,217	0,353	0,010	0,021	0,003	0,347	.	0,000
T	-0,05	,179*	-,216*	0,036	0,109	-0,136	,561**	-
	0,587	0,048	0,017	0,696	0,231	0,136	0,000	.

\*\* Correlation is significant at the 0.01 level (2-tailed).

\* Correlation is significant at the 0.05 level (2-tailed).

The next conclusions can be drawn from Table 4.

**Made versus solved.** It can be noted that the more confusions are made (CI+), the more\* confusions are solved (CI-). Similarly, the more errors made (E+), the more\* errors solved (E-). This makes sense, since issues that are not made cannot be solved. Yet, these two correlations are important to interpret the following results.

**Made/solved versus total.** The more confusions made (CI+), the more\* remain in the final model (CI). The more errors made (E+), the more\* remain in the final model (E). This appears to be obvious but given the previously discussed correlation between confusions/errors made and confusions/errors solved, it is important to verify that not all issues are solved in the end. This also explains why solving more confusions/errors (CI-/E-) does not result in less confusions/errors in the final model (CI/E).

**Making/solving versus operations and time.** Making more confusions (CI+) does not per se cost more operations (Op) or time (T). Solving more confusions (CI-) costs more<sup>x</sup> operations (Op) and more\* time (T). Interestingly, making and solving errors are both related with more\* operations (Op), and more<sup>x</sup> time (T). This is consistent with the findings by Bolle & Claes [19].

**Confusions versus errors.** The most interesting part of Table 4 however, is the relation between confusions and errors. Making more confusions (CI+) is related to making more<sup>x</sup> errors (E+), solving less\* errors (E-) and more\* errors remaining (E). Solving confusions (CI-) is not found to be related to making, solving, or remaining errors. Remaining confusions (CI) is related to making more<sup>x</sup> errors (E+), solving less<sup>x</sup> errors (E-), and to more\* errors remaining (E).

In summary, these are statistically (in)significant indications of a positive relation between confusions (CI+/CI-/CI) and errors (E+/E-/E). However, the above tests neglect the *timing* of confusions and errors. Since we assume a causal relation between confusion and errors, it is more precise to verify whether more errors are made/solved *at the time* that more confusion exists in the model. This does not proof causality, but temporal ordering is a necessary condition for causality (on top of correlation). Let us now examine this in more detail.

#### 4.2 More errors are made when more confusion exists in the model

For the analysis of the relation between errors made (e+) during modeling and the number of confusions existing in the model (ci), only those entries in the dataset are selected where errors are made. Note that a single operation can make one error and at the same time solve another one. The net effect on e is zero, but this record is still included. The correlation results are summarized in Table 5.

---

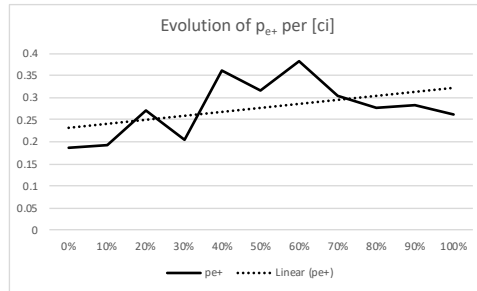
\* Statistically significant result

<sup>x</sup> Results are not statistically significant

Table 5. Relation between confusion and errors made (displaying Spearman’s rho and the 2-tailed significance)

N=493	e+	p <sub>e+</sub>
ci	0	,312**
	0,992	0,000
[ci]	0,044	,177**
	0,328	0,000

\*\* Correlation is significant at the 0.01 level.



As can be seen, there is no statistically significant relation in our dataset between the number of errors made (e+) and the absolute number of confusions present in the partial models at the time of the error (ci). The reason for this may be because the number of errors that can be made is dependent on the number of operations of the modeler. Therefore, the data was normalized. The number of errors made is considered per operation made (p<sub>e+</sub> = e+/op). It represents the chance for an operation to be an error. The number of confusions in the partial models is considered relative to the maximum number in the model ([ci] = ci/max(ci)). This gives a more correct comparison between models and between number of operations. The correlation analysis now shows a statistically significant relation.

The chart in Table 5 shows a more detailed view of the relation in our dataset. It represents the deciles of the confusion level ([ci]) and sets out the average chance to make errors (p<sub>e+</sub>), as well as a linear trendline of this relation. In general, it can be concluded that the more relative confusions were present in the partial model, the more\* the chance was to make errors.

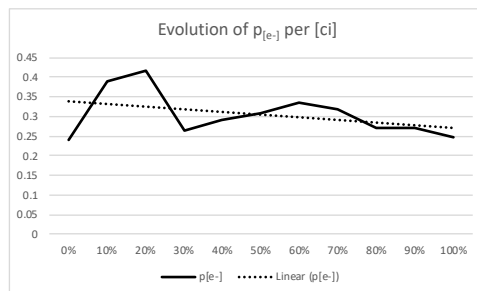
### 4.3 Less errors are solved when more confusion exists in the model

For the analysis of the relation between errors solved (e-) during modeling and the number of confusions existing in the model (ci), only those entries in the dataset are selected when errors are solved. The correlation results are summarized in Table 6.

Table 6. Relation between confusion and errors solved (displaying Spearman’s rho and the 2-tailed significance)

N=397	e-	[e-]	p <sub>[e-]</sub>
ci	-0,022	-,228**	-0,046
	0,661	0,000	0,363
[ci]	-0,017	-,173**	-,147**
	0,739	0,001	0,003

\*\* Correlation is significant at the 0.01 level.



Again, there is no statistically significant relation in the dataset between the number of errors solved ( $e^-$ ) and the absolute number of confusions in the partial models ( $ci$ ). The number of errors that can be solved depends on the number of errors existing in the model and on the number of operations of the modeler. The data was again normalized for more correct comparison. The number of errors solved is considered per number of errors existing ( $[e^-] = e^-/e$ ) and per operation made ( $p[e^-] = [e^-]/op$ ). The correlation analysis now shows statistically significant relations.

The chart in Table 6 shows the more detailed view of the relation in our dataset. It represents the deciles of the confusion level ( $[ci]$ ) and sets out the average chance to solve errors ( $p[e^-]$ ), as well as a linear trendline of this relation. In general, it can be concluded that the more relative confusions were present in the partial model, the less\* chance there was to solve errors.

#### 4.4 Which confusions cause which errors?

Next, the effects of individual confusions, irresolutions, and errors are studied. We limited the study to those variables who have been observed at least 50 times, which can be derived from Table 1. As such, the effect of confusions *multiple start events* ( $c_s$ ), *multiple end events* ( $c_e$ ), *multiple optional sequence flows towards non-gateway* ( $c_{jx}$ ), *no label for edge departing from XOR splits* ( $c_{lx}$ ), and irresolution *one gateway combines a join and split feature* ( $i_c$ ) on the errors *not all paths are closed* ( $e_p$ ), *multiple optional sequence flows from non-gateway* ( $e_{sx}$ ), and *gateway with only 1 incoming and 1 outgoing sequence flow* ( $e_{1e}$ ) are calculated. The results are summarized in Table 7. They are discussed below, column per column.

Table 7. Correlations at issue level (unexpected signs are marked in grey) (displaying Spearman's rho and the 2-tailed significance)

	$pe_{p+}$	$pe_{sx+}$	$pe_{1e+}$	$pe+$	$p[e_p^-]$	$p[e_{sx}^-]$	$p[e_{1e}^-]$	$p[e^-]$
$c_s$	0,155 0,267	,222* 0,04	,108* 0,043	,216** 0	0,166 0,347	,448* 0,017	0,044 0,418	-,203** 0
$c_e$	,422** 0,002	-0,028 0,796	,177** 0,001	,162** 0	-0,152 0,392	-0,228 0,243	0,099 0,069	0,031 0,543
$c_{jx}$	0,1 0,477	,255* 0,018	-0,012 0,819	0,037 0,409	-0,294 0,091	0,146 0,46	-0,07 0,2	-,107* 0,032
$c_{lx}$	,353** 0,01	0,141 0,197	,206** 0	,239** 0	0,079 0,657	0,15 0,447	,220** 0	,275** 0
$i_c$	0,136 0,331	,216* 0,046	0,003 0,951	0,068 0,132	-0,275 0,115	-0,134 0,498	-0,008 0,879	-0,08 0,111
$[ci]$	,496** 0	,247* 0,026	,126* 0,018	,177** 0	-,517** 0,002	0,134 0,522	0,025 0,644	-,147** 0,003

\*\* Correlation is significant at the 0.01 level.

\* Correlation is significant at the 0.05 level (2-tailed).

Not closing all paths of the model ( $e_{p+}$ ) happens more\* when there are multiple end events in the model ( $c_e$ ) and when more edges from xor gateways are not labeled ( $c_{lx}$ ). We expect that **using multiple end events causes the modeler to forget to close all paths in the model (H1)**, whereas we do not assume a *causal* relation



between not labeling edges and forgetting to close all paths. On the other hand, these confusions do not appear to relate<sup>x</sup> with *solving* this error.

Using an implicit xor split gateway ( $e_{sx+}$ ) is not allowed by the BPMN syntax. This error was made more<sup>\*</sup> when there were multiple start events ( $c_s$ ), when the modeler used also more implicit xor join gateways ( $c_{jx}$ ) and gateways that combined split and join functionalities ( $i_c$ ). We propose that **people who use implicit gateways are not always aware of when this is allowed ( $c_s$ ,  $c_{jx}$ ,  $i_c$ ) and when not ( $e_{sx}$ ) (H2)**. Surprisingly, these errors are more<sup>\*</sup> *solved* when there are more start events in the model.

Having gateways in the model that are not used for splitting or joining multiple paths ( $e_{1e+}$ ) happens more<sup>\*</sup> when there are multiple start events ( $c_s$ ) or multiple end events ( $c_e$ ) in the model, and when there are more edges from xor gateways without labels ( $c_{lx}$ ). **Perhaps having multiple start and end events increases the structural complexity of the model, causing the modeler to forget adding the postponed paths for which a gateway was already created (H3)**. We do not assume a *causal* relation between the labels of edges and forgetting some paths at gateways, which is supported by the unexpected correlation with *solving* these errors.

This more detailed analysis does not bring conclusive answers. It contributes to the study by adding preliminary insights that can be derived from the statistics. They are formulated in the form of hypotheses (H1-H3), which can be studied in future work.

## 5 Discussion

### 5.1 Impact

The impact on research of this study is that it provides (additional) empirical evidence for a number of proposed process modeling guidelines of the Seven Process Modeling Guidelines (7PMG) [14], Ten Process Modeling Guidelines (10PMG) [15], Concrete Syntax Patterns (CSP) [17], Abstract Syntax Patterns (ASP) [18], and Quality Indicators (QI) [16]. The general lack of such evidence on the content, interrelations, and relevance of these guidelines is denounced in multiple critiques [20–23]. The guidelines to which supporting (additional) evidence is formulated in this paper, are listed below.

- *Use 1 start and 1 end event* (7PMG) and *Use no more than 2 start and end events* (10PMG). This relates to confusions S, E, 0s, 0e; errors 0se, 0es; and hypotheses H1, H2, H3.
- *Model as structured as possible* (7PMG, 10PMG, ASP) and *Use design patterns to avoid mismatch* (10PMG). This proposes to use explicit and paired gateways, which should avoid confusions Sa, Jx; irresolutions C, W, I, N, T, DS; and errors Sx, Ja, 1e; and it relates to hypothesis H2.

- *Use explicit representation (CSP)*. This may refer to avoiding implicit events and gateways, which is related to confusions 0s, 0e, Sa, Jx; irresolutions C, I, T, DS; errors 0se, 0es, Sx, Ja; and hypothesis H2.
- *Limit the difference in the number of input/output flows between splits and joins (QI)* refers to both previous examples, because it is realized by structured modeling and/or using explicit gateways.
- *Use of textual annotation (CSP) and Naming guidance (CSP)*. This links to the use and format of text in models, which relates to the confusion Lx, the irresolution La, the error Ls, and the discussion about Lx in Section 4.4.

The impact on practice is that the research provides extra insights into the relation between confusion and errors to practitioners. Even in models that are made as input for computer programs, where only pure syntax errors could seem to be important, it now appears to be important to avoid confusing constructs as well, since they may cause modeling errors during modeling. Besides the modelers, this research should also support teachers. It is always easier to train modelers to apply certain guidelines when the reason *why* they are important can be illustrated. This paper contributes to such illustration of the importance to also focus on avoiding confusion, being a potential cause for errors. Third, tool developers have spent a great deal of effort to support modelers in avoiding syntax errors by highlighting them or by providing an overview of the syntax errors after modeling. The current study provides input to add a level of *warnings* to their support features (just as programming editors do).

## 5.2 Limitations and future work

As discussed in the introduction, the scope of the research is limited. The reduced language, the artificial case and artificial modelers limit the ecological validity of the research. The focus on syntax only, on the modeler's contribution only, on one case only, and the lack of focus on consequences for the readers of a model put a limit to the external validity of the research. Therefore, one should be cautious to generalize the relations discussed above. The research should be considered as an explorative empirical study that provides initial insights and hypotheses for further research.

In Section 4.4, three hypotheses are formulated, which can be studied further. But more generally, it would be useful to study systematically the effect of all listed confusions and irresolutions, both on making errors during modeling and on the final user understandability of the model. Further, although methodologically more challenging, it is advised to include not only pragmatic (cf. user-understanding), but also semantic quality into the research. Using the same artificial setup where a modeler is instructed to create a diagram representing the knowledge described in a textual description, it is possible to use a similar methodology as the one applied here to derive the timing and type of semantic errors (missing, wrong, redundant, inconsistent, and unnecessary constructs), and to study their interrelations and links with syntax and understanding.

## 6 Conclusion

The data of 122 (of 146) modeling sessions was used to build a dataset containing the timing of all operations to construct the model. By adding whether each operation initiated or solved certain types of syntax issues, we were able to study the relation between confusing and wrong syntax constructs. In general, the conclusion is that confusion may lead to errors and therefore it should be avoided as much as real errors. The contribution of this paper is not in this conclusion per se, but in its detailed proof (31.588 operations were analyzed, and 2.489 syntax issues were documented), and explanatory knowledge that is added to this conclusion. It provides interesting knowledge about the presence of various types of confusing constructs and syntax errors and their potential (causal) relations.

## 7 References

1. Moreno-Montes De Oca, I., Snoeck, M., Reijers, H.A., et al.: A systematic literature review of studies on business process modeling quality. *Inf. Softw. Technol.* 58, 187–205 (2015).
2. Aguilar-Savén, R.S.: Business process modelling: Review and framework. *Int. J. Prod. Econ.* 90, 129–149 (2004).
3. Recker, J.: Opportunities and constraints: the current struggle with BPMN. *Bus. Process Manag. J.* 16, 181–201 (2010).
4. Grosskopf, A., Edelman, J., Weske, M.: Tangible Business Process Modeling – Methodology and Experiment Design. In: *Proc. BPM '09 Workshops. LNBP 43*. pp. 489–500 (2010).
5. Hassan, N., Recker, J., Bernhard, E.: A study of the use of business process modelling at suncorp. , Brisbane, Australia (2011).
6. Gruhn, V., Laue, R.: What business process modelers can learn from programmers. *Sci. Comput. Program.* 65, 4–13 (2007).
7. Mendling, J., Verbeek, H.M.W., Van Dongen, B.F., et al.: Detection and prediction of errors in EPCs of the SAP reference model. *Data Knowl. Eng.* 64, 312–329 (2008).
8. Claes, J., Vanderfeesten, I., Gailly, F., et al.: The Structured Process Modeling Method (SPMM) - What is the best way for me to construct a process model? *Decis. Support Syst.* 100, 57–76 (2017).
9. Zur Muehlen, M., Recker, J.: How much language is enough? Theoretical and practical use of the Business Process Modeling Notation. In: Bellahsene, Z. and Léonard, M. (eds.) *Proc. CAiSE '08. LNCS 5074*. pp. 465–479. Springer (2008).
10. De Bock, J., Claes, J.: The Origin and Evolution of Syntax Errors in Simple

- Sequence Flow Models in BPMN. In: Matulevičius, R. and Dijkman, R.M. (eds.) Proc. Caise '18 Workshops. LNBIP 316. pp. 155–166. Springer (2018).
11. Figl, K.: Comprehension of procedural visual business process models - A literature review. *Bus. Inf. Syst. Eng.* 59, 41–71 (2017).
  12. De Meyer, P., Claes, J.: An overview of process model quality literature - The Comprehensive Process Model Quality Framework, (2018).
  13. Becker, J., Rosemann, M., Von Uthmann, C.: Guidelines of business process modeling. In: Van der Aalst, W.M.P. et al. (eds.) *Business Process Management. Models, Techniques, and Empirical Studies. Part I. LNCS 1806.* pp. 30–49. Springer (2000).
  14. Mendling, J., Reijers, H.A., Van der Aalst, W.M.P.: Seven process modeling guidelines (7PMG). *Inf. Softw. Technol.* 52, 127–136 (2010).
  15. Mendling, J., Sánchez-González, L., García, F., et al.: Thresholds for error probability measures of business process models. *J. Syst. Softw.* 85, 1188–1197 (2012).
  16. Sánchez-González, L., García, F., Ruiz, F., et al.: Quality indicators for business process models from a gateway complexity perspective. *Inf. Softw. Technol.* 54, 1159–1174 (2012).
  17. La Rosa, M., Ter Hofstede, A.H.M., Wohed, P., et al.: Managing process model complexity via concrete syntax modifications. *IEEE Trans. Ind. Informatics.* 7, 255–265 (2011).
  18. La Rosa, M., Wohed, P., Mendling, J., et al.: Managing process model complexity via abstract syntax modifications. *IEEE Trans. Ind. Informatics.* 7, 614–629 (2011).
  19. Bolle, J., Claes, J.: Investigating the trade-off between the effectiveness and efficiency of process modeling. In: Proc. BPM '18 Workshops. pp. 1–12 (2018).
  20. Chen, C.: Top 10 unsolved information visualization problems. *IEEE Comput. Graph. Appl.* 25, 12–16 (2005).
  21. Nelson, H.J., Poels, G., Genero, M., et al.: A conceptual modeling quality framework. *Softw. Qual. J.* 20, 201–228 (2012).
  22. Rockwell, S., Bajaj, A.: COGEVAL: Applying cognitive theories to evaluate conceptual models. *Adv. Top. Database Res.* 4, 255–282 (2005).
  23. Rogers, Y., Scaife, M.: How can interactive multimedia facilitate learning? In: Lee, J. (ed.) *1st International Workshop On Intelligence and Multimodality in Multimedia Interfaces. Research and Applications. AAAI.* pp. 1–25 (1998).