Best Practice Guidelines for BPMN 2.0

Gerardo Navarro Suarez, Jakob Freund and Matthias Schrepfer
camunda services GmbH, Germany

INTRODUCTION

In practice modeling projects often tend to be quite large. Adopting BPMN 2.0 eases the creation of process models for business and technical projects. However, the creation of models in large modeling projects is still not a trivial task. The introduction of modeling guidelines guides and supports modeling projects. This article introduces an approach to establish such modeling guidelines for individual modeling projects using BPMN 2.0 as modeling notation. The article discusses the concept of modeling guidelines and shows why their application can help to apply BPMN 2.0 in practice. A framework for the creation of guidelines is described in detail. Real-world examples illustrate the use of modeling guidelines and constitute the effectiveness of best practice guidelines.

WHAT ARE MODELING GUIDELINES?

Modeling guidelines are documents which collect and provide best practices for modeling projects. In the context of modeling guidelines, best practices are techniques, methods or examples to support process modeling. If applied in modeling projects, best practices illustrate effective ways to capture real-world scenarios in process models. These best practices derive from experience as well as from academic research. Modeling guidelines cover best practices in form of instructions or rules which should be followed while modeling business processes in BPMN 2.0. Modeling guidelines guide and influence process modelers to use best practices for process modeling. Additionally, best practice guidelines can be used to enforce specific instructions in projects.

WHY DO WE NEED BEST PRACTICE GUIDELINES?

Modeling projects usually require the creation of several process models. These models often tend to be quite complex and large. This complexity must be dealt with by process modelers who map real-world process into process models. In real projects, modelers often lack knowledge in process modeling. Thus, they need guidance for the modeling process in order to create high quality processes. Best practice guidelines compensate these issues by guiding modelers with examples. Best practices assist modelers especially while mapping real-world processes and knowledge into process models. Furthermore, best practices help to manage and govern modeling projects especially if many models need to be created. Best practices further contribute to achieve higher quality of process models. Process modelers are advised to use modeling and visualization styles that are superior to others. This leads to more sophisticated model representations which in turn positively affect the readability and understandability of the created process models. The number of errors in process models decreases due to the increased comprehensibility. Model viewers understand the models faster and better. In addi-

tion, the purpose of process models, most often the communication between project stakeholders, is achieved at a higher degree.

CONTENT OF BEST PRACTICE GUIDELINES

Modeling projects differ in their purposes and goals. Due to this the contents of best practice guidelines also varies. The establishment of modeling guidelines requires the availability of the scope definition of the modeling project. The guideline content should be based on this definition so that users can apply the guideline to support the project-specific goals. In practice the guideline content is divided into categories. These categories assist organizations to develop individual guidelines. Best practice examples are put into categories based on their scope. In the following we introduce and define the categories that best practice guidelines should contain. Examples illustrate the application of guidelines.

BPMN 2.0 Symbol Set

The specification of BPMN 2.0 defines the symbols to be used for creating process models. The specification describes the syntax and semantics of all symbols. In practice, process modelers are free to choose symbols to map real-world situations into process models. Therefore, we recommend you to define the symbol set which you want to use within projects. Based on the scope of the modeling project the set of symbols should be chosen. An agreed set of symbols helps to avoid misinterpretations of symbols and supports the harmonization of process models. The symbol set in modeling guidelines is divided into two sub-categories: symbols and artifacts. Both categories are defined as follows.

BPMN Symbols

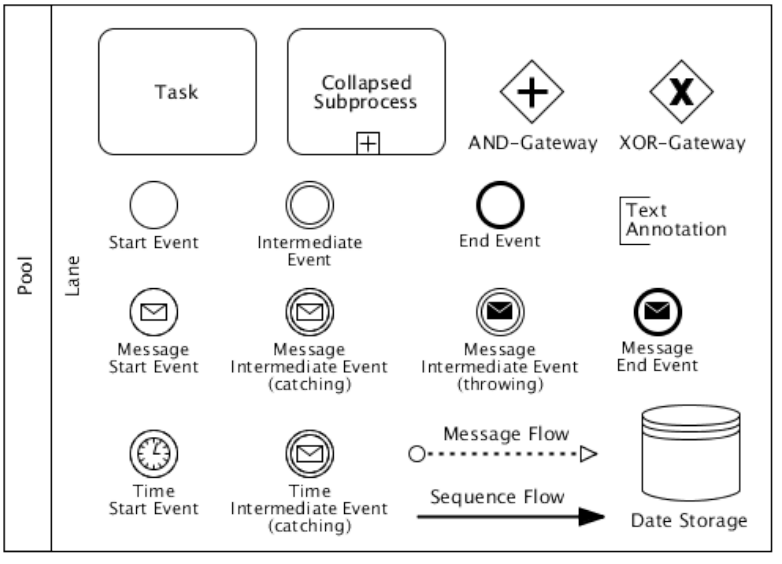
As mentioned before the BPMN 2.0 specification defines the symbols. Due to the large amount of symbols in BPMN 2.0 some elements are seldom used in actual modeling projects. The large set of BPMN symbols might be too large. Then a subset of symbols might be convenient. The restriction decreases the number of symbols which helps process modelers to focus on the actual modeling goals. However, for creating such subsets the project purposes and goals must be known. Thus, the symbol subset can differ among projects. The category contains all specified BPMN 2.0 elements except the so-called artifacts. To illustrate the application of symbol sets we describe how to define such subsets.

BPMN Symbols – Examples

A subset defines the BPMN 2.0 symbols which can be used in modeling projects. The elements of a subset must be suitable to capture all situations of the project in process models. The subsets must be designed carefully so that they fit the projects. The BPMN 2.0 specification already provides three predefined subsets, so called conformance sub-classes: descriptive, analytic, common executable. The descriptive and the analytical sub-classes focus on process elements and a small amount of attributes. The common executable sub-class concentrates on executable process models. It contains all BPMN 2.0 symbols and a large number of attributes of process elements to prepare models for their automatic execution. In case these classes do not meet the purpose of modeling projects, individual subsets can be created. Establishing such subsets works as follows. Symbols of the specified BPMN 2.0 symbol set that are suitable for the project are taken and grouped as subset.

In order to illustrate the application of symbol subsets, the following example is used: The goal of a modeling project is to create process diagrams with the objective to document processes suitable for the management level of an organization.

To actually model such high-level processes, the full symbol set of BPMN 2.0 is not needed. Thus, we define a small subset that contains the necessary symbols. The subset reduces the amount of symbols and helps the process modeler not to get lost in details. The modeling guideline for the project describes the symbols of the subset. In Figure 1 the subset is depicted. The subset is suitable to create models that are intuitively understandable to the management level without showing too many details of the processes.



Caption 1: A Symbol Subset for Documenting High-level Processes

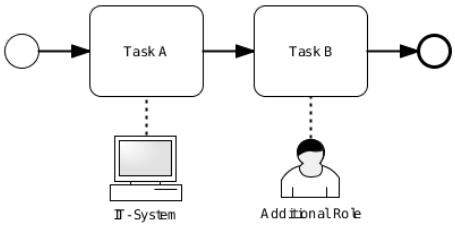
BPMN Artifacts

The BPMN 2.0 specification defines so-called artifacts. Artifacts can be used to provide additional information within process diagrams. They are connected to flow elements, e.g. tasks, with the help of associations. The specification allows the user to create customized artifacts which can be added to the standard symbol set. The specification defines the syntax of using artifacts so that the semantics of process diagrams is kept correctly. Modelers are free to define customized artifacts and use them within process models.

For individual modeling projects, customized artifacts are useful to highlight certain meanings within process models. The definition of customized artifacts helps to emphasize specific information in models. In such cases it improves the comprehensibility of process models. Customized artifacts help to present situations visually in a clearer and more precise way. The definition of artifacts must be based on the project purposes and chosen on individual basis. However, if artifacts are customized, their semantic meaning must be defined properly.

BPMN Artifacts – Examples

Figure 2 shows a process model where two customized artifacts are used. An artifact depicts an IT system, e.g. Salesforce or a SAP system, and is associated to task A. The second artifact expresses an organizational role and is associated with task B. Both artifacts add additional information to the model. The IT system expresses that task A uses the system to fulfill the task. The role associated to task B expresses that an additional participant takes part in task B.



Caption 2: Two Customized Artifacts

Modeling Styles

The category of modeling styles contains rules that are generally known and accepted. These rules apply to all modeling projects. This type of styles does not rely on specific modeling situations or contexts. Modeling styles are also not dependent on actual modeling goals and purposes. The styles assist process modelers by guiding them with general instructions on how to create process models more effectively. Modeling styles restrict the freedom of process modelers to draw models as the modelers must apply the style definitions. These restrictions support the governance and standardization of process models. Modeling styles influence the modeling behavior of modelers which often leads to an increase in the comprehensibility of process models.

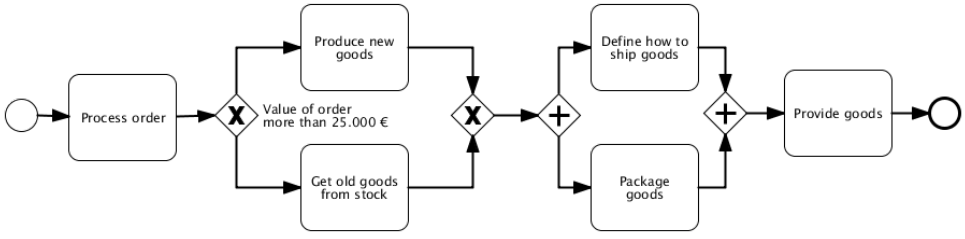
Modeling Styles – Examples

In order to illustrate the use of general modeling styles we propose three examples and explain them in detail.

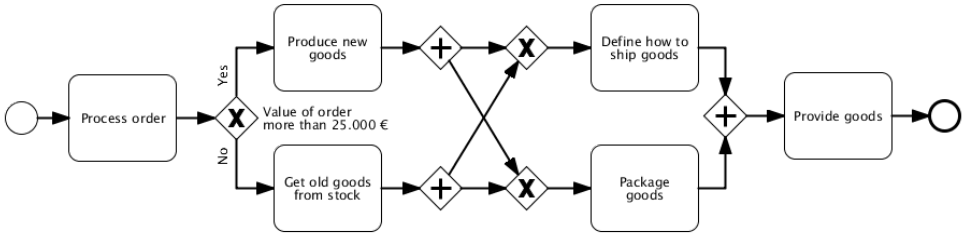
- You should model in a structured and symmetric way.
- You should decompose models with a large number of activities.
- You should use as few symbols in models as possible.

The first modeling style deals with structured and symmetric modeling. A process model is structured and symmetric if every split gateway has a respective join gateway of the same type. This modeling style leads to formations in process models that are known as blocks. Process models that incorporate block structures are easier to understand due to their increased comprehensibility. Empirical studies showed that people have difficulties to understand unstructured process models because they expect every split gateway to match its respective join gateway. Furthermore, it is more likely that unstructured process models contain more errors, e.g. deadlocks or livelocks. It must be mentioned that certain situations cannot be captured symmetrically. In such cases process models incorporate unstructured parts.

The next two figures present a scenario that is modeled in a structured and an unstructured way. Figure 3 shows the structured and symmetric process model. Notice that every opening gateway has its corresponding closing gateway. Figure 4 shows the same situation in an unstructured model. The model in Figure 4 is less comprehensible as model viewers do not recognize the model structure easily. Therefore, process modelers are advised to model as symmetric as possible.



Caption 3: Structured Process Model



Caption 4: Unstructured Process Model

The second modeling style limits the number of elements in process models. Empirical studies showed that the larger a model the more difficult it is to understand. The size of process models is thus directly connected to the understandability and error probability of a process model. Therefore, the modeling style advices to decompose process models when the number of process elements reaches a certain limit. The limit depends on a number of factors, however, we recommend to set it lower than 50. Large models should be decomposed into smaller ones with the help of sub-processes.

The third modeling style advices modelers to use a few elements as possible in process models. This style is based on similar reasons as the second one. However, as few elements as possible means that a process modeler should only use BPMN 2.0 symbols if really needed. There should be no superfluous elements in models. Process models should only contain elements that focus on the control flow in the model or those that add value to the model, e.g. in the form of additional information. Although the modeling style sounds trivial, in practice this style is not always met by modelers.

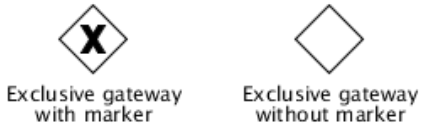
Layout Styles

The layout of process models is important when it comes to read and comprehend the model content. Models with improved layouts are easier to read than those whose layouts are obfuscated by means of visual cues, e.g. color or line width. Layout styles define rules that influence how modelers draw process models. Modeling notations do not define how to place elements within process models. Modelers are generally free to place symbols anywhere within the modeling canvas. Layout styles restrict this freedom by enforcing rules that define how process models should be visualized. Layout rules do not change the graph structure of process models so that the semantic meaning of them remains the same. Layout styles can be divided into two groups: general and optional layout styles. General layout styles should be applied to all modeling projects. Optional layout styles are recommendations to further improve visual representations. They should be fol-

lowed but they are not mandatory. In practice there are a number of rules in both categories.

Layout Styles – Examples

In the following two examples for layout styles are shown. To illustrate the use of layout styles, there will be one example for each category. One general style concerns the representation of the exclusive gateway (XOR). The specification allows the user to draw the gateway with and without an internal marker. Although it advises to use only one representation within a single process model, process modelers can change their style according to their modeling behavior. In practice this can lead to different layouts which might confuse model viewers. Therefore, it is advisable to stick to the representation of the exclusive gateway with internal marker (shown on the left side of Figure 5). Enforcing this general layout style leads to the harmonization of process models with respect to this visual cue.

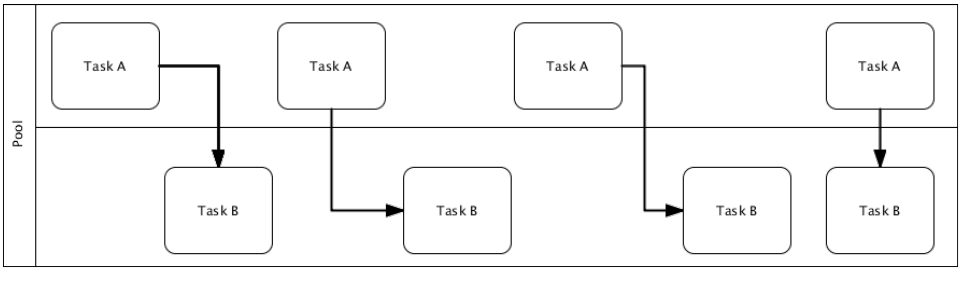


Caption 5: Representation of the Exclusive Gateway

An optional layout style is to enforce rules for the drawing of sequence flows between symbols. BPMN 2.0 does not define the way of drawing sequence flows. Process modelers are free to choose the way to draw the flows. Figure 6 shows several but not all ways for connecting two symbols with a sequence flow. All representations are semantically equivalent but they differ in placing the flows. In modeling projects where a large number of models is created, the enforcement of best practice rules for placing sequence flows in models contributes to increase their comprehensibility. For this optional layout style we recommend the following two rules.

- Incoming sequence flows should enter a symbol from the left, top or bottom side.
- Outgoing sequence flows should leave a symbol from the right, top or bottom side.

Organizations should introduce general as well as optional layout styles to control and harmonize the visual representations of process models to increase their comprehensibility.



Caption 6: Drawing of Sequence Flows between Symbols

Modeling Alternatives

Process modeling is a task where humans usually judge real-world situations and map them into models. The mapping often leads to different representation of the same situations. Although the representations differ, the information of the models is semantically identical. The differences occur due to the freedom of process modelers to use symbols from the given symbol set and to connect them according to the underlying syntax. Especially in large projects with a lot of modelers the representations for one situation usually differ. If modelers use different symbols, the visualizations of the models change which immediately affects the comprehensibility of the underlying models. If there are no rules or recommendations, modelers base their decision on intuition which is often known to be wrong, especially for novice modelers. In order to govern situations where alternative process models represent the same information, organizations should provide best practices for certain situations.

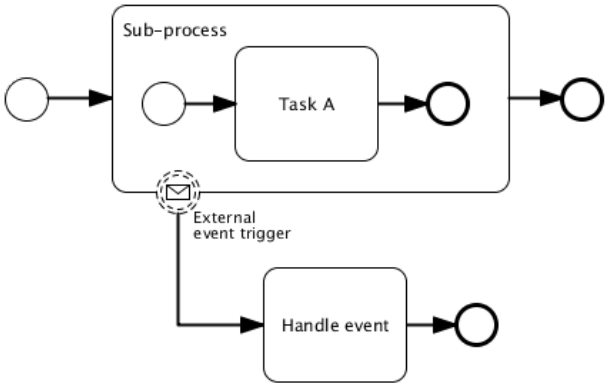
The category of modeling alternatives presents those best practices. It describes the alternatives and recommends the best one for a specific situation. The recommendations can be used in several contexts. The best practices guide process modelers to choose the best representation for specific situations.

Modeling Alternatives – Examples

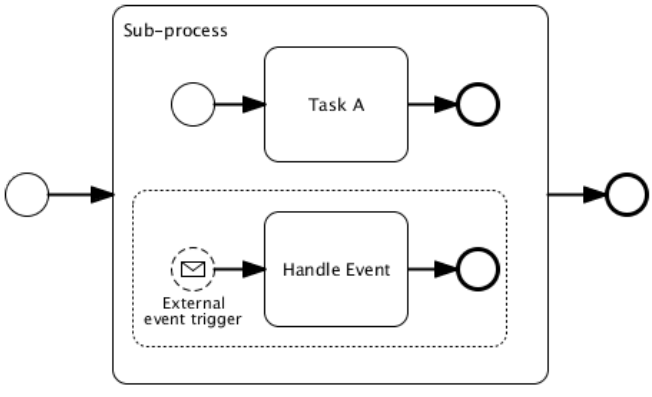
In order to illustrate the concept of modeling alternatives two examples are given. The first example deals with non-interrupting message events. The second one shows alternatives for the modeling of conditional flows.

Figure 7 depicts a sub-process that has a non-interrupting intermediate event attached to its boundary. The event is triggered by an external event. Figure 8 displays the same situation but uses different BPMN 2.0 symbols. In Figure 8 an event sub-process is modeled which starts the event handling if it is triggered by the external event. Both alternatives are identical from a control flow perspective. Therefore, modelers can choose between the alternatives. Organizations should focus on one alternative and enforce it in all process models if possible.

It must be noted that the alternatives differ in their data handling if they are executed in process engines. The event handling in Figure 7 runs in the scope of the main process. Thus, the event handling cannot access the data used within the scope of the sub-process. In Figure 8 the event sub-process runs in the scope of the sub-process. It has full access to the data used by the sub-process. With respect to reusability, the two alternatives can differ. If sub-processes are marked as call activity, the alternatives differ. Process modelers must be aware of these details and take appropriate actions. The figures shown below are identical with respect to usability as sub-processes are not marked as call activity. The modeling guideline should definitely point out these details to advice the readers.

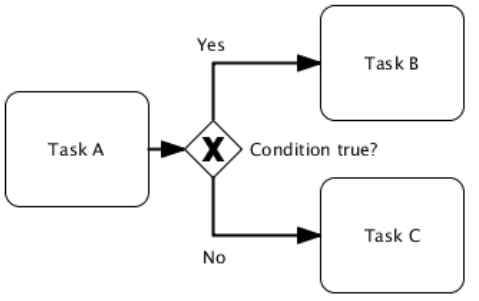


Caption 7: Sub-Process with Non-Interrupting Message Event

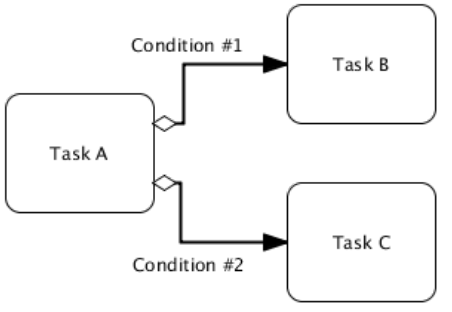


Caption 8: Sub-Process with Event Sub-Process

The second example is concerned with the modeling of conditions. Figure 9 depicts a common decision using an exclusive gateway. The gateway decides on the path that is activated based on the outcome of the decision. The same situation is modeled differently in Figure 10. The decisions to activate the outgoing paths are made separately with the help of two conditions. For each path a condition is attached and shown as small diamond leaving the task A. Both situations are semantically equivalent if the conditional flows in Figure 10 incorporate the same conditions as in Figure 9. However, in order to comprehend this information the model viewer has to understand the semantic meaning of the conditional sequence flows. It is advisable to use the alternative shown in Figure 9 as this version is easier to comprehend. Organizations should choose this alternative and enforce it for all process models.



Caption 9: Exclusive Gateway



Caption 10: Implicit Decisions with Conditional Flows

Design Patterns

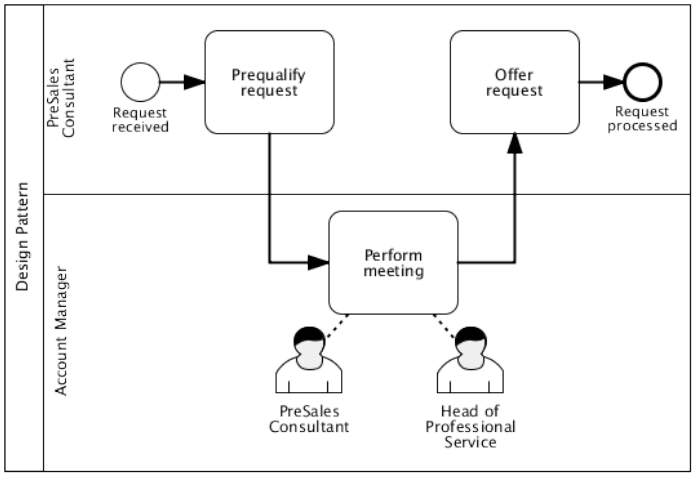
Design patterns are based on specific situations or contexts which should be captured in process models. Design pattern rely on these contexts or situations. Patterns can only be used if the real-world scenario matches the context of the pattern. Thus, process modelers have to check whether they can use design patterns for a given situation. Design patterns describe best practices for specific situations. They show the most intuitive and comprehensible version to model. Best practices are especially useful at the beginning of modeling projects. They help novice modelers to become familiar with process modeling. In certain situations design patterns are also applicable to other contexts. However, process modelers must check these situations to ensure the proper application of design patterns. Modelers must have modeling expertise to perform such checks. With the help of design patterns organizations influence and improve the way modelers capture situations in process models.

Design Patterns – Examples

Design patterns are applicable for specific contexts. The use of design patterns is illustrated in two examples. The first example focuses on the modeling of a meeting where several people participate. Although the context seems trivial, capturing this situation often leads to different process models. Figure 11 shows a design pattern to model such situations. The PreSales Consultant triggers a meeting which is held by the Account Manager. Although the task “Perform Meeting” is shown in the lane of the Account Manager, additional resources participate in the meeting. These additional roles are shown as customized artifacts and associated with the meeting. In total, three roles participate in the meeting as shown in the

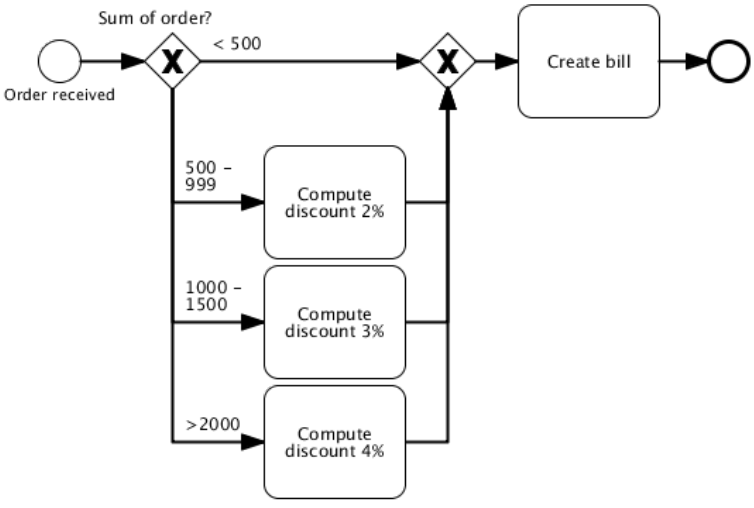
example. The pattern can be reused for other meeting situations or collaborations. It shows an effective way to capture the situation in the model.

Another possibility to capture the meeting in a model is the creation of lanes for all meeting participants. All participants execute the task "perform meeting" in parallel. This representation requires a lane for each participant, the duplication of the meeting task as well as the parallel execution of the task and the corresponding synchronization. Summing up, the design pattern shown in Figure 10 is a superior representation of the meeting context and easier to comprehend.

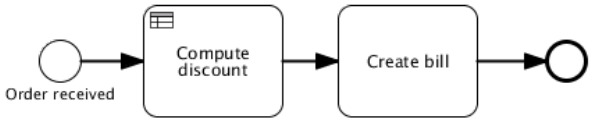


Caption 11: Modeling Meetings

Another example for a design pattern is the use of business rule tasks. Modelers can represent rules explicitly in models as shown in Figure 12. However, in case the rules get updated, extended or deleted, the whole process models must be changed in order to capture the changes. Thus, the process model in Figure 12 should be used with care when depicting business rules. A better representation for business rules is shown in Figure 13. The computation of the discounts is done in the task "Compute discount". The task is marked as a business rule task as described in the BPMN 2.0 specification. The marking means that the rules are stored in an external container, e.g. a file or a table. It must be mentioned that the container is not shown in Figure 13. The model in Figure 13 is easier to read and comprehend than the one in Figure 12. Additionally, business rules can be modified or changed without changing the process models. This becomes especially important if processes contain several business rule tasks. Process models that explicitly capture rules in the model tend to get large and complex. Additionally, model viewers face difficulties to focus on individual rules. The pattern shown in Figure 13 is recommended to avoid these drawbacks. The pattern for business rules can be applied to similar situations where rule tasks are applied.



Caption 12: Business Rules explicitly modeled



Caption 13: Business Rule Task

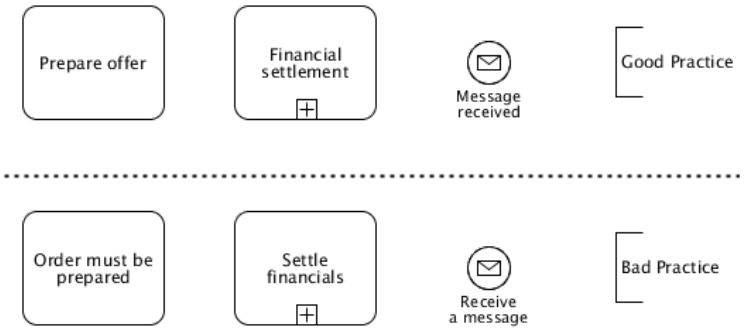
Naming Conventions

Process models exhibit the control flow of processes. In order to add semantic information to symbols, e.g. tasks, they get labeled. The naming usually relies on the process modeler and his naming style. The semantic information given by labels is important when it comes to the verification of process models. The verification checks how well process models capture real-world scenarios. Thus, labels should be present in models. The quality assurance of name tags is important for verification tasks. It is advisable to propose naming conventions to guide the labeling of process elements. The standardization of name tags assists the governance of models.

The labels of process elements are typically chosen according to their actual types. Name tags of tasks differ from those of events. The labels of tasks are chosen with the help of the verb-object style. This style enforces modelers to choose a verb together with a business object. The verb-object style is known to be less ambiguous and easier to understand. Naming conventions describe the styles to propose labels for different element types.

Naming Conventions – Examples

Labeling process elements is an important feature. In practice, the names of elements are often chosen freely which can lead to misinterpretations. Figure 14 shows how labels for different process elements can be chosen. The upper part of the figure depicts labels that can be understood well. The lower part illustrates the same element types but shows labels which should be avoided. Organizations should describe how labels for elements are chosen.



Caption 14: Naming Conventions

HOW TO DEVELOP BEST PRACTICE GUIDELINES?

Best practice guidelines should be individually developed for modeling projects. Each modeling project has its own characteristics, purposes and goals. Modeling projects can largely differ in their purpose, e.g., process automation projects vs. process documentation. Thus, these differences should be taken into account when developing best practice guidelines. It is important to tailor best practices to the unique purposes of process models and the project roles involved. Of course, certain best practices can be used in all modeling projects but must still be checked. The introduction of modeling guidelines should be done by expert modelers who do have large expertise and experience. Best practice guidelines should also be updated if new best practices arise or existing ones change due to project-specific requirements.

In order to develop best practice guidelines you can use the framework described in this paper. Along with the categories you should focus on your individual modeling project and collect best practices to meet your goals. The categories capture characteristics of modeling projects that influence the modeling behavior and, thus, the creation of process models. The categories guide the development of best practice guidelines. The content of the categories should be proposed by the organization and process modelers involved in the project. We suggest to define modeling guidelines before the actual modeling in projects takes place. This avoids misinterpretations especially at the beginning of modeling projects. Modeling guidelines can be updated

SUMMARY

The establishment of modeling guidelines for specific modeling projects requires proper planning. Guidelines should contain best practices that fit the purposes and goals of the actual projects. The introduction of modeling guidelines enables organizations to govern and manage process models as well as process modeling more efficiently. Best practice examples guide process modelers to create models that meet agreed specifications. Especially novice modelers are supported by modeling guidelines. Best practice guidelines further assist the validation of process models which in turn helps to avoid semantic problems in models.

In this paper we presented a framework to develop best practice modeling guidelines. The framework is suitable to establish guidelines that meet the individual purposes of modeling projects. Best practice guidelines for BPMN 2.0 incorporate best practices that often also fit for BPMN 1.x. Existing guidelines for BPMN 1.x can be reused while creating guidelines for the BPMN 2.0. We recommend an

agile development of modeling guidelines. The management and enforcement of best practice guidelines plays an important part in modeling projects. If organizations introduce best practice modeling guidelines for their modeling projects, they have to keep their guideline up to date. If they do so, they gain effective measures to manage their modeling projects more successfully.