KU LEUVEN

FACULTY OF ECONOMICS AND BUSINESS

Pragmatic guidelines for Business Process Modeling

Moreno-Montes de Oca I, Snoeck M.

KBI_1509

Pragmatic guidelines for Business Process Modeling

Technical Report

Isel Moreno-Montes de Oca
Department of Computer Science
Central University of Las Villas
Villa Clara,
Cuba

and

Monique Snoeck
Management Information Systems group
Faculty of Economics and Business
KU Leuven
Belgium

20 November 2014

Contents

1. Introduction

1.1. Abstract

This document presents an overview of pragmatic guidelines for undertaking business process modeling tasks. These guidelines can support practitioners and non-experts in the difficult task of modeling business process models. In particular, the guidelines focus on obtaining high quality of business process models in terms of their quality as a model, i.e. as a representation of a certain reality. In other words, the guidelines do not say anything about the quality of business processes as operationalization of a certain business strategy: they do not make statement in terms of e.g. cost effectiveness, customer friendliness, … The guidelines presented in this paper focus on the quality of the model as a representation of a reality: is the representation correct w.r.t the reality, does it make a correct use of the symbols, is it understandable, … etc. A more detailed description of the different dimensions of model quality used in this paper can be found in [1].

The guidelines presented in this document have been derived from a systematic literature review [2] and are supplemented with other aspects the authors consider relevant in order to make the guidelines more useful (e.g., threshold values, action or example, description, and associated quality metrics). The guidelines have a wide applicability across various industries and business processes modeling tasks. They are aimed at students and practitioners who are new to the task of business process modeling.

1.2. Structure of the document

Sections 2 to 4 of this document present a summary of the guidelines extracted from different sources. This summary is complemented with a detailed presentation of the literature in Appendix 1. For every guideline collected in this document, we present in Appendix 1 a table with the following information extracted from the literature:

- Original Guidelines: guidelines as stated in the paper.
- Reference: source of the guideline.
- Type of guideline: refers to the term as used by the authors (Guideline, Pattern, Metric, Smell, etc.). Knowledge is for those "guidelines" derived from hypothesis testing.
- Empirical Evidence: The value "yes" is used when the guidelines are supported with empirical validation. In a minor number of cases, the global research has been validated, but not the individual guideline itself. In these cases, we put a "(yes)". *An example are the Process Model Smells from* [3]. If the guidelines were not validated or do not rely on other validated sources, it is a "no".
- Type of Quality (CMQF): This refers to the quality dimension of the CMQF [1] addressed by the guideline. A (*) indicates that the quality dimension was not defined by the authors of the source paper, but added by the researchers. In case the author did not use a name from the CMQF framework, the researchers translated the term used by the author to the corresponding quality dimension of the CMQF.
- Threshold: threshold value mentioned in the source paper.
- Associated Metric: metric mentioned by the source paper. Indicated by a (*) are those metrics suggested by the researcher.

We classified the guidelines into three different groups.

- A first group consists of guidelines that count a number of elements. These guidelines are presented in section 2.
  These guidelines are not relating elements with each other, like relating a split gateway to a join gateway; they just count elements.
    - We categorize *gateway complexity* in this group, because this metric refers to the number of incoming arcs and outgoing arcs for a gateway (the same holds for *events complexity*).

- A second group looks at the morphology of a process in terms of cycles, parallelism, depth, structuredness, etc. These guidelines are presented in section 3.
    - We classify *diameter* here because it looks at the shape of a model (long & thin versus wide & short). *Diameter* in some way is related to the use of gateways. More nested gateways implies wider model, less use of gateways implies thin model.
    - Also the guidelines about the usage of (inclusive) OR gateways are included here as they refer to the structuredness of the model.
- Finally, section 4 groups all guidelines related to presentation aspects:
  A third group of guidelines pertains to presentation of the model that can be applied without changing the semantics of the model itself. Here we find subgroups about layout and labeling style.

For some guidelines, we create "sub-guidelines". These are the guidelines related to the guideline and that refer to specific aspects of the more general guideline.

We use a fixed format to document the guidelines and to discuss different factors related to them. This format contains: (a) problem, (b) discussion, (c) unified guideline, (d) motivation, (e) action and (f) associated metrics. The 'unified guideline' is our proposal for formulating the guideline such that it summarizes all related guidelines found in the literature into one guideline. Motivation links guidelines to quality dimensions of the CMQF. Action, on the other hand, is a possible way to fix the problems in the models in order to satisfy the guideline. Finally, metrics lists structural metrics that allow to quantify the effect of the use of the guideline in the models. Whenever metrics are present, we include the metrics suggested by the authors; if not, we suggest our own metrics (indicated with a **).

2. Guidelines that count elements

2.1. Number of elements

Problem 1:

- The model contains a high number of elements (i.e. gateways, activities and events).

Discussion:

- When capturing complex business processes, the resulting process model may be too large to be understood at once [4]. To improve the readability of these "big process models", it is recommended to simplify them by hiding certain parts within subprocesses (a subprocess represents a self-contained, composite activity that can be broken down into smaller units of work). As a result, some details of the process model will be omitted in the higher level process and the readability of the whole process model is improved. The size of the (expanded) process model is not reduced. Thus, we suggest to use as few elements as possible per process model level, and in particular to decompose a process model if this has more than 31 flow objects [4].

Unified Guideline 1:

- Decompose models with more than 31 elements.

Motivation:

- To improve the readability and understandability of the model by stakeholders. Larger models are also more difficult to maintain [3, 5, 6]. Applying this guideline will lead to an improvement of pragmatic and empirical quality. This guideline is supported by empirical evidence.

Action:

- In order to use a subprocess, we first need to identify groups of related activities, i.e. those activities that achieve together a particular goal or generate a particular outcome in the process model under analysis. These activities, and their connecting gateways, can be encapsulated in a subprocess [4]. In order to identify candidate subprocesses, we can use, for example the block-structuredness criterion as recommended in [7]. Another criterion that can be followed when decomposing the models is "connectedness". Approaches that consider the connectedness of subprocesses seem the most attractive to pursue according to [8]. A collection of nodes is connected if the nodes in the collection are more strongly connected by arcs to each other than to nodes outside this collection [8]. Note that this definition implies that for an automatic discovery of subprocesses a parameter is involved that should express how strong the connection between nodes in a collection must be.

Associated Metric:

- **Number of elements per process model level

Problem 2:

- Model contains duplicate elements (e.g. identical start events, identical end events, identical activities) or fragments, capturing the same control-flow logic.

Discussion:

- When capturing business processes, the resulting process model may contain duplicate elements (e.g. one activity is repeated in the model). Avoiding repetitions is essential since models with duplicated elements may have purposeless bigger size, and this influences negatively understanding and readability of the models by stakeholders. Some error patterns (e.g. those presented in[9]) refer to this problem. However, there are more situations than the ones given in these patterns.

Sometimes, when representing real process models, the same set of steps is carry out more than one time. In these cases, it is common to find repeated model fragments within a single and across different process models.

Unified Guideline 2:

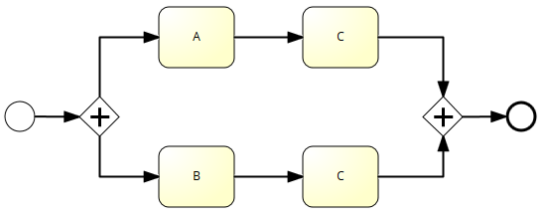
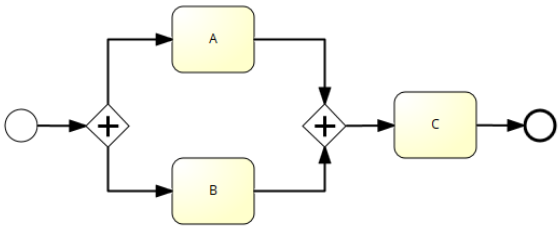
- Avoid duplicate elements and fragments in your process models.

Motivation:

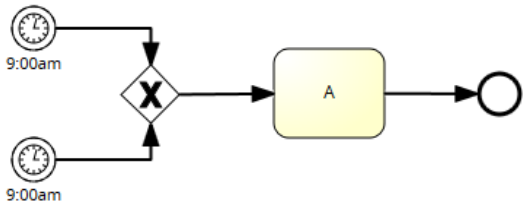
- Duplications lead to problems in model maintenance due to oversized models as well as model inconsistencies [3]. Reduction of duplications implies decrease in size, which at the same time improves the pragmatic and empirical quality of the models. This guideline is supported by empirical evidence.

Action:

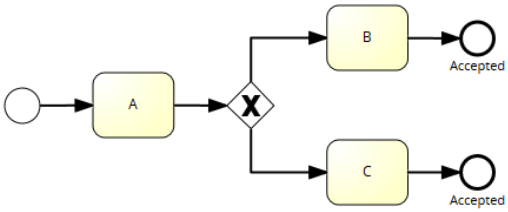
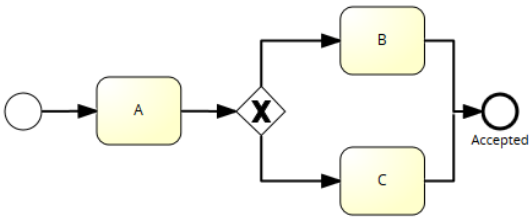
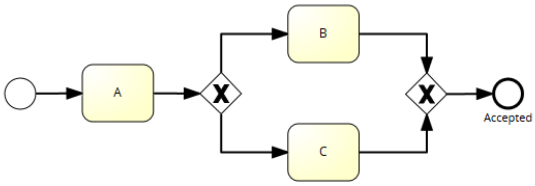
- **Detect duplicate elements e.g. by means of error patterns** [9].
  In the following figure, the same activity is located at the last position of all incoming paths of an AND join. In such a case, the activity can be moved outside the control block. The figure shows only one of many different cases: the gateway does not have to be an AND gateway (it is extensible for exclusive and inclusive OR gateways), and to move one or more identical activities out of a control block is also possible if identical nodes directly follow a split node.

a) Business process model with duplicate activity	b) Reduction of model a) by unifying the duplicated activity
	

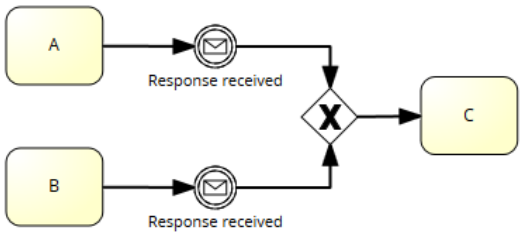
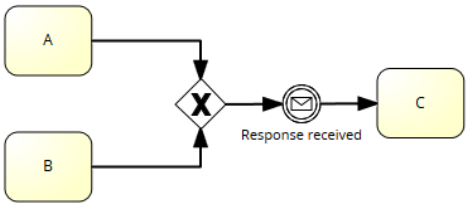
- Simplify the model if it contains identical start events preceding the same join gateway [3].

a) Business process model with duplicate start event	b) Reduction of model a) by unifying the start event
	

· Simplify the model if it contains identical end events:

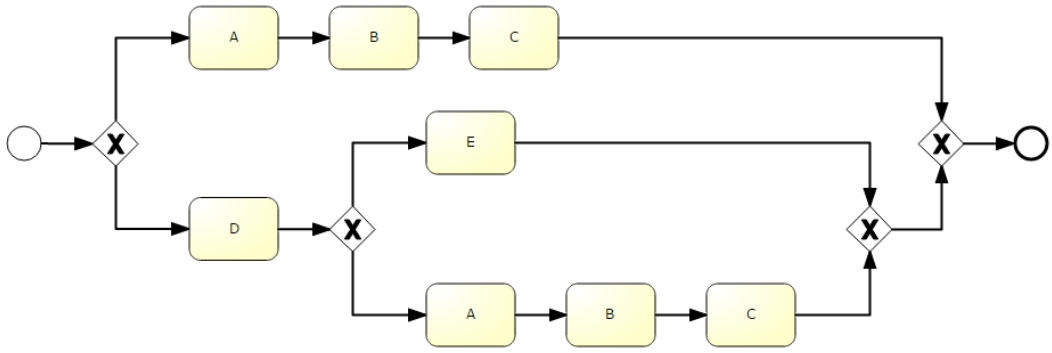
<table>
<tr><td>a) Business process model with duplicate end event.</td><td>b) Reduction of model a) by unifying the end event.</td></tr>
<tr><td rowspan="3"> 
</td><td> 
</td></tr>
<tr><td>The modeler should consider closing the control-flow block by adding a corresponding join gateway in order to remove the duplicated end event. It also makes the model easier to read.</td></tr>
<tr><td> 
</td></tr>
</table>

· Duplicate intermediate events should be removed from the model:

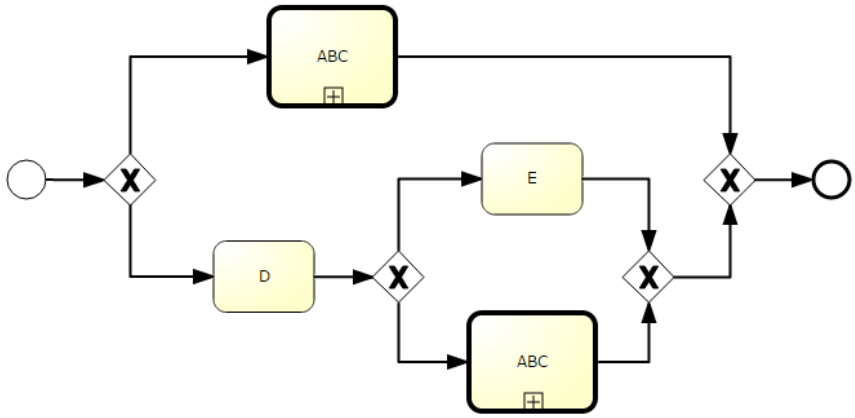
<table>
<tr><td>a) A join is preceded by two events with the same meaning</td><td>b) Simplify model a) by placing the event after the join</td></tr>
<tr><td> 
</td><td> 
</td></tr>
</table>

· **Collapse repeated fragments into a global subprocess.**
This action refers to a subprocess that is not embedded within any process model, and as such can be invoked by other process models within the same process model collection. This activity type is known as 'call activity' in BPMN.
These actions correspond to proposed refactoring patterns: RF4 (Extract Process Fragment), RF5 (Replace Process Fragment by Reference), RF8 (Remove Redundancies) in [3]. Notice that the authors speak about redundancy while the pattern is actually about duplication of process fragments.

a) Business process model with repeated fragment of activities



b) Replacing duplicated fragment with a global subprocess



Associated Metrics:

- **Number of duplicated elements.
- **Number of duplicated fragments in the model.

Problem 3:

- Models contain unnecessary elements.

Discussion:

- When capturing business processes, the resulting process model may contain unnecessary (redundant) elements (e.g. one empty arc between an AND split and an AND join). Redundant elements can be detected with the help of patterns.

Unified Guideline 3:

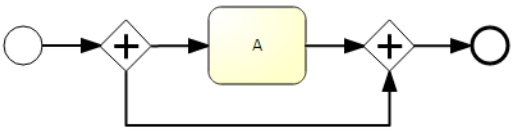
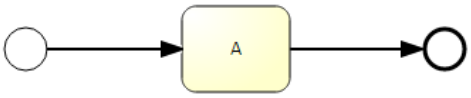
- Avoid unnecessary elements.

Motivation:

- Avoiding unnecessary elements in models amounts to optimizing process models. This will also influence positively the quality of the models as a decrease in size yields better understandability, readability and usability of the models. Larger models are also more difficult to maintain [3, 5, 6]. Applying this guideline will lead to an improvement of pragmatic and empirical quality.

Action:

- Remove redundant elements (e.g. by means of transformation rules and/or error patterns [9]).

a) Model contains useless gateways	b) Removing useless gateways from the model
	

Associated Metric:

- **Number of useless/redundant elements.

2.2. Number of events

Problem 4:

- High number of events (i.e. number of events is above 7).

Discussion:

- When capturing complex business processes, the resulting process model may have more than the recommended amount of events. Empirical research has demonstrated that models with more than 7 events are not easy to understand [10]. To improve the readability of these "complex process models", it is recommended to simplify them by hiding certain parts within subprocesses. As a result, some details of the process model will be omitted and the readability of the whole process model is improved. The size of the process model is not reduced. The number of events will increase with the creation of new subprocesses with at least one starting and one ending event per subprocess, to explicitly indicate when the subprocess starts and completes. However, each individual subprocess or main process should count less than 7 events.

Unified Guideline 4:

- Avoid models with more than 7 events.

Motivation:

- By decomposing a model, we can distribute events between main process and subprocesses. This will not reduce the total number of events; instead, it will likely increase the total number of events in the expanded process. Yet it will allow better understanding and readability of the model by hiding part of the complexity of the main process model (referred in this case by "number of events"). Applying this guideline will lead to an improvement of pragmatic and empirical quality. This guideline is supported by empirical evidence.

Action:

- Split the number of events among process and subprocesses by introducing subprocesses as recommended in guideline 2.1.

Associated Metric:

- **Total number of events per process level.

Start and end events

Problem 5:

- The model contains multiple start/end events.

Discussion:

- In [6, 11-13] authors recommend to use only one start event. Based on empirical findings of a recent study [5] the use of two start events is still fine. In [14] the authors suggest to restrict the use of multiple start events to improve understandability of the models.Regarding end events, in [15] the authors say that a higher number of end events increases error probability, but not very strongly while, for example, in[14], [6] and [12]authors state that models with only one end event are less error prone and more understandable. Based on empirical findings of the most recent study [5] the use of two end events is fine. Based on practical experience[11]proposes to use only one start event in subprocesses. The author also recommends the use of two end events in subprocesses to distinguish success and failure states. While this seems a reasonable good practice, it lacks empirical fundaments.

Unified Guideline 5:

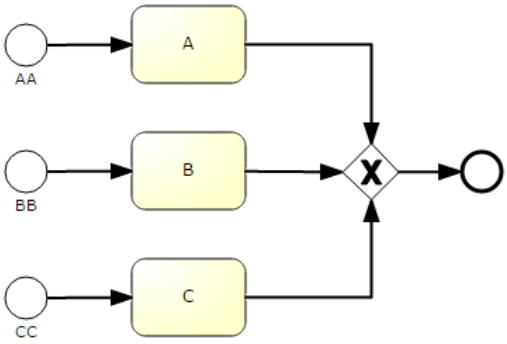
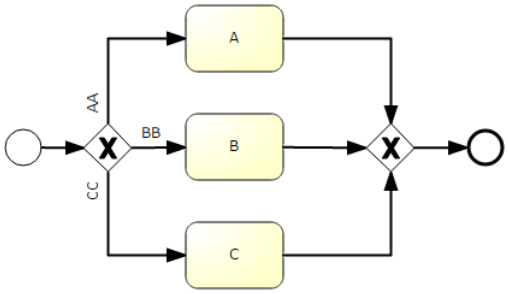
- Use no more than two start/end events in the top process level.
- Use one start event in subprocesses.
- Use two end events to distinguish success and fail states in subprocesses.

Motivation:

- Models with one start/end event are easier to understand [6] (Pragmatic quality)and less error prone [14] (Syntactic, semantic and empiricial quality) than models with multiple start/end events.

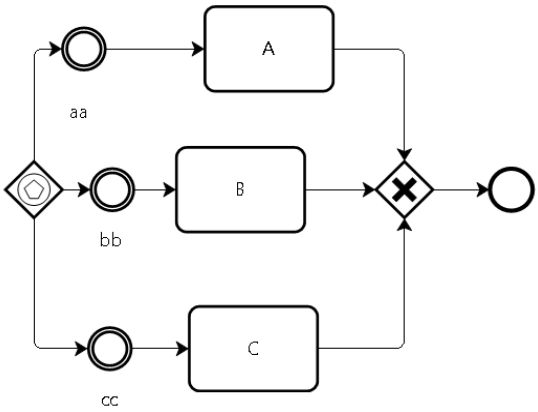
Action:

- **Reduce the number of start events**:
  Replace all start events with only one start event succeeded by an XOR split gateway [12]. Connect this gateway to each activity that was preceded by one of the original start events.

a) Model contains more than two start events	b) Replacement of several start events by only one
	

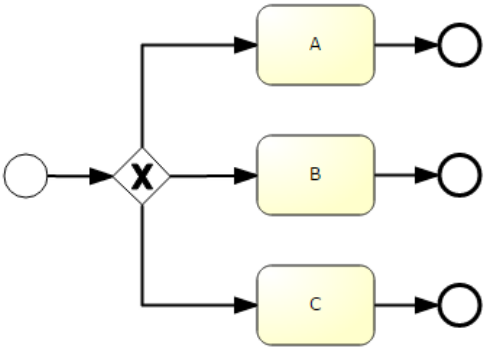
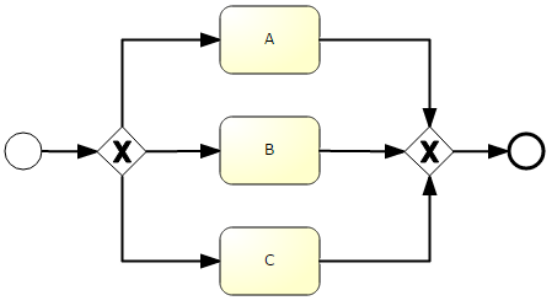
Recently, in BPMN a specific construct was added that allows replacing multiple start events by a start gateway. The above example can be represented as follows using an event-based start gateway:

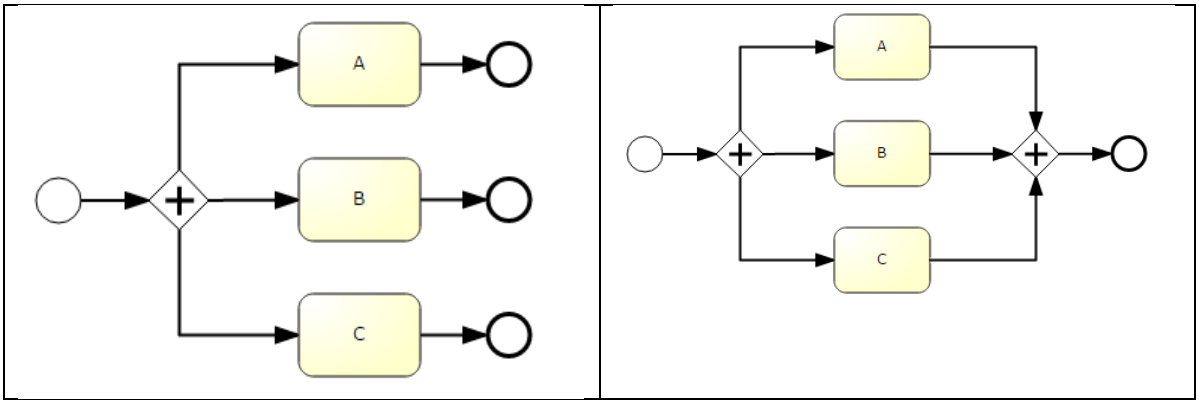
a) Replacement of multiple start events by a start gateway



With this construct, different start events are replaced by one event-based gateway that starts the process, followed by intermediate events or receive tasks. The use of this construct reduces the number of start events, but it does not reduce the number of events overall. It is unclear to what extent the use of this construct makes a model more readable: according toProblem 6, the absence of a start event (because of using an event-based gateway instead) may hamper understandability.

· Reduce the number of end events:
Replace all identical end events with only one end event preceded by a join gateway. Connect the gateway to each activity that was followed by the original end events.

a) Model contains more than two duplicated end events	b) Replacement of duplicated end events by one
	



Associated Metrics:

- **Number of start events per process level
- **Number of end events per process level

Problem 6:

- Absence of start/end events.

Discussion:

- This is one of the "normalization" rules that the BPMN2BPEL tool implementation automatically performs in a preprocessing step to ''normalize" the process model.

Unified Guideline 6:

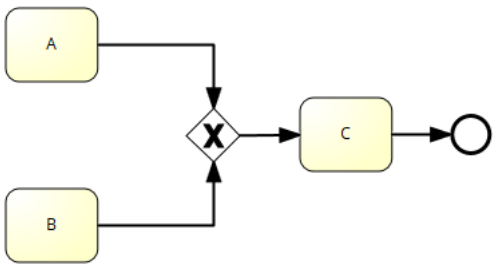
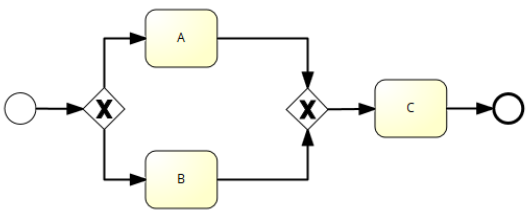
- Do not omit start and end events.

Motivation:

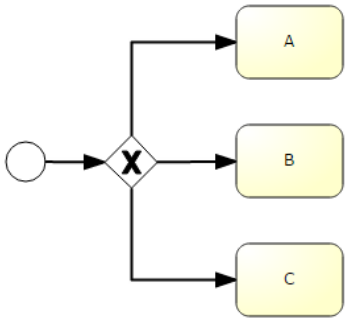
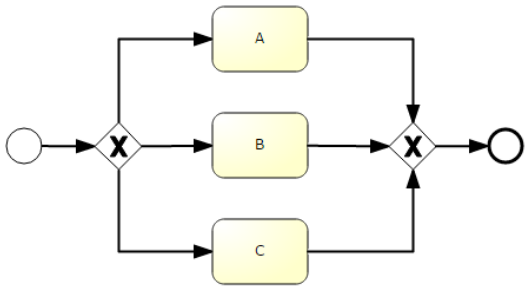
- A model should contain at least one start and one end event to accomplish a better correspondence with the reality (**Semantic quality**) [12] and better understandability (**Pragmatic quality**) of the model. There is empirical support for this guideline.

Action:

- **Create start/end events** using transformation rule TR1 [12].
  Have each activity without incoming flows preceded by a start event (extensible to AND gateways and iOR gateways):

a) Absence of start event in a model	b) Same model with a start event
	

- Have each activity without outgoing flows followed by an end event (extensible to AND gateways and iOR gateways):

a) Absence of outgoing flows from activities	b) Adding outgoing flows to the activities shown in a)
	

Associated Metrics:

- ** Number of start events per process level.
- ** Number of end events per process level.

Number of intermediate events

Problem 7:

- High number of intermediate events.

Discussion:

- When capturing complex business processes, the resulting process model may contain a high number of intermediate events. Empirical research has demonstrated that a higher number of intermediate events increases error probability and reduces understandability of the models. To improve the readability of these "complex process models", it is recommended to simplify them by hiding certain parts within subprocesses and to remove duplicate or unnecessary events (in case they appear in the model). As a result, some details of the process model will be omitted in the higher level process model and the readability of the whole process model is improved.

Unified Guideline 7:

- Avoid high numbers of intermediate events in the process model.

Motivation:

- By decomposing a model, we can distribute intermediate events between process and subprocesses. This will not reduce the total number of events; instead, it will allow better understanding of the model by hiding part of the complexity of process models (referred in this case by "number of events"). A higher number of intermediate events also tends to increase error probability, but not very strongly [15]. This guidelines relies on empirical support.

Action:

- Introduce subprocesses as recommended in Unified Guideline 1 and distribute the intermediate events between process and subprocesses.
- For duplicated and unnecessary intermediate events, apply Unified Guideline 2 and Unified Guideline 3.

Associated Metrics:

- **Number of intermediate events per process level

- **Number of *duplicate* intermediate events per process level
- **Number of *unnecessary* intermediate events per process level

2.3. Number of arcs

Problem 8:

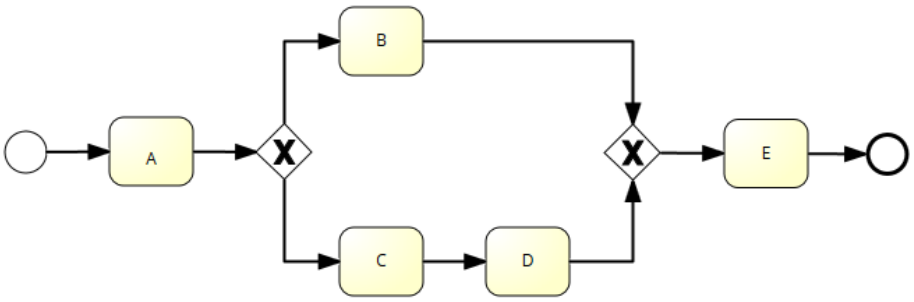
- Model contains more than 34 arcs.

Discussion:

- When capturing complex business processes, the resulting process model may contain a high number of arcs. If these complex models are decomposed, this will increase their understanding by stakeholders. One criterion that can be followed when decomposing the models is "connectedness", as explained in guideline 2.1. When using this criterion the number of arcs is split up between process and subprocesses.

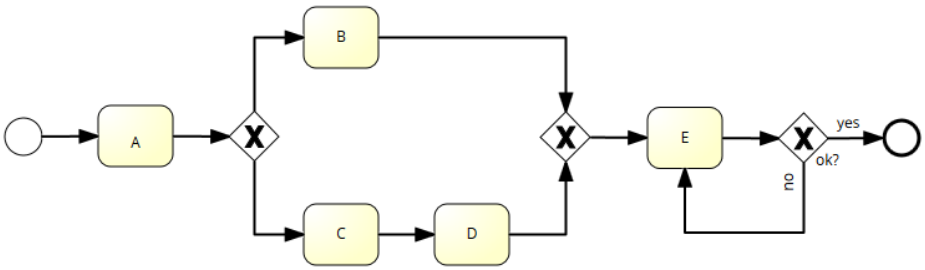
  Several metrics have been proposed that are related to arcs. Some of these metrics include coefficient of connectivity(CNC), connectivity level between activities (CLA), number of arcs (Sa), density, average connector degree (ACD), cross-connectivity metric (CC). From these metrics, we believe that sometimes values obtained through CLA and CNC do are not in line with the principles of representational measurement, i.e. with the expectation that a higher value should corresponds to what is also intuitively understood as a more complex model (see examples further down).

  Example 1: The metric CLA for this model is 5/1, so results in 5. This is a relatively high number, above the threshold of 1.1 suggested in [10], yet, this model is (subjectively)very easy to understand.



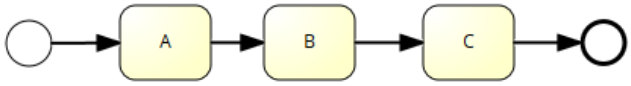
Example 2: CLA for this model is 6/2=3

For this model, CLA is lower than the CLA of the previous example (3<5) while (intuitively) it is more difficult to understand.



These two models provide a counterexample for CLA. It seems that from the perspective of representational measurement theory [16] sometimes CLA is not a good metric to indicate the understandability of the model.

Example 3: for this model we calculate the CNC metric as 4/5 = 0.8. This value is higher than the threshold (0.6) suggested in [10], yet the model is easy to understand..



For all of the above examples, we propose to use number of arcs, density and average connector degree metrics instead of CLA and CNC.

Unified Guideline 8:

- Avoid models with more than 34 arcs.

Motivation:

- A higher number of arcs increases error probability and impedes understandability of the model. This guideline is supported by empirical evidence.

Action:

- Introduce subprocesses as recommended in guideline 1 or, alternatively, use connectedness criterion to split up the number of arcs between process and subprocesses as explained in the discussion of the current guideline.

Associated Metrics:

- Number of arcs per process level (S_A)
- Density per process level
- Average connector degree per process level (ACD)

2.4. Number of gateways

Problem 9:

- Model contains more than 12 gateways.

Discussion:

- When capturing business processes, the resulting process model may contain more than 12 gateways. An increase of the number of gateways implies an increase in error probability, and also an increase in effort needed to understand the model. According to the recommendations based on practical experience formulated in [11], the number of gateways can be reduced as gateways that join parallel paths immediately preceding and end event are unnecessary. We however do not follow this recommendation since it violates the structuredness principle (see also discussion of Unified Guideline 20).

Unified Guideline 9:

- Avoid models with more than 12 gateways.

Motivation:

- A lower number of gateways will improve pragmatic and empirical quality of the models. It also decreases error probability of the models. There is empirical evidence for this guideline.

Action:

- Decompose the model (see Unified Guideline 1).

Associated Metrics:

- Total Number of Gateways (TNG)
- Gateway complexity indicator (GCI)
- Control Flow Complexity (CFC)

2.5. Number of activities

Problem 10:

- High number of activities.

Discussion:

- When capturing business processes, the resulting process model may contain high number of activities. An increase of the total number of activities (TNA) implies an increase in error probability. TNA also correlates positively with modifiability of the models: a lower number of activities makes models more maintainable.

Unified Guideline 10:

- Minimize the number of activities.

Motivation:

- With an increasing number of activities process models become more difficult to understand and maintain (i.e. a decrease of empirical and pragmatic quality). This guideline is supported by empirical evidence.

Action:

- Decompose the model as proposed in Unified Guideline 1.
- Remove duplicate activities as proposed in Unified Guideline 2.

Associated Metric:

- Total Number of Activities per process level.

2.6. Elements complexity

2.6.1. Gateways complexity

Problem 11:

- High number of routing paths per gateway

Discussion:

- When reality is complex, models become complex too. We can find an evidence of this when average connector degree (ACD) and maximum connector degree (MCD) of gateways is above 3 (i.e. join gateways in the model have more than 3 inputs, and split gateways have more than 3 outputs). In the literature, it is possible to find different threshold values for the gateway metrics ACD and MCD. We adhere to the threshold value given in [5] because this is the most recent experimental research. Notice also the difference between the ACD and NSFG (number of sequence flows from gateways) metrics. While ACD refers to incoming and outgoing arcs of gateways, NSFG refers only to the number of outgoing sequence flows from gateways [10].
  One possible solution to reduce ACD, MCD, NSFG and CFC (control flow complexity) of the models could be to split a gateway into several gateways of the same type in order to decrease the number

incoming/outgoing flows. This solution, however, as a consequence has an increment of the number of gateways of the model (TNG) and may also increase the nesting depth of a model (see example with the next guideline). New empirical research is needed to investigate the optimal trade-off between reducing the number of gateways versus the complexity of individual gateways in terms of model quality dimensions such as understandability, errors and maintainability.

Unified Guideline 11:

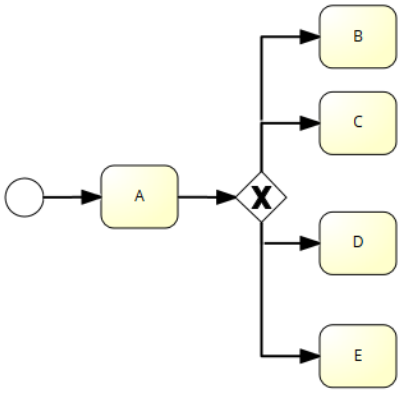
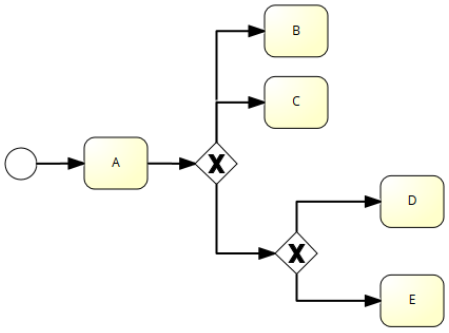
- Use no more than 3 routing paths per gateway.

Motivation:

- The higher the degree of an element in the process model, i.e. the number of input and output arcs together, the harder it becomes to understand the model. As shown in [14] there is a strong correlation between the number of modeling errors and the average or maximum degree of elements in a model. This guideline is supported by empirical evidence.

Action:

- Reduce the routing paths per gateways.

a) Gateway with a high number of outgoing arcs	b) Same model where the gateways has be split to reduce the number of outgoing arcs
	

Associated Metrics:

- Average connector degree (ACD)
- Maximum connector degree (MCD)
- Number of sequence flow from gateways (NSFG)
- Control flow complexity for AND splits (CFCand split)
- Control flow complexity for inclusive OR splits (CFCor split)
- Control flow complexity for XOR splits (CFCxor split)

Problem 12:

- Split/join gateways have more than one incoming and outgoing flows (i.e. two behaviors on the same gateway).

Discussion:

- Models with this characteristic have split and join semantic in the same gateway. This brings an increment of the average connector degree (ACD) (average connector degree) and of the maximum connector degree (MCD) of the model. Moreover, avoiding join behavior from split behavior in gateways is also stated as a notion of well-formedness of BPMN process in [13]. When decomposing

gateway with unified semantic into split/join gateways, the total number of gateways is incremented.

Unified Guideline 12:

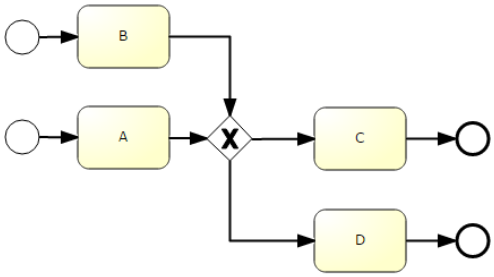
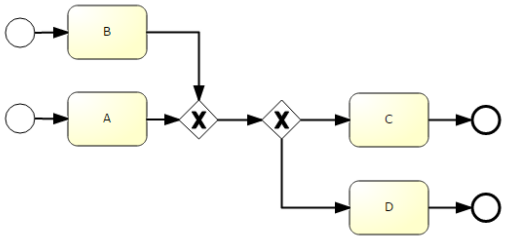
- Do not combine multiple inputs and multiple outputs on the same gateway.

Motivation:

- Based on practical experience, [11] recommends to avoid use of multiple inputs and multiple outputs on the same gateway to increase pragmatic quality of the model

Action:

- Separate join behavior from split behavior by decomposing a gateway with multiple incoming and multiple outgoing flows into a join gateway followed by a split gateway, where the join gateway has all the incoming flow and the split gateway has all the outgoing flows.

a) Gateway with join and split behaviours	b) Same model with join gateway and split gateway (behavior separated)
	

Associated Metric:

- **Number of gateways with multiple inputs and outputs on the same time (split/join behavior).

2.6.2. Events complexity

Problem 13:

- There are more than 4 outgoing sequence flows from an event

Discussion:

- The number of outgoing sequence flows from events could be bigger than 4. This is negatively correlated with understandability. One possible way to avoid this is to use a split-parallel gateway after the event. This amounts to replacing an implicit split by a gateway, but moves the problem to the gateway: the result will be a gateway with a higher than recommended degree.

Unified Guideline 13:

- Do not use more than 4 outgoing sequence flows from events

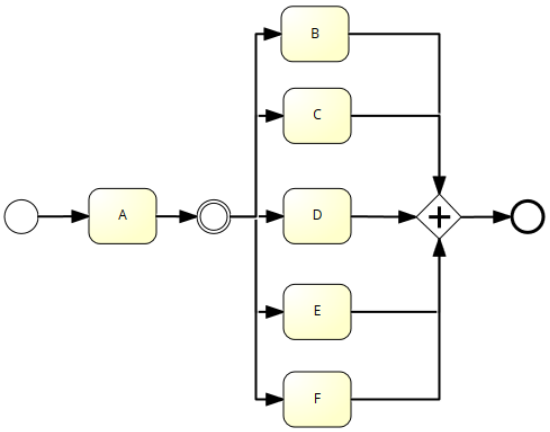
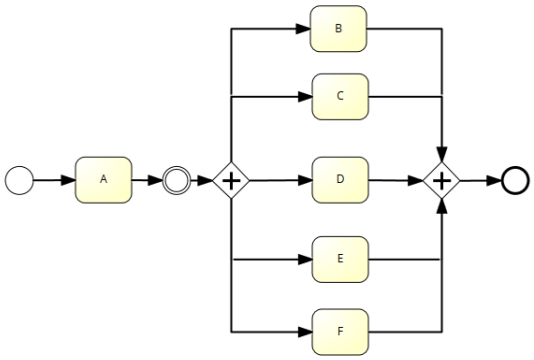
Motivation:

- Number of sequence flows from events correlates with understandability of the models. This guideline relies on empirical support.

Action:

- Reduce the number of outgoing sequence flow from events

Multiple outgoing flows indicate parallelism. As this must be modeled if it exists in real domain, use a gateway to model it explicitly. The resulting model may be more complex with an increment of the number of gateways:

a) High number of outgoing flows from event (parallelism)	b) Using AND split gateway
	

Associated Metric:

- Number of sequence flows from events (NSFE)

3. Morphology

3.1. Depth (nesting)

Problem 14:

- Model has deeply nested structured blocks.

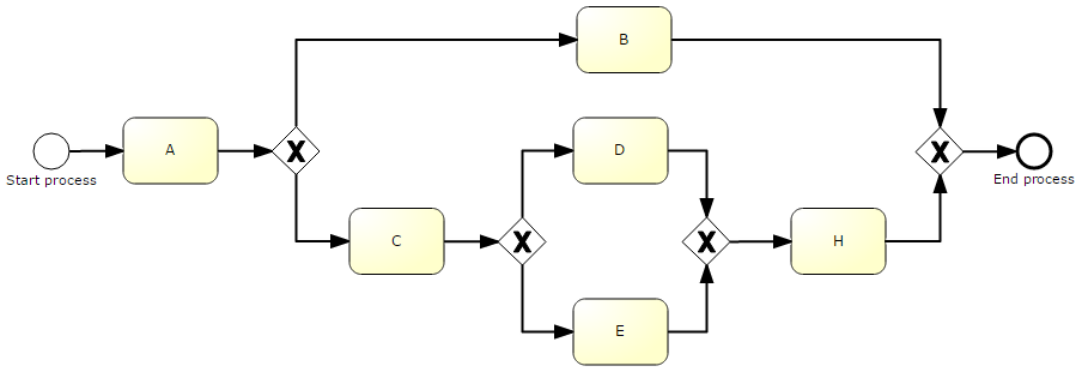
Discussion:

- Sometimes, when modeling reality, multiple choices need to be taken before performing certain activities (i.e. how many routing constructs in the BPM have to be understood in order to reason about the execution of an activity). Such models have deeply nesting structured blocks and are difficult to understand and more error prone, as state in [10, 14]. If nesting depth correctly corresponds to reality, then it should be consistently modeled for high semantic quality.
- In programming "nesting depth" can for example be reduced by replacing nested 'if-then-else' statements with a 'case' statement. The example below gives a corresponding example for BPMN
- We propose the use of another metric to obtain the real idea of the real depth of the process model (taking into account the real maximum nesting depth). We call it Maximum nesting depth:

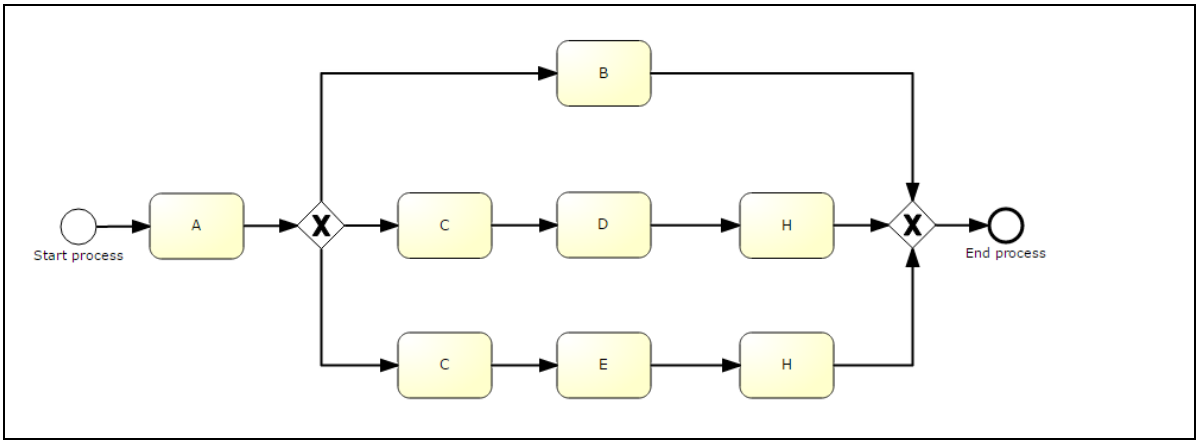
$$\lambda''(n) = Maximum\ of\ indepth(n)\ and\ outdepth(n)$$

Depth metric [17] calculates depth as the maximum depth of all nodes. The problem with this metric is that for each node it takes into account the minimum depth (i.e. $\lambda(n) = Minimum\ of\ \lambda_{in}(n) and\ \lambda_{out}(n)$) which does not give the real idea of the maximum deep of the overall model.

a) Business process model with two nested gateways

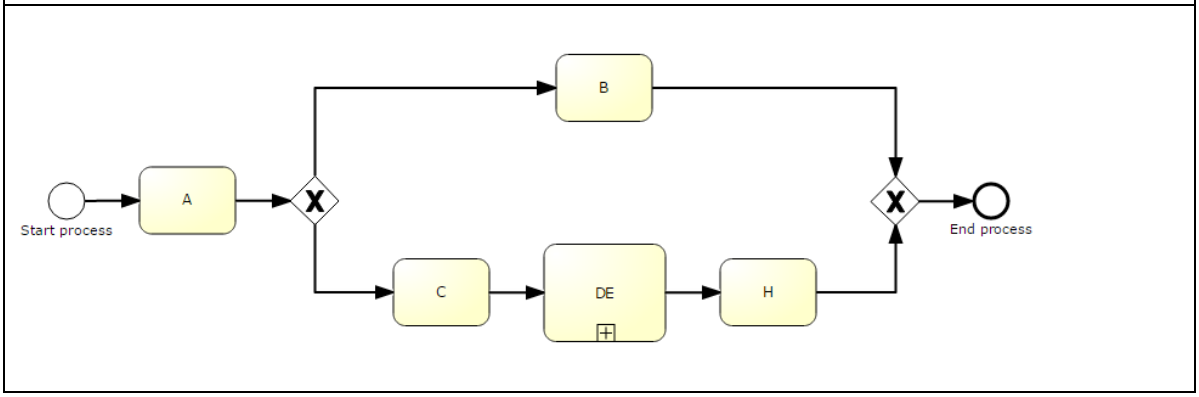


b) Reducing nesting



While the simplified structure may yield a better understandability, this solution has 2 disadvantages. First, it introduces duplicate elements: in the bottom diagram, C and H are duplicated. Second, gateway complexity is increased, which is in contradiction with Unified Guideline 11. On the positive side, the total number of gateways is reduced (see Unified Guideline 9). Another example is code folding, which is the equivalent of introducing subprocesses. For the above example, the introduction of a subprocess would yield the following diagram:

c) Reducing depth in the high level diagram by introducing subprocess



This solution has the advantage of a lower number of gateways while not increasing gateway complexity, nor duplicating elements.

Unified Guideline 14:

- Avoid deeply nesting structured blocks.

Motivation:

- An increase in depth implies an increase in error probability and less understanding of the model [10, 14]. This guideline is based on empirical support.

Action:

- Introduce subprocesses to hide complexity or flatten subsequent decision gateways into a single or a lower number of gateways.

Associated Metrics:

- Depth
- **Maximum nesting depth

3.2. Cyclicity

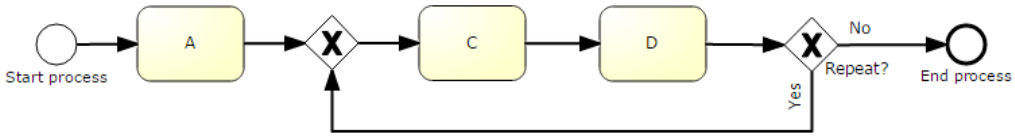
Problem 15:

- Model contains multiple cycles.

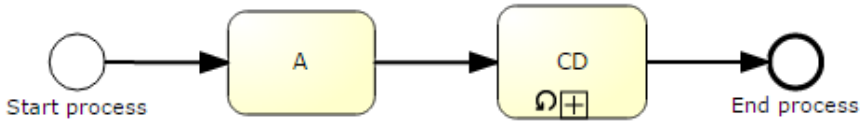
Discussion:

- Cycles are fragments of a model that can be repeated n-times. For example, a person checks an application form for completeness repeatedly until is complete. If we try to avoid cycles when modeling this behavior, the model will have duplicated tasks. Also, if is not known in advance how many times the cycle should be repeated, there is no possible way to represent it without the use of cycles or loop activities.

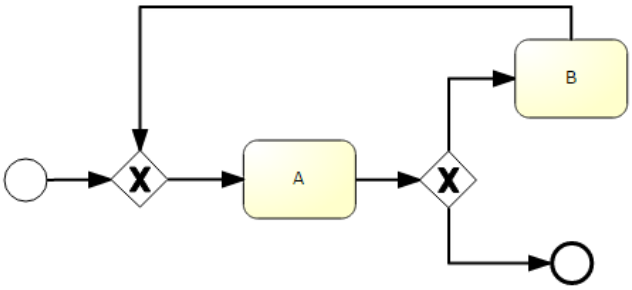
a) Cycle



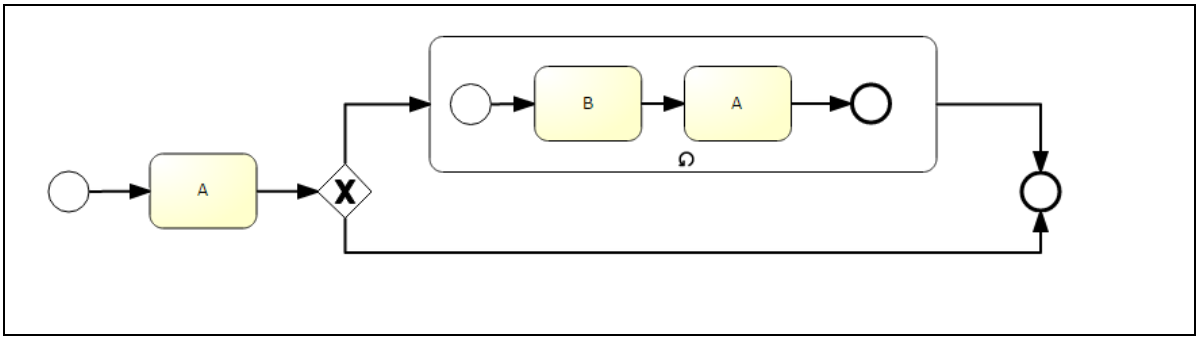
b) Cycle using loop activity



a) Cycle, a second example



b) Using loop activity



- However, it is not always feasible to replace a cycle by a loop activity. This is the case for the unstructured cycles.

Unified Guideline 15:

- Avoid cycles in your process models.
- Avoid unstructured cycles (i.e. cycles with multiple exit points).

Motivation:

- Models with cycles were found to be inherently unstructured [18] and are more difficult to understand and to maintain [5].

Action:

- Use a loop activity to model structured cycles (i.e. cycles with only one exit point).

Associated Metrics:

- Cyclicity (CYC)
- Separability
- Sequentiality
- **Number of repetition blocks in the model

Problem 16:

- Badly formed cycles: the backward connection of a loop construct doesn't begin in an XOR split or doesn't lead back to an XOR join

Discussion:

- To create well-formed cycles the backward connection of a loop construct should begin in an XOR split and should lead back to an XOR join. It is also possible to use an inclusive OR split in the beginning of a cycle and lead back to an XOR join. Nevertheless, we recommend the use of XOR in both places (at the beginning and at the end of the cycle) since this decreases error probability while it respects the structuredness principle (see Unified Guideline 19).

Unified Guideline 16:

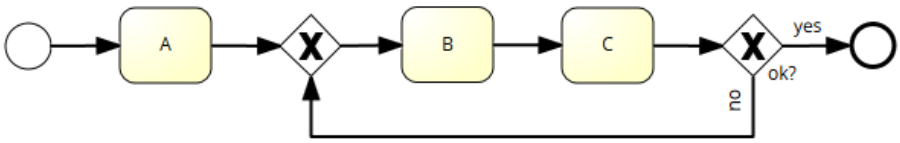
- When modeling cycles, the backward connection should begin in an XOR split and lead back to an XOR join.

Motivation:

- Other gateways combinations could lead to deadlock or livelock situations (i.e. this guideline contributes to semantic quality of the model). This guideline relies in empirical fundaments.

Action:

- Correct the backward and forward connections
  Replace any other combination by this one:



Associated Metric:

- **Number of gateways different from XOR at the beginning/end of repetition blocks.

Problem 17:

- Multiple exit points per cycle

Discussion:

- Cycles with multiple exit points were found to be inherently unstructured.

Unified Guideline 17:

- Avoid multiple exit points per cycle.

Motivation:

- Unstructured cycles may produce unstructuredness in the process models. Unstructuredness also causes high error probability, less understandability and less modifiability of the models as explained in Unified Guideline 19 (i.e. semantic, pragmatic and empirical quality). There is empirical support for this guideline.

Action:

- Reduce number of exit points per cycle if possible
- Check whether there is no logical error in the model and whether is possible to remove the additional exit points.

Associated Metrics:

- **Number of unstructured cycles
- **Average number of exit points per cycle

3.3. Parallelism

Problem 18:

- High level of parallelism (the sum of the output-degrees of AND and iOR gateways should be at most 8)

Discussion:

- The total output degree of AND/iOR gateway should not be above 8. We acknowledge that parallelism cannot always be reduced: *If it exists in real domain, then it should not be avoided.*

Unified Guideline18:

- Avoid high level of parallelism in your process models.

Motivation:

- An increase in the level of concurrency implies an increase in error probability of the overall model and lower understandability and maintainability of the model.

Action:

- If the token split is above the threshold, consider decomposing the model to make it less complex.
- Remove *unnecesary* paralellism by applying the patterns presented in Unified Guideline 3 [9].

Associated Metric:

- **Token split per process level

Problem 19:

- Bad parallelism: Parallel paths don't reach end events or do not syncrhonize.

Discussion:

- If the activated parallel paths don't reach end events and are not synchronized then the process model will not end properly. This means the model contains errors.

Unified Guideline 19:

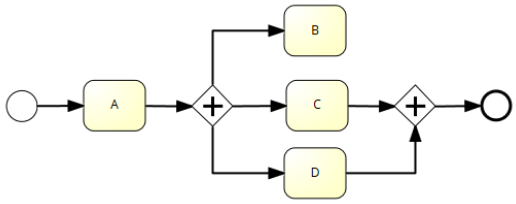
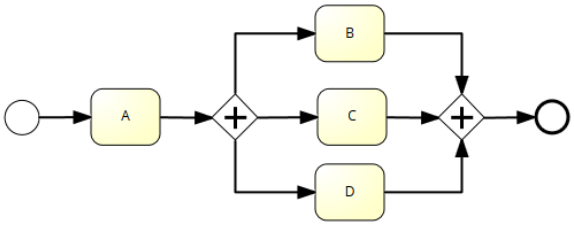
- Each parallel path must reach an end event or must be syncrhonized.

Motivation:

- To increase syntactic and semantic quality.

Action:

- Correct the parallelism:

a) Process model with hanging tasks	b) All activated parallel paths must reach end events or be synchronized
	

Associated metric:

- **Number of unsynchronized paths in the process models

3.4. Structuredness

Problem 20:

- High level of unstructuredness (i.e. the sum of mismatches for each connector type is above 4.5)

Discussion:

The following two statements are formulated in [11]. They are however in contradiction with the structuredness principle.

- '*An XOR gateway used as a merge is the same as no gateway at all*'. While it is true that the semantics of both constructs is the same, this statement seems to suggest that the two are equivalent in terms of model quality. Not using a gateway at all to merge paths that have been previously split by means of an XOR gateway however contradicts the principle that gateways should come in matching pairs.
- '*We do not need to use a gateway to join parallel paths into a None end event*'. The statement proposes to give preference over implicit joins rather than an explicit join by means of an AND

gateway. When the parallel paths originate from a parallel AND split, this contradicts the principle that gateways should come in matching pairs.

Since the structuredness principle is based on empirical fundaments, we propose to follow it and to give preference to include gateways and to avoid implicit merges and splits.

In [19] authors study the problem of automatically transforming unstructured process models into equivalent well-structured models. The method has been implemented as a tool in Signavio.com. See *action* for pragmatic hints to increase structuredness.

Unified Guideline 20:

- Every split gateway should matches a respective join gateway of the same type.

Motivation:

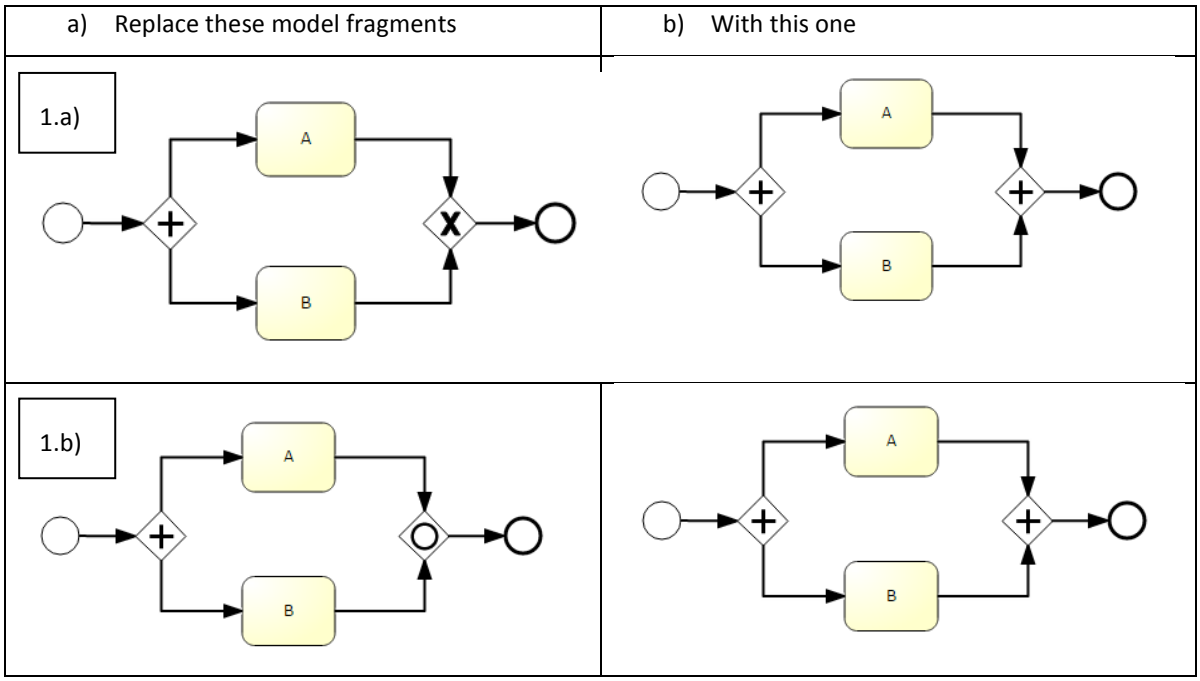
- Structured models can be seen as formulas with balanced brackets, i.e., every opening bracket has a corresponding closing bracket of the same type. Unstructured models are not only more likely to include errors [5, 14], people also tend to understand them less easily [18] and they are more difficult to modify. This guideline is supported by empirical evidence.

Action:

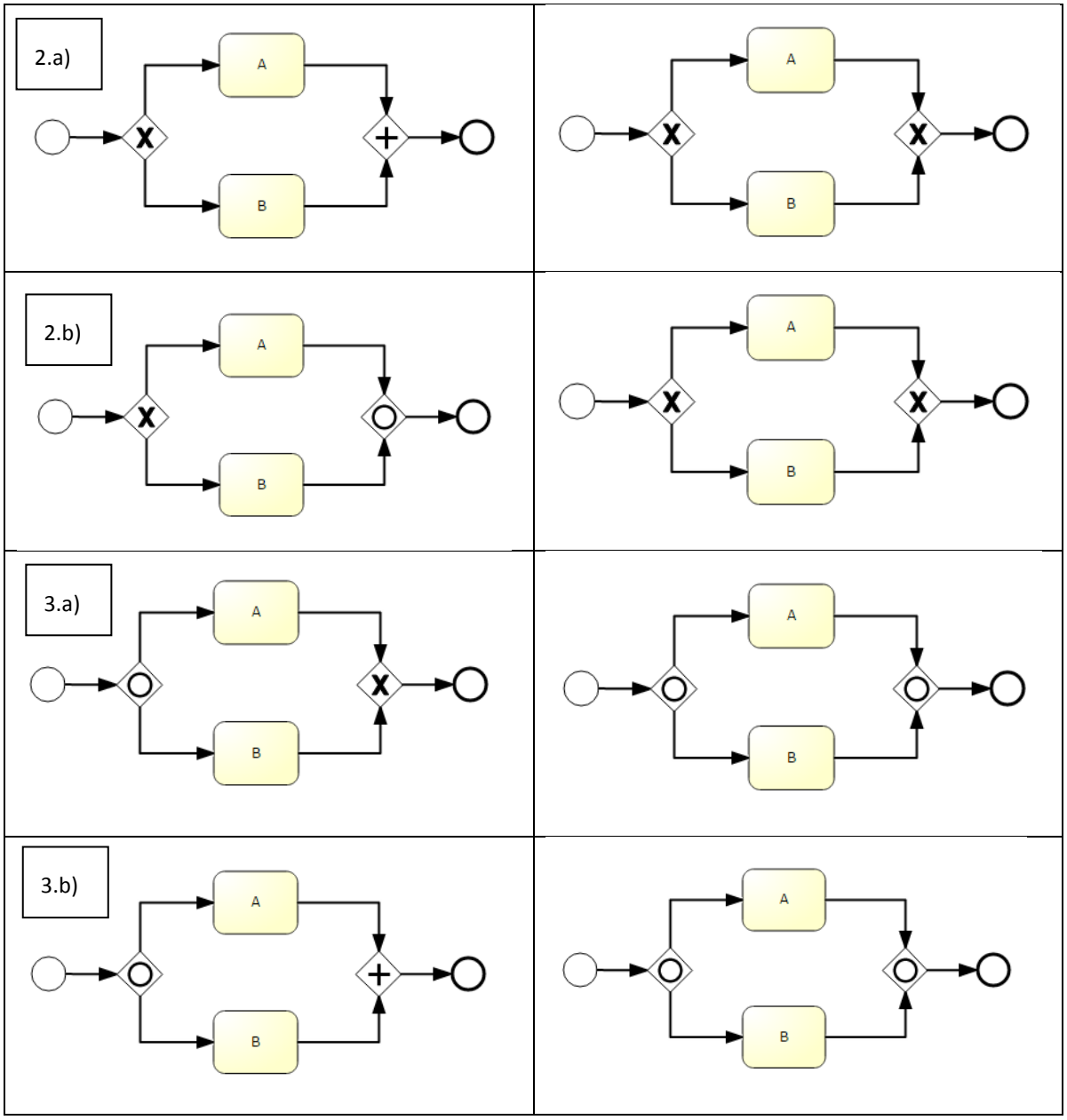
Pragmatic actions to improve structuredness:

- Introduce a join gateway per every split gateway (of the same type) that doesn't match .
- Introduce a split gateway per every unmatched join gateway of the same type.
- Avoid implicit joins and splits.

If the model has unmatched pairs of gateways, make them corresponding using the following rules[1]:



[1]The rules point to reduction of gateway heterogeneity, improvement of soundness of the model (removal of deadlocks and lack of synchronization), and recommendations of inclusive OR removal from the models into account (http://academic.signavio.com/)



Associated Metrics:

- Connector mismatch (MM)
- Structuredness

3.5. Diameter (longest path)

Problem 21:

- Model contains a long path from start node to end node.

Discussion:

- This guideline originates from research on metrics. A limitation of this guideline is that not all long models are problematic: there are large models in terms of diameter that are unlikely to have errors, e.g. if the model is purely sequential.

Unified Guideline 21:

- Keep the path from a start node to the end as short as possible.

Motivation:

- This is a structural aspect that affects the readability of a process model [4]. A larger business process model in terms of *diameter* would be more likely to contain errors than a small one [20]. This guideline is supported by empirical evidence.

Action:

- Decompose the model to hide complexity

Associated Metric:

- Diameter (diam)

3.6. iOR Gateways

Problem 22:

- High gateway diversity

Discussion:

- Heterogeneity or gateway diversity has the worst values when the three gateway types appear in the process model. An increase in gateway diversity implies an increase in error probability. Besides, models with high values of gateway diversity are more difficult to understand and to modify. Nevertheless, if processes are complex in reality and their models demand the use of all three different gateway types, the gateway diversity cannot be reduced.

Unified Guideline 22:

- Minimize gateway diversity.

Motivation:

- Reducing gateway diversity improves empirical and pragmatic quality of the models. In addition, it decreases their error probability. This guideline relies on empirical support.

Action:

- Reduce the diversity of the gateways, if possible
  This can be done through model analysis and assessing the possibility of removing one of the three different gateway types from the model (e.g., cases 1.b) and 2.b) from Unified Guideline 20).
  By improving the structuredness of the model, we can decrease gateway heterogeneity value.

Associated Metric:

- Gateway heterogeneity (GH)

Problem 23:

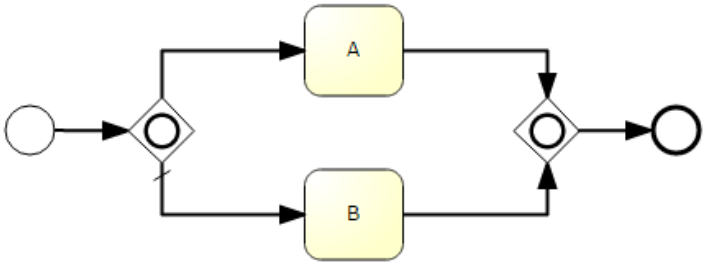
- Existence of inclusive OR gateways in the process models.

Discussion:

- Inclusive OR splits activate one, several, or all subsequent branches based on conditions. They can result in synchronization problems when several branches are executed in parallel and are then merged through an XOR join [21]. They can produce a deadlock if they are synchronized with an AND join gateway [5]. They need to be synchronized with iOR join elements, which are difficult to implement in the general case [17, 22]. In contrast, iOR joins can be used with all kinds of split gateways, but it is better minimize their usage to decrease gateway heterogeneity as much as possible [9].
- Inclusive OR gateways cannot always be avoided. To model situations where a decision may lead to one or more options being taken at the same time, we need to use an inclusive OR split gateway.

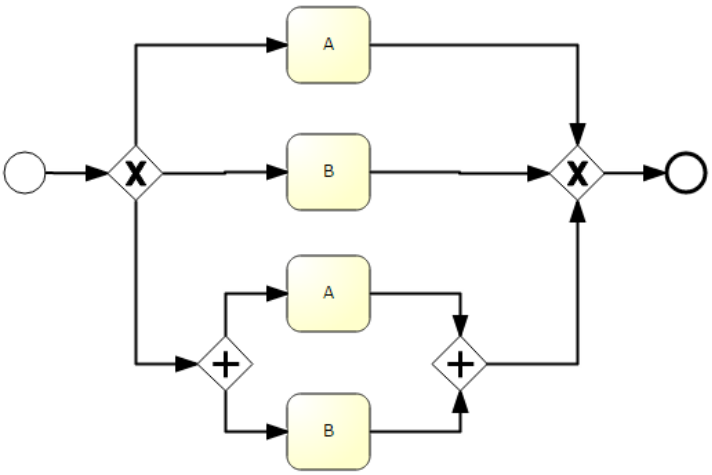
Example: Let´s assume there is a process with two possible tasks: A and B. When the process starts, A and B can each be done on its own, or both tasks can be performed. Afterwards, the process finalizes.

Modeling an inclusive decision with an iOR gateway.



Next figures shows three possible solutions to avoid the use of inclusive OR gateway, two of which have been proposed in [4]:

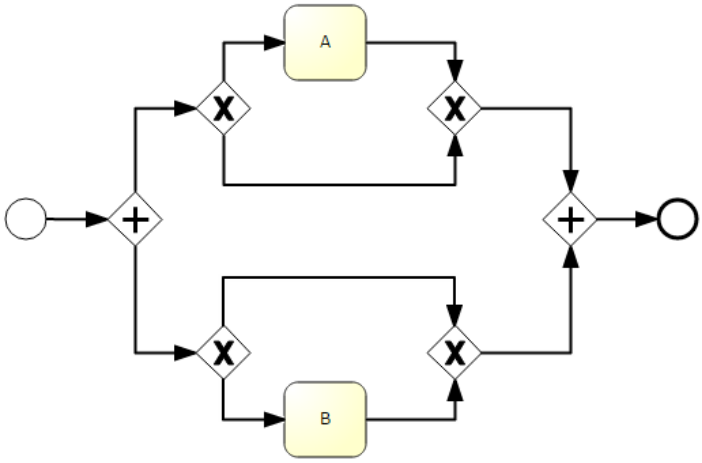
a) Modeling an inclusive decision, first trial



While this model captures the scenario correctly, the resulting diagram is somewhat complicated, since we need to duplicate the two activities A and B twice. In addition, if we had more than two activities, the number of duplicated activities would increase. For this reason, this solution is not scalable.

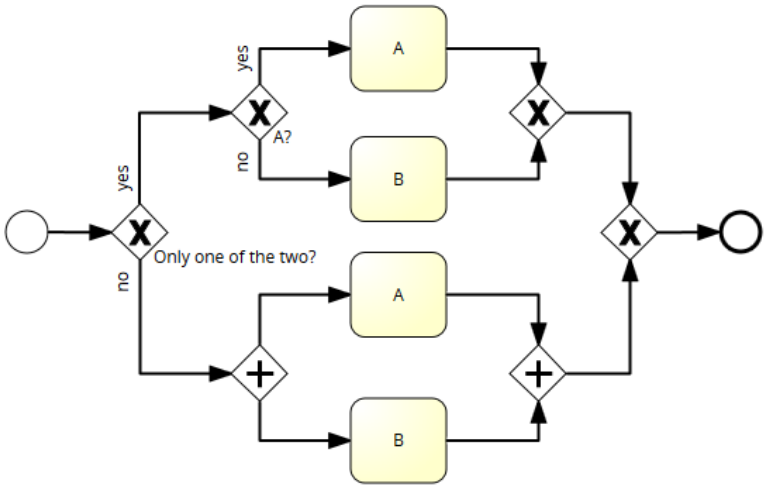
b) Modeling an inclusive decision, second trial.

A second possible way to model an inclusive or, is to make use of a parallel split followed by a choice between performing a task or skipping the task.



What is the problem with this second solution? The example scenario allows three cases: to perform A only, to perform B only, to perform both. However also the scenario where neither A or B is performed is allowed. Despite being more compact than the first solution, this solution is wrong.

c) Modeling an inclusive decision, third trial



The third refactoring presents a correct solution, but the problem with this solution is also the scalability.

Unified Guideline 23:

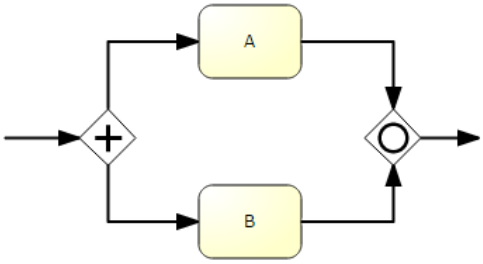
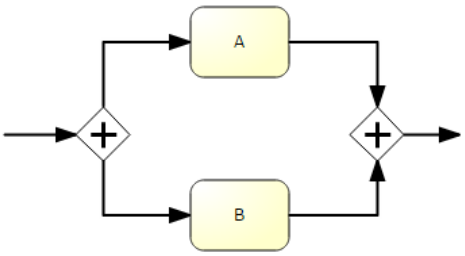
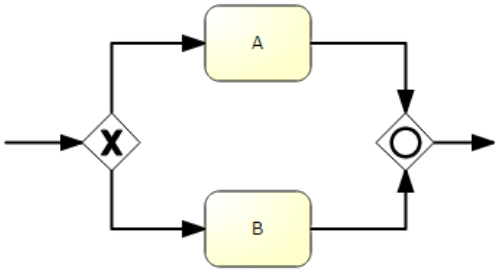
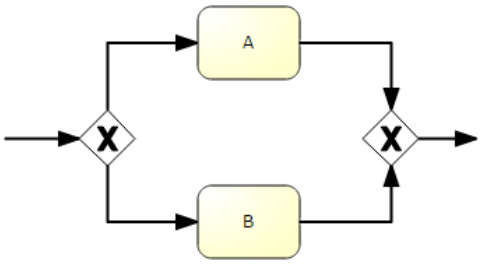
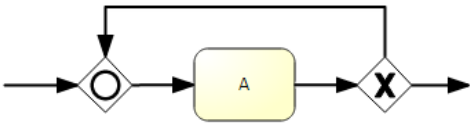
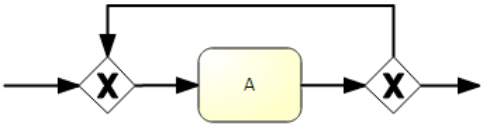
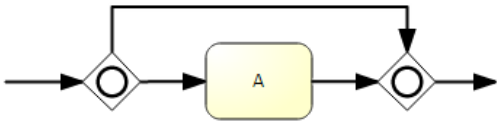
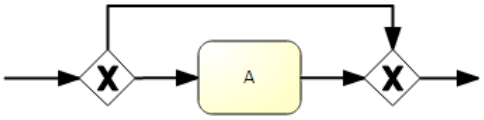
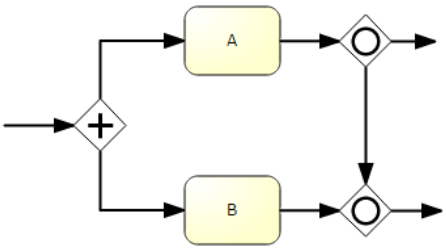
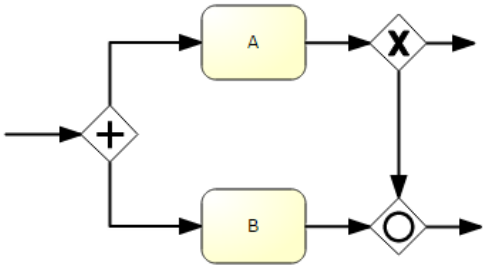
- Avoid the use of inclusive OR gateways.

Motivation:

- To improve correctness and semantic quality of the model. This guideline is supported by empirical evidence.

Action:

Reduce number of iOR gateways if possible by replacing the iOR gateway as shown in the followings patterns [9]:

a) Replace this model fragment	b) By this one
	
	
	
	
	

Associated Metrics:

- Number of inclusive OR splits (iOR Splits)
- Number of inclusive OR joins (iOR Joins)

3.7. General complexity

Problem 24:

- High complexity in the model

Discussion:

- Sometimes reality demands the construction of very complex business process models. However, there are some cases when people add complexity purposeless, and it should be avoided since it makes the model more difficult to understand, to maintain and yields a higher probability of error.

Unified Guideline 24:

- Select the less complex alternative when modeling.

Motivation:

- It has been demonstrated in different research works that more complex models are more error prone (Syntactic Q.), less understandable (Pragmatic Q.) and more difficult to modify (Empirical Q.).

Action:

- Reduce general complexity of the model if possible by applying some patterns given in other guidelines

Associated Metrics:

- Structuredness
- Coefficient of connectivity (CNC)
- Cyclicity (CYC)
- Separability
- Connector heterogeneity (CH)
- Connector mismatch (MM)
- Diameter (diam)
- Density
- Average connector degree (ACD)
- Cross-Connectivity (CC)
- Number of nodes (SN)
- Number of events (SE)
- Number of functions (SF)
- Number of connectors (SC)
- Number of arcs (SA)
- Maximum connector degree (MCD)
- Sequentiality
- Depth
- Control flow complexity (CFC)
- Token splits (TS)
- Number of activities and control-flow elements in a process (NOAC)
- Number of activities, joins, and splits in a process (NOAJS)
- Control Flow Complexity (CFC)

3.8. Modularity

Problem 25:

- Model lacks modularity

Discussion:

- Unnecessary information and high complexity should be hidden from the readers to increase understanding and maintainability of the models. Modularity guidelines refer to different quality dimensions. They can be used to enhance presentation (e.g. hide details, see Unified Guideline 1) of the higher level models without changing the semantics of the (expanded) model. Modularity should be cautiously introduced in the model. The modeler should decide whether introduce modularity or not in order to foster understanding of the model.
- According to [23], both selection criteria (i.e. block-structuredness and connectness) are difficult to understand by novice business modelers. However, expert modelers could follow these criteria to decompose along with their domain knowledge. Results from [24] suggest that there is potential for supporting designers in modularizing their process models. The connectness criterion suffers from another problem, which is that it can lead to producing a large number of very small subprocesses. Since inclusion of many small process models is bad for maintenance [3], subprocesses containing between 1–5 nodes can be merged.
- Additionally, subprocesses are used to create a scope for exception handling that applies to a group of Activities.

Unified Guideline 25:

- Fit each process level on one page by decomposing models with more than 31 nodes (see Unified Guideline 1)**.**
- Avoid decompositions into small subprocesses with less than 5 activities.

Motivation:

- Modularity helps understanding of process models [7, 8, 24]. It icreases pragmatic and empirical quality.

Action:

- Apply block-structuredness or connectedness criteria to decompose models.

- Block- structuredness criterion:
  Create subrocesses from those fragments of a model that are components with a single input and a single output control flow arc.

- Connectedness criterion:
  Create subprocesses from those fragments of a model of which the nodes are more strongly connected by arcs to each other than the nodes out-side this collection.

Associated Metric:

- **Number of activities per process level

4. Presentation

4.1. Layout guidelines

Problem 26:

Model is not readable because of suboptimal layout.

Discussion:

- Layout mainly concerns readability of the models. Graphical constructs and their structural layout influence the way a process model is understood by end users. A model with a "good" layout should be easier to read. Some of the proposed guidelines are harder to meet given the definition of the BPMN notation. For example, the proposal to use colors for matching gateways or to use iconic representations for the different identified verb classes is not possible when one has to stick to the defined BPMN symbols. However, in the latter case, we can formulate the advice to at least make maximal use of the task types proposed by BPMN. A dedicated discussion of automatic layouting of process models and their benefit to comprehension is missing so far [25].

Unified Guideline 26:

- General: Keep the diagram as neat and consistently organized as possible by following the following list of advices *(organized according to the number of occurrences in the literature):*

1. Minimize the number of crossing lines.
2. Minimize the number of overlapping (connection) elements (Nodes should not overlap edges or other nodes.).
3. Minimize the bends in connecting elements.
4. Maximize the number of orthogonally drawn connecting objects.
5. Make your models long and thin (instead of square): maximize the number of connecting objects respecting workflow direction.
6. Place elements as symmetric as possible.
7. Minimize the drawing area.
8. Place related elements close to each other.
9. Adapt the size of objects such that elements have enough space.
10. Consider the use of partitions, e.g. pools and swimlanes.
11. Specify task types, especially user (human task) and service (automated task).
12. Use a uniform style for flow layout.

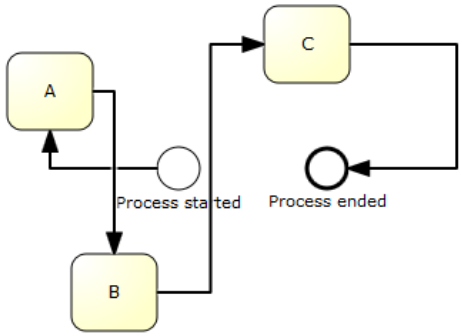
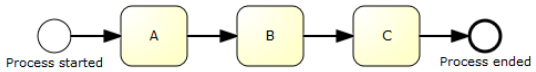
Motivation:

- To improve empirical quality of the models.

Action:

- Apply the guidelines to accomplish a good visual representation of the model.

- Example:

a) Model with bad layout	b) Correct layout for model shown in a)
	

Associated Metrics:

The following are some metrics related to the visual representation of the process models **:

- Size of the drawing Area (i.e. pool area b*h)
- Number of bends
- Number of overlappings between nodes or between nodes and edges
- Number of non-orthogonally drawn connecting objects

4.2. Label style

Problem 27:

Labels are not correct:

- Labels do not follow verb-object style
- Labels are long
- Pools label is different from the process
- Timer events are not labeled with the duration or date/time parameter
- Gateways are not labeled
- Black box is not labeled with the participant name
- There are other constructions than Send/Receive task types that are labeled as Send or Receive

Discussion:

- It has been demonstrated through empirical studies that verb-object activity labels contributes to improved model quality in terms of less ambiguity and higher usefulness [26, 27]. Length of activity labels is negatively related to understandability of the models. Shorter labels contribute to a less complex model, which contributes to a better understanding.

Unified Guideline 27:

Labeling activities:

- Use verb-object activity labels
- Use short labels
- *Reserve the keywords Send and Receive in task names for Send and Receive task types*

Labeling events:

- *Label a Timer event with the duration or date/time parameter*

Labeling gateways:

- *Label gateways and all the sequence flows out*

Labeling pools:

- *Label pools with the name of the process*
- *The black box pool label should name the participant*

Motivation:

- To improve empirical and pragmatic quality of the models.

Action:

Analyze labels in the activities of theprocess model and replace them as needed to acomplish good empirical quality of the model by following the actions:

For activities:

- Replace labels as needed to acomplish verb-object style. Also, analyze how to make them shorter.

For events:

- Add the corresponding duration or time parameter to every timer event of the process model as needed.

For gateways:

- Add labels to every gateway of the process model if they are absent.

Associated Metrics:

- The following are examples of metrics dedicated to measure labeling style of the process models **:
- Number of non-verb-object activity labels in the model.
- Maximum activity label length
- Average label length (total activity label length/number of activities)
- Total activity label length

Appendix

No.	Original Guidelines	Ref.	Type of guideline	Empirical Evidence	Type of Quality (CMQF)	Threshold	Associated Measure
Number of Elements							
	Do not use more than 31 elements	[5]	Guideline	Yes	Error probability	31	Number of nodes (SN)
	Number of nodes is below 31 the model is very efficient in understandability tasks. If number of nodes is below 37 the model is considered efficient.	[10]	Threshold	Yes	Pragmatic Q.	31/37	Number of Nodes (SN)
	Use as few elements in the model as possible.	[6]	Guideline	Yes (relies on validated sources)	Error probability Pragmatic Q.		
	Model size is the primary factor that impacts model understandability	[29]	Knowledge	Yes	Pragmatic Q.		
	An increase in number of nodes could imply an increase in error probability of the overall model	[30]	Metric	Yes (relies on validated sources)	Error probability		Number of Nodes (SN)
	Larger, real world process models	[15]	Knowledge	Yes	Error probability		

	tend to have more formal flaws (such as e.g. deadlocks or unreachable end states) that smaller models						
	An increase in number of nodes implies an increase in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Number of Nodes (SN)
	An obvious way to improve understandability of a BPM is to reduce the number of its nodes	[9]	*a comment in the paper*	Yes (relies on validated sources)	Pragmatic Q.		
C4	PMS3 Redundancy: Both within a single and across different process models, there may be fragments capturing the same control-flow logic.	[3]	Smell	Yes	Empirical Q.		
	Avoid redundancy in process models	[21, 31]	Anti-patterns	No	Correctness (usability, clarity, comparability and maintainability)		
	Identical Start Events Precede Same Join (Definition: There are two start events S1 and S2 with the same meaning. There is a path p1 from S1 to a join J and a path from S2 to the same join J such that the only common node of p1 and p2 is J.)	[32]	Error Pattern	No	Semantical Q. cal Q.		

Number of Events							
	A model with a total number of events below 7 is easy to understand	[10]	Threshold	Yes	Pragmatic Q.	7	Total number of events (TNE)
	There is a correlation between the times of understandability and total number of events.	[33]	Knowledge	Yes	Pragmatic Q		Total number of events (TNE)
	Total number of events correlates with understandability of the models.	[34]	Knowledge	Yes	Pragmatic Q.		Total number of events (TNE)
	An increase in total number of events implies an increase in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Total number of events (TNE)
Start and end events							
S3	Use no more than 2 start and 2 end events.	[5]	Guideline	Yes	Error Probability	2.5	Number of start events ($\mathbf{S_{E_S}}$) Number of end events ($\mathbf{S_{E_E}}$)
	Use one start and one end event	[6]	Guideline	Yes - (relies on validated sources)	Semantic Q. (Error probability)		
	Restrict the use of multiple starts and ends.	[14]	Knowledge	Yes	Pragmatic Q.		
	A higher number of end events increases error probability, but not	[15]	Knowledge	Yes	Error		Number of end events ($\mathbf{S_{E_E}}$)

	very strongly				probability		
S4	Distinguish success and failure end states in a process with separate end events. Label them appropriately. Use an end event to represent the end of each path of a process or subprocess.	[11]	Best Practice	No	Pragmatic Q.		
	Use only one start event in a process or subprocess	[11]	Best Practice	No			
	TR2: Transform a process that has multiple start (end) events into a process with one start (end) event by replacing all start (end) events with only one start (end) event succeeded (preceded) by an XOR split (XOR join) gateway, and connect this gateway to each activity that was preceded (followed) by one of the original start (end) events.	[12]	Transformation Rule	Yes	Pragmatic Q. Semantic Q.		
	Do not omit start and end events.	[11]	Best practice	No	Pragmatic Q.		
	TR1: Transform a process that does not have a start or an end event into a process that does by preceding each task without incoming flows by a start event and succeeding each task without	[12]	Transformation Rule	Yes	Pragmatic Q. Semantic Q.		

	outgoing flows by an end event.						
	Identical Start Events Precede Same Join (Definition: There are two start events S1 and S2 with the same meaning. There is a path p1 from S1 to a join J and a path from S2 to the same join J such that the only common node of p1 and p2 is J.)	[32]	Error Pattern	No	Semantical Q. cal Q.		
Number of intermediate events							
	A higher number of internal events increase error probability	[15]	Knowledge	Yes	Error probability		Number of internal **events (**E_{int})
	There exists a correlation between the times of understandability and total number of intermediate events.	[33]	Knowledge	Yes	Pragmatic Q		Total number of intermediate events (TNIE)
Number of arcs							
	An increase in number of arcs implies an increase in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Number of arcs (Sa)
	Negative effect of a relatively high number of arcs on a model's understandability.	[29]	Knowledge	Yes	Pragmatic Q.		Density Average Connector degree (ACD)
	There is a strong correlation between the density and understanding of the model.	[35]	Knowledge	Yes	Pragmatic Q.		Density

	An increase in coefficient of connectivity implies an increase in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Coefficient of connectivity (CNC)
	models with values of coefficient of connectivity below 0.6 are easy to understand	[10]	Threshold	Yes	Pragmatic Q.	0.6	Coefficient of connectivity (CNC)
	A process models are easier to understood and contain less errors if they have a high cross-connectivity	[36]	Knowledge - Metric	Yes	Syntactic, Semantic, Empirical and Pragmatic Q.		Cross connectivity (CC)
	models with values of total number of sequence flows below 34 are easy to understand	[37]	Threshold	Yes	Pragmatic Q.	34	Total number of sequence flows (TNSF)
	TNSF correlates with understandability of the models.	[34]	Knowledge	Yes	Pragmatic Q.		Total number of sequence flows
	models with values of connectivity level between activities below 1.1 are easy to understand	[10]	Threshold	Yes	Pragmatic Q.	1.1	Connectivity level between activities (CLA) CLA=Total number of activities/Number of sequences between activities
	Negative effect of a relatively high number of arcs on a model's understandability.	[29]	Knowledge	Yes	Pragmatic Q.		Average connector degree (ACD) Density

Number of gateways							
	A model with total number of gateways below to 5 is considered efficient in understandability tasks	[10, 37]	Threshold	Yes	Pragmatic Q.	5	Total number of gateways (TNG)
S5	A model with total number of gateways smaller or equal than 12 is considered easy to understand/modify	[38]	Threshold	Yes	Pragmatic Q. Empirical Q.	12	Total number of gateways (TNG)
	Total number of gateways correlates with understandability of the models.	[34]	Knowledge	Yes	Pragmatic Q.		Total number of gateways (TNG)
	A model with a gateway complexity indicator smaller or equal than 8.77 is easy to understand and modify.	[38]	Threshold	Yes	Pragmatic Q. Empirical Q.	8.77	Gateway Complexity Indicator (GCI)
	A model with control flow complexity below or equal to 22	[38, 39]	Threshold	Yes	Semantic Q. Pragmatic Q.	22	Control flow complexity (CFC)
	An increase in number of connectors implies an increase in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Control flow complexity (CFC) Number of connectors (Sc)
Number of activities							
	PMS4: With an increasing number of activities process models become more difficult to understand and maintain Solution: RF4 (Extract Process	[3]	Smell	Yes	Pragmatic Q. Empirical Q.		

	Fragment)						
	Total number of activities correlates with modifiability of the models.	[34]	Knowledge	Yes	Empirical Q.		Total number of activities (TNA)
	Confirmed relationship between number of functions metric and error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Number of functions (Sf)
Elements complexity - Gateways complexity							
C1	Minimize the routing paths per element **in this study authors used "element" term but they refer to gateways*	[6]	Guideline	Yes	Error probability Pragmatic Q.		Average connector degree (ACD) Maximum connector degree (MCD)
C1	Use no more than 3 inputs or outputs (inputs + outputs) per connector	[5]	Guideline	Yes	Error probability	3	Average connector degree (ACD) Maximum connector degree (MCD)
	A model with number of sequence flows from gateways below to 11 is easy to understand	[10]	Threshold	Yes	Pragmatic Q.	11	Number of sequence flows from gateways (NSFG)
	A model with number of sequence flows from gateways below to 9 is easy to modify	[10]	Threshold	Yes	Empirical Q.	9	Number of sequence flows from gateways (NSFG)
	Number of sequence flows from gateways correlates with	[34]	Knowledge	Yes	Pragmatic Q.		Number of sequence flows

	understandability and modifiability of the models.				Empirical Q.		from gateways (NSFG)
	An increase in average connector degree and maximum connector degree implies an increase in error probability	[14]	Knowledge - Metrics	Yes	Error probability		Average connector degree (ACD) Maximum connector degree (MCD)
	A model with average connector degree below3.83 is considered easy to understand/modify	[38]	Knowledge	Yes	Pragmatic Q.	3.83	Average connector degree (ACD)
	A model with maximum connector degree below or equal to5 is considered easy to understand/modify	[38]	Knowledge	Yes	Pragmatic Q.	5	Maximum connector degree (MCD)
	There is a strong correlation between the average connector degree and understanding of the model.	[35]	Knowledge	Yes	Pragmatic Q.		Average connector degree (ACD
	Negative effect of a relatively high number of arcs on a model's understandability.	[29]	Knowledge	Yes	Pragmatic Q.		Average connector degree (ACD
	A model with CFCand split = 0 is considered easy to understand/modify	[10]	Threshold	Yes	Pragmatic Q. Empirical Q.	0	Control flow complexity for AND splits (CFCand split)
	A model with CFCand split<= 1 is considered easy to understand	[40]	Threshold	Yes	Pragmatic Q.	1	Control flow complexity for AND splits (CFCand split)

	A model with CFCor split <= 1 is considered easy to understand/modify	[10]	Threshold	Yes	Pragmatic Q. Empirical Q.	1	Control flow complexity for inclusive OR-splits (CFCor split)
	A model with CFCor split <= 2 is considered easy to understand	[40]	Threshold	Yes	Pragmatic Q.	2	Control flow complexity for inclusive OR splits (CFCor split)
	A model with CFCxor split = 8 is considered easy to understand/modify	[10]	Threshold	Yes	Pragmatic Q. Empirical Q.	8	Control flow complexity for XOR splits (CFCxor split)
	A model with CFCxor split<=12 is considered easy to understand	[40]	Threshold	Yes	Pragmatic Q.	12	Control flow complexity for XOR splits (CFCxor split)
	Do not use multiple inputs and multiple outputs on the same gateway.	[11]	Best practice	No	*Pragmatic Q.		
	Decompose an AND (or XOR) gateway with multiple incoming and multiple outgoing flows into an AND (or XOR) join gateway followed by an AND (or XOR) split gateways, where the join gateway has all the incoming flow and the split gateway has all the outgoing flows;	[13]	Transformation rule	No			
Elements complexity - Events complexity							
	A model with number of sequence flows from events<= 4 is easy to	[10]	Threshold	Yes	Pragmatic Q.	4	Number of sequence flows from events(NSFE)

	understand						
	Number of sequence flows from events correlates with understandability of the models.	[34]	Knowledge	Yes	Pragmatic Q.		Number of sequence flows from events(NSFE)
	There exists a correlation between the times of understandability and NSFE	[33]	Knowledge	Yes	Pragmatic Q		Number of sequence flows from events(NSFE)
Morphology - Depth (nesting)							
	An increase in Depth implies a decrease in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Depth
	Models with values of Depth below 1 are easy to understand	[10]	Threshold	Yes	Empirical Q.	1	Depth
	Nesting depth value has its impact onto the structured related complexity metrics.	[41]	Knowledge	No	Empirical Q. Pragmatic Q.		Depth
	The depth of the nesting shows how many routing constructs in the BPM have to be understood in order to reason about the execution of an activity.	[42]	*mentioned in the paper*	No	Pragmatic Q.		Depth
Morphology - Cyclicity							
	Higher values of separability had a significant correlation with understandability I:* separability is the ratio of the	[43]	Knowledge	Yes	Pragmatic Q.		Separability

	number of cut-vertices divided by the total number of nodes in the process model. The concept is related to cyclicity						
	An increase in separability implies a decrease in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Separability
	An increase in cyclicity should imply an increase in error probability of the overall model.	[30]	Metric	Yes (relies on validated sources)	Error Probability		Cyclicity (CYC)
	An increase in separability should imply a decrease in error probability of the model.	[30]	Metric	Yes (relies on validated sources)	Error Probability		Separability
	models with values of cyclicity below 0.005 are less error prone	[5]	Knowledge - Threshold	Yes	Error probability	0.005	Cyclicity (CYC)
	models with values of separability above 0.49 are less error prone	[5]	Threshold	Yes	Error probability	0.49	Separability
	models with values of Sequentiality above 0.6 are easy to understand and modify	[10]	Threshold	Yes	Pragmatic Q. Empirical Q.	0.6	Sequentiality
	models with values of Sequentiality above 0.21 are less error prone	[5]	Threshold	Yes	Error probability	0.21	Sequentiality
	An increase in Sequentiality should imply a decrease in error	[30]	Metric	Yes (relies on validated	Error Probability		Sequentiality

	probability of the overall model.			sources)			
	An increase in Sequentiality implies a decrease in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Sequentiality
	In a loop, the backward connection should begin in an XOR split and lead back to an XOR join	[21]	Anti-patterns	No	Correctness (usability, clarity, comparability and maintainability)		
C8	Models with cycles (with multiple exit points)were found to be inherently unstructured	[18]	Knowledge	Yes	Pragmatic Q. Empirical Q.		*Cyclicity (CYC)
Morphology - Parallelism							
C7	Minimize the level of concurrency (parallelism)	[5]	Guideline	Yes	Error probability	7.5	Token Split (TS)
	An increase in Token Split implies an increase in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Token Split (TS)
	An increase in Token Split should imply an increase in error probability of the overall model	[30]	Knowledge	Yes (relies on validated sources)	Syntactic Q.		Token Split (TS)
	models with values of Sequentiality above 0.6 are easy to understand and modify	[10]	Threshold	Yes	Pragmatic Q. Empirical Q.	0.6	Sequentiality

	models with values of Sequentiality above 0.21 are less error prone	[5]	Threshold	Yes	Error probability	0.21	Sequentiality
	An increase in Sequentiality should imply a decrease in error probability of the overall model.	[30]	Metric	Yes (relies on validated sources)	Error Probability		Sequentiality
	An increase in Sequentiality implies a decrease in error probability	[14]	Knowledge - Metrics	Yes	Error Probability		Sequentiality
	Each parallel path must reach an end event or must be synchronized	[11]	Best practice	No	*Semantic Q.		
Morphology – Structuredness							
	A model with connector mismatch <= 6 is considered easy to understand/modify	[38]	Threshold	Yes	Semantic Q. Pragmatic Q.	6	Connector mismatch (MM)
	An increase in MM implies an increase in error probability *(Minor importance predictor of error.)*	[14]	Knowledge - Metrics	Yes	Error Probability		Connector mismatch (MM)
	Models with values of connector mismatch below 6 are easy to understand	[10]	Threshold	Yes	Pragmatic Q.	6	Connector mismatch (MM)
	Models with values of connector mismatch below 4 are easy to modify	[10]	Threshold	Yes	Empirical Q.	4	Connector mismatch (MM)

	In a well-structured model, splits and joins are properly nested such that each split has a corresponding join for the same type.	[44]	Metric	Yes	Correctness Errors		Unmatched Connector Count(UCC) Degree of structuredness (DoS)
	A decrease in degree of structuredness implies an increase in error probability.	[45]	Metric	Yes	Error Probability		Degree of structuredness (DoS)
	An increase in unmatched connector count implies an increase in error probability	[45]	Metric	Yes	Error Probability		Unmatched connector count (UCC)
M1	Model as structured as possible (every split connector matches a respective join connector of the same type)	[6]	Guideline	Yes	(Error probability) Pragmatic Q.		Structuredness
	Use design patterns to avoid mismatch	[5]	Guideline - Threshold	Yes	Error probability	4.5	Connector mismatch (MM)
	Model as structured as possible (structuredness greater than 0.79 can be interpreted as indicators of poor design quality as regards error probability)	[5]	Guideline - Threshold	Yes	Error probability	0.79	Structuredenss
	Structured models are less error prone	[14]	Knowledge - metrics	Yes	Error probability		Structuredness
	Structuring leads to more understandable models if it does not increase the number of	[18]	Knowledge	Yes	Pragmatic Q.		* Connector mismatch (MM) *Structuredness

	gateways (*This is not a general rule as we have found one instance where a clear increase in understandability is achieved despite an increase in the number of gateways.*)						
	An increase in structuredness should imply a decrease in error probability of the overall model.	[30]	Metric	Yes (relies on validated sources)	Error probability		Structuredness
	For each decision node there must be a matching merge node	[46]	Well-formedness and style guidelines	No	Structuredness - Empirical Q.		
	Pattern “Straight Road with Exits" (Good excuse for unstructured modeling)	[47]	Pattern	No	Empirical Q.		
Morphology - Diameter							
S2	A larger business process model in terms of diameter should be more likely to contain errors than a small one, since the modeler would only be able to perceive a certain amount of consecutive nodes in a certain period.	[20]	Metric	Yes	Error Probability		Diameter (diam)
	An increase in diameter implies an increase in error probability. This is	[14]	Knowledge - Metrics	Yes	Error Probability		Diameter (diam)

	a minor importance predictor of error.						
Morphology - iOR Gateways							
C2	A model with GH<=0.79 is considered easy to understand/modify	[38]	Knowledge	Yes	Empirical Q. Pragmatic Q.	0.79	Gateway Heterogeneity (GH)
	An increase in connector heterogeneity implies an increase in error probability Restrict the modeling to the two connector types AND and XOR, and use iOR gateways only in structured blocks.	[14]	Knowledge - Metrics	Yes	Error Probability		Connector Heterogeneity (CH)
	Avoid iOR joins and iOR splits	[5]	Guideline	Yes	Error probability	0.5 0.56	Number of inclusive OR splits Number of inclusive OR joins
	Minimize the heterogeneity of connector types.	[5]	Guideline	Yes	Error probability	0.4	Connector Heterogeneity (CH)
C6	Avoid iOR routing elements	[6]	Guideline	Yes	Error probability Semantic Q.		
	Avoid use of inclusive OR gateways Avoid the use of inclusive decision. Model all possible combinations of branches using decision and forks.	[21]	Anti-patterns	No	Correctness (usability, clarity, comparability and maintainability)		

General complexity							
C3	Higher complexity impacts on error probability	[15] [14, 20]	Knowledge	Yes	Error probability		Structuredness Coefficient of connectivity (CNC) Cyclicity (CYC) Separability Connector heterogeneity (CH) Connector mismatch (MM) Diameter (diam)
	Simpler models are preferable over more complex models	[48]	Knowledge	Yes	Errors probability Pragmatic Q. Empirical Q.		
	Structural metrics apparently seem to be closely connected with understandability and modifiability The correlation analysis results indicate that there is a significant relationship between structural metrics and the time and efficiency of understandability	[37]	Knowledge	Yes	Pragmatic Q. Empirical Q.		*For Understandability:* Number of nodes (SN) Gateway mismatch (MM) Depth Coefficient of connectivity (CNC) Sequentiality *For Modifiability:*

							Gateway mismatch (MM) Density Sequentiality
	The more complex the model is the worse will be the understandability of it.	[35]	Knowledge	Yes	Pragmatic Q.		Density Average connector degree (ACD) Cross-Connectivity (CC)
	PMS2: Using the more complex alternative may negatively affect model understanding, and thus make maintenance of the model more difficult. RF3: Substitute Process Fragment	[3]	Knowledge	Yes	Pragmatic Q. Empirical Q.		
	Confirmed relationship between metrics and error probability	[14]	Knowledge - Metrics	Yes	Syntactic Q.		Number of nodes (S_N) Number of events (S_E) Number of functions (S_F) Number of connectors (S_C) Number of arcs (S_A) Diameter (diam) Coefficient of connectivity (CNC) Average connector degree (ACD) Maximum connector degree

							(MCD) Separability Sequentiality Structuredness Depth Connector mismatch (MM) Connector heterogeneity (CH) Control flow complexity (CFC) Cyclicity (CYC) Token splits (TS)
	Complexity has undesirable effects on, among others, the correctness, maintainability, and understandability of business process models.	[49]	Metric	No	Empirical Q. Pragmatic Q. correctness		Metrics: Number of activities (NOA) Number of activities and control-flow elements in a process (NOAC) Number of activities, joins, and splits in a process (NOAJS) Control Flow Complexity (CFC) Halstead-based Process Complexity (HPC)

							Complexity of a procedure (PC)
	Decompose a model with more than 31 nodes	[5]	Guideline	Yes	Error probability	31	*Number of Nodes (S_N)
	Decompose the model if it has more than 50 elements	[6]	Guideline	Yes - (Built on empirical insights)	(error probability)		*Number of Nodes (S_N)
	The results of the experiments we presented point to the usefulness of modularization to improve the understandability of a process model.	[8]	Knowledge	Yes	Pragmatic Q.		
	Subprocesses foster the understanding of a complex business process model by their ‘‘information hiding’’ quality	[24]	Knowledge	Yes	Pragmatic Q.		
	We infer that modularity may be helpful for understanding a process model because it shields the reader from unnecessary information.	[7]	Knowledge	Yes	Pragmatic Q.		
	Make your models hierarchical Hide details of an end-to-end process using subprocess	[11]	Best practice	No	Empirical Q. Pragmatic Q.		

	Fit each process level on one page. Add major steps as subprocesses						
	Some of the tasks were depicted as subprocesses although they just represent the single activities of the resources.	[50]	Problem	No.	*Semantic Q.		
	If models become too complex to be understood by end users, they should be split into simpler models	[48]	Knowledge	Yes- and other empirical evidence in [15, 51]	Error probability Empirical Q. Pragmatic Q.		
	PMS5: Inclusion of many small process models is bad for maintenance. RF6: Inline Process Fragment (Decompositions which are too extreme (i.e., which result in many tiny process models) are not optimal in terms of maintenance and usability.)	[3]	Knowledge	Yes	Empirical Q.		
M6	Inclusion of many small **process** models	[8, 24]	Problem	Yes	*Empirical Q.		
	While there is no source that specifies an optimal, lower bound for the number of activities in	it was said in: [3], and	Guideline	No	*Empirical Q.		

	subprocesses, guidelines suggest that this number should range from 5 to 7 [52].	ref. to: [52]					
M4	Good candidates for subprocesses are fragments of a model that are components with a single input and a single output control flow arc (Block-structuredness)	[8, 24, 53]	Modularization criterion	Yes	Pragmatic Q.		
M5	Good candidates for subprocesses are those fragments of a model which nodes are more strongly connected by arcs to each other than the nodes outside this collection (Connectness) - this is the most attractive criterion according to [8, 24]	[8, 24]	Modularization criterion	Yes	Pragmatic Q.		
Layout							
	The use of colors to highlight matching operator transitions have a significant, positive impact on understanding accuracy for novices.	[25]	Knowledge	Yes	Pragmatic Q.		
	Make your models long and thin (instead of square)	[54]		No			
	Include iconic representations for the different identified verb classes	[55]	Knowledge	No	Empirical Q. Pragmatic Q.		

	The number of edge bends negatively affects the understanding of a process model	[56]	Layout factor	No	Empirical Q. Pragmatic Q.		
L1	Line Crossings: The higher the number of crossings within a graphical layout, the lower the readability and understanding of that layout	[56]	Layout factor	No	Empirical Q. Pragmatic Q.		
	spaghetti-like modeling (layout)	[50]	Problem	No	Empirical Q.		
	Minimize the number of crossings of connecting elements.	[57]	layout aesthetic	No	Empirical Q.		
	Symmetry: Graphical layouts where elements are placed more symmetric are easier readable. (the effects of symmetry are lower than line crossings)	[56]	Layout factor	No	Empirical Q. Pragmatic Q.		
	Use of Locality: Graphical elements which are related to each other shall be placed close to each other making them easier recognizable and leading to a higher understanding of the model	[56]	Layout factors	No	Empirical Q. Pragmatic Q.		
L2	Minimize the area of the drawing.	[57]	layout aesthetics	No	Empirical Q.		
L3	Minimize the number of bends of connecting elements.	[57]	layout aesthetics	No	Empirical Q.		

L4	Minimize the number of overlapping (connection) elements.	[57]	layout aesthetics	No	Empirical Q.		
L5	Maximize the number of orthogonally drawn connecting objects.	[57]	layout aesthetics	No	Empirical Q.		
L6	Maximize the number of connecting objects respecting workflow direction.	[57]	layout aesthetics	No	Empirical Q.		
L7	Adapt the size of objects such that elements have enough space.	[57]	layout aesthetics	No	Empirical Q.		
L8	Consider the use of partitions, e.g. pools and swimlanes.	[57]	layout aesthetics	No	Empirical Q.		
	Specify task types, especially user (human task) and service (automated task).	[58]	Best practice	No	Empirical Q. Pragmatic Q.		
	Keep the diagram as neat and consistently organized as possible - general (contains some other layout guidelines)	[58]	Best practice	No			
	Detected Layout errors: Absence of uniform style for flow layout	[59]	Style violation	No	Empirical Q.[2]		

[2]Author calls this “layout quality”; we call it Empirical Quality.

Label style							
LS1	Use verb-object activity labels	[6]	Guideline	Yes- (Built on empirical insights)	Pragmatic Q.		
	Verb-object style is less ambiguous	[26]	Knowledge	Yes	Pragmatic Q.		
	Verb-object style is less ambiguous and more useful than other labeling styles	[27]	Knowledge	Yes	Empirical Q. Pragmatic Q. Perceived Semantic Q.		
LS2	Shorter activity labels improve model understanding Textlength was significantly correlated with the different proxies for understandability.	[43]	Knowledge	Yes	Pragmatic Q.		
	PMS1: Improper labels may not reveal the intended content or purpose to readers.	[3]	Smell	Yes	Pragmatic Q., Empirical Q. and (errors)		
	Labeling of pools, swimlanes and (connection) elements must be feasible	[57]	Layout aesthetics	No	Empirical Q.		
LS3	Absence of uniform style for names	[59]	style violation	No	Semantic Q. Empirical Q.		

	Tasks and subprocesses are not compliant with using verb-noun labeling style. *	[50]	Problem	No.	Empirical Q.		
	Label process activities VERB-NOUN	[11]	Best practice	No	Empirical Q. Pragmatic Q.		
	Label a timer event with the duration or date/time parameter	[11]	Best practice	No			
	Label gateways and all the outgoing sequence flows	[11]	Best practice	No			
	Make the process logic visible in the diagram Label everything in your diagram (activities, subprocesses, intermediate events, gateways, sequence flows, end events and message flows) using naming conventions, except AND splits and joins. Show exception handling logic explicitly in the diagram process logic: sequence of activity flow, which paths are concurrent or conditional, the various end states, etc.	[11]	Best practice	No	Empirical Q. Pragmatic Q.		
	The white box pool label should name the process	[11]	Best practice	No	Empirical Q. Pragmatic Q.		

	The black box pool label should name the participant	[11]	Best practice	No	Empirical Q. Pragmatic Q.		
	Reserve the keywords Send and Receive in task names for Send and Receive task types	[11]	Best practice	No	Empirical Q. Pragmatic Q.		

References

1. Nelson, H.J., et al., *A conceptual modeling quality framework.* Software Quality Journal, 2012. **20**: p. 201-228.
2. 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.
3. Weber, B., et al., *Refactoring large process model repositories.* Computers in Industry, 2011. **62**(5): p. 467-486.
4. Dumas, M., et al., *Fundamentals of Business Process Management*. 2012: Springer Heidelberg New York Dordrecht London.
5. Mendling, J., et al., *Thresholds for error probability measures of business process models.* Journal of Systems and Software, 2012. **85**(5): p. 1188-1197.
6. 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.
7. Reijers, H. and J. Mendling, *Modularity in Process Models: Review and Effects*, in *Business Process Management*, M. Dumas, M. Reichert, and M.-C. Shan, Editors. 2008, Springer Berlin / Heidelberg. p. 20-35.
8. Reijers, H.A., J. Mendling, and R.M. Dijkman, *On the Usefulness of Subprocesses in Business Process Models*. 2010, BPM center report: Eindhoven. p. 32.
9. Gruhn, V. and R. Laue, *Reducing the Cognitive Complexity of Business Process Models*. Proceedings of the 8th Ieee International Conference on Cognitive Informatics, ed. G. Baciu, et al. 2009. 339-345.
10. Sánchez-González, L., et al., *Improving Quality of Business Process Models*, in *Evaluation of Novel Approaches to Software Engineering, ENASE 2011*, L.A. Maciaszek and K. Zhang, Editors. 2013, Springer: Beijing, China. p. 130-144.
11. Silver, B., *BPMN: Method and Style*. 2009: Cody-Cassidy Press.
12. Claes, J., et al., *Tying process model quality to the modeling process: the impact of structuring, movement, and speed*, in *Business Process Management Conference*. 2012. p. 33-48.
13. 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.
14. Mendling, J., G. Neumann, and W.M.P. van der Aalst, *Understanding the occurrence of errors in process models based on metrics*, in *On the Move to Meaningful Internet Systems*, R.T.Z. Meersman, Editor. 2007. p. 113-130.
15. Mendling, J., et al., *Detection and prediction of errors in EPCs of the SAP reference model.* Data & Knowledge Engineering, 2008. **64**(1): p. 312-329.
16. Fenton, N., *Software Measurement: A Necessary Scientific Basis.* Ieee Transactions on Software Engineering, 1994. **20**(3).
17. Mendling, J., *Metrics for process models: empirical foundations of verification, error prediction, and guidelines for correctness*. 2008: Springer-Verlag Berlin Heidelberg.
18. Dumas, M., et al., *Understanding Business Process Models: The Costs and Benefits of Structuredness*, in *Advanced Information Systems Engineering*, J. Ralyté, et al., Editors. 2012, Springer Berlin Heidelberg. p. 31-46.

19. Polyvyanyy, A., L. García-Bañuelos, and M. Dumas, *Structuring acyclic process models.* Information Systems, 2012. **37**(6): p. 518-538.

20. Mendling, J., *Detection and Prediction of Errors in EPC Business Process Models*, in *Institute of Information Systems and New Media*. 2007, Vienna University of Economics and Business Administration (WU Wien): Austria.

21. Koehler, J. and J. Vanhatalo, *Process anti-patterns: How to avoid the common traps of business process modeling*, in *Tech. Rep. Report RZ-3678*. 2007, IBM Zurich Reseach Lab.

22. Kindler, E., *On the semantics of EPCs: Resolving the vicious circle.* Data & Knowledge Engineering, 2006. **56**: p. 23-40.

23. Moreno-Montes de Oca, I. and M. Snoeck, *A look into business process modeling guidelines through the lens of the technology acceptance model*, in *Submited to QMMQ-14 Workshop*. 2014, UCLV-KULeuven: Atlanta, GA USA.

24. Reijers, H.A., J. Mendling, and R.M. Dijkman, *Human and automatic modularizations of process models to enhance their comprehension.* Information Systems, 2011. **36**(5): p. 881-897.

25. Reijers, H.A., et al., *Syntax highlighting in business process models.* Decision Support Systems, 2011. **51**(3): p. 339-349.

26. Mendling, J. and H.A. Reijers. *The impact of activity labeling styles on process model quality*. in *European Symposium on Analysis, Design, Use and Societal Impact of Information Systems*. 2008. Marburg, Germany.

27. Mendling, J., H.A. Reijers, and J. Recker, *Activity labeling in process modeling: Empirical insights and recommendations.* Information Systems, 2010. **35**(4): p. 467-482.

28. Mendling, J., M. Strembeck, and J. Recker, *Factors of process model comprehension-Findings from a series of experiments.* Decision Support Systems, 2012. **53**(1): p. 195-206.

29. Mendling, J., H.A. Reijers, and J. Cardoso, *What makes process models understandable?*, in *Business Process Management*, G. Alonso, P. Dadam, and M. Rosemann, Editors. 2007, Springer-Verlag Berlin Heidelberg: Brisbane, Australia. p. 48-63.

30. Mendling, J. and G. Neumann. *Error metrics for business process models*. in *International Conference on Advanced Information Systems Engineering*. 2007.

31. Becker, J., M. Kugeler, and M. Rosemann, *Process Management: A Guide for the Design of Business Processes*. 2003: Springer.

32. Gruhn, V. and R. Laue, *Detecting Common Errors in Event-Driven Process Chains by Label Analysis.* Enterprise Modelling and Information Systems Architectures, 2011. **6**(1): p. 3-15.

33. Rolón, E., et al. *An Exploratory Experiment to Validate Measures for Business Process Models*. in *International Conference on Research Challenges in Information Science*. 2007. Ouarzazate, Morocco.

34. Rolon, E., et al. *Prediction models for BPMN usability and maintainability*. in *2009 IEEE Conference on Commerce and Enterprise Computing*. 2009.

35. Reijers, H.A. and J. Mendling, *A Study Into the Factors That Influence the Understandability of Business Process Models.* Ieee Transactions on Systems, Man, and Cybernetics - Part A, 2011. **41**(3): p. 449-462.

36. Vanderfeesten, I., et al., *On a quest for good process models: The cross-connectivity metric*, in *Advanced Information Systems Engineering*, Z. Bellahsene and M. Leonard, Editors. 2008. p. 480-494.

37. Sánchez-González, L., et al., *Quality Assessment of Business Process Models Based on Thresholds*, in *On the Move to Meaningful Internet Systems*, R. Meersman, T. Dillon, and P. Herrero, Editors. 2010. p. 78-95.

38. Sánchez-González, L., et al., *Quality indicators for business process models from a gateway complexity perspective.* Information and Software Technology, 2012. **54**(11): p. 1159-1174.

39. Rolón, E., et al., *Analysis and Validation of Control-Flow Complexity Measures with BPMN Process Models*, in *Enterprise, Business-Process and Information Systems Modeling*, T. Halpin, et al., Editors. 2009, Springer-Verlag Berlin Heidelberg. p. 58-70.

40. Sánchez-González, L., et al. *Towards Thresholds of Control Flow Complexity Measures for BPMN models*. in *2011 ACM Symposium on Applied Computing*. 2011. ACM.

41. Gruhn, V. and R. Laue, *2 Approaches for business process model complexity metrics*, in *Technologies for Business Information Systems*, W. Abramowicz and H.C. Mayr, Editors. 2007, Springer Netherlands. p. 13 - 24.

42. Figl, K. and R. Laue, *Cognitive Complexity in Business Process Modeling*, in *Advanced Information Systems Engineering*. 2011. p. 452-466.

43. Mendling, J. and M. Strembeck, *Influence factors of understanding business process models*, in *Business Information Systems*, W.F.D. Abramowicz, Editor. 2008, Springer: Innsbruck, Austria. p. 142-153.

44. Laue, R. and J. Mendling, *The impact of structuredness on error probability of process models*, in *Information Systems and E-Business Technologies*, R. Kaschek, et al., Editors. 2008, Springer-Verlag Berlin Heidelberg. p. 585-590.

45. Laue, R. and J. Mendling, *Structuredness and its significance for correctness of process models.* Information Systems and E-Business Management, 2010. **8**(3): p. 287-307.

46. Reggio, G., et al., *On business process modelling with the UML: a discipline and three styles*. 2011, Dipartimento di Informatica e Scienze dell'Informazione (DISI) - University of Genova Italy.

47. Gruhn, V. and R. Laue. *Good and Bad Excuses for Unstructured Business Process Models*. in *12th European conference on pattern languages of programs (EuroPLoP 2007)*. 2007. UVK - Universitaetsverlag Konstanz.

48. Lassen, K.B. and W.M.P. van der Aalst, *Complexity metrics for Workflow nets.* Information and Software Technology, 2009. **51**(3): p. 610-626.

49. Cardoso, J., et al. *A discourse on complexity of process models*. in *Business Process Management Workshops*. 2006. Springer-Verlag Berlin Heidelberg.

50. Arkilic, I.G., H.A. Reijers, and R.R.H.M.J. Goverde, *How good is an as-is model really?*, in *Business Process Management Workshops*, M. Rosa and P. Soffer, Editors. 2013, Springer Berlin Heidelberg. p. 89-100.

51. Mendling, J., et al., *Faulty EPCs in the SAP Reference Model.*, in *Business Process Management*, S. Dustdar, J.L. Fiadeiro, and A.P. Sheth, Editors. 2006, Springer: Vienna, Austria. p. 451-457.

52. Sharp, A. and P. McDermott, *Workflow Modeling: Tools for Process Improvement and Application Development*. 2001: Artech House.

53. Vanhatalo, J., H. Volzer, and F. Leymann. *Faster and more focused control-flow analysis for business process models through sese decomposition*. in *5th International Conference on Service-Oriented Computing*. 2007. Vienna, Austria: Lecture Notes in Computer Science.

54. Davis, R., *Business Process Modelling with Aris: A Practical Guide*. 2001: Springer.

55. Mendling, J., J. Recker, and H.A. Reijers, *On the usage of labels and icons in business process modeling.* International Journal of Information System Modeling and Design, 2010. **1**(2): p. 40-58.

56. Schrepfer, M., et al. *The impact of secondary notation on process model understanding*. in *The Practice of Enterprise Modeling, Working Conference*. 2009. Stockholm, Sweden.

57. Effinger, P., N. Jogsch, and S. Seiz, *On a Study of Layout Aesthetics for Business Process Models Using BPMN*, in *Business Process Modeling Notation*, J. Mendling, M. Weidlich, and M. Weske, Editors. 2010, Springer-Verlag Berlin Heidelberg. p. 31-45.

58. Silver, B. *Ten tips for effective process modeling*. Bpms watch, 2008.

59. Reggio, G., M. Leotta, and F. Ricca. *Precise is better than light - a document analysis study about quality of business process models*. in *2011 First International Workshop on Empirical Requirements Engineering*. 2011.

FACULTY OF ECONOMICS AND BUSINESS
Naamsestraat 69 bus 3500
3000 LEUVEN, BELGIË
tel. + 32 16 32 66 12
fax + 32 16 32 67 91
info@econ.kuleuven.be
www.econ.kuleuven.be