

# Towards Automated Process Modeling based on BPMN Diagram Composition

Piotr Wiśniewski, Krzysztof Kluza and Antoni Ligęza

AGH University of Science and Technology  
al. A. Mickiewicza 30, 30-059 Krakow, Poland  
{wpiotr,kluza,ligeza}@agh.edu.pl

**Summary.** Modeling a business process is a complex task which involves different participants who should be familiar with the chosen modeling notation. In this paper, we propose an idea of generating business process models based on a declarative specification. Given an unordered list of process activities along with their input and output data entities, our method generates a synthetic, complete log of a process. The generated task sequences can then serve as an input to a selected process mining method or be processed by an algorithm constructing a BPMN model directly based on the log and additional information included in the declarative process specification.

**Keywords:** business processes, BPMN, automated planning, constraint programming, process mining, business process composition

## 1 Introduction

One of the challenges within the area of business process management is the constant improvement and optimization of business processes. Manual redesign of a workflow is a time-consuming activity which requires close cooperation of a business analyst or a process engineer with a domain expert aware of goods production or service delivery.

As a solution to this problem, the use of a process composition technique is proposed. Composition, as one of the twenty Business Process Management use cases [1], may be regarded as a set of methods which, based on the identified tasks or subprocesses will allow the operator to generate a correct business process model. Our approach uses declarative activity specifications that include initial conditions and execution effects of the process activities, as well as rules for task repetition. Such a specification also focuses on the goal of the modeled process represented by the produced output data.

The proposed approach is based on Business Process Model and Notation (BPMN), which is one of the most widely recognized languages for business workflow modeling. In addition, the composition method presented in this paper is modeled as a Constraint Satisfaction Problem (CSP), which ensures the correct order of tasks and the compliance with business process modeling guidelines. The set of solutions generated by a CSP solver can be then translated into a business

process model using the existing process mining tools or by executing a dedicated algorithm for graph-based model construction.

This position paper is organized as follows: Section 2 presents the state of the art solutions in the area of business process planning as well as the application of constraint programming to process modeling. Section 3 describes our research methodology and techniques being applied in the proposed approach. The composition algorithm presented in Section 4 is followed by concluding remarks and plans for future work included in Section 5.

## 2 Related works

The composition of business process models is present in the literature in combination with different approaches. Process models can be composed using reusable process parts called Relevant Process Fragments [2] which are stored in a component repository. Another approach is the service composition problem, which aims to combine existing functionalities into a new sequence flow. Meyer and Weske [3] proposed the approach that uses a heuristic search to produce a list of possible event logs. Another approach provides a conceptual framework for task composition [4] including structure suggestion and validation regarding task parallelism and preconditions.

The use of automated planning within the area of business process management is dedicated to design and rebuild phases of the modeling procedure. Providing a higher level of automation for a process can be regarded as a way to overcome issues such as constant changes of requirements and unpredictable environmental factors [5]. One of the main challenges of applying common AI planning techniques for service composition is the necessity to consider complex workflow structures such as conditions and loops [6]. Process planning may focus on the goal of the analyzed workflow by finding a set of models whose task postconditions are compliant with the desired final state or use partial-order planning in order to resolve potential concurrency conflicts [7]. Business process planning can also be used to optimize the workflow by removing unnecessary redundancies in branches which follow an exclusive gateway [8].

Business processes are represented by their declarative models which specify relations between tasks and their execution conditions instead of describing the workflow explicitly [9]. According to the research conducted by Mrasek et al. [10], automated generation of a process model from such data significantly increases time efficiency compared to the manual model design. If a process modeling task is defined as a constraint satisfaction problem, this type of process specification can be used to create optimal execution plans [11] as well as to modify actual workflow traces where artifacts appear or relevant events are missing [12].

Our idea also refers to process mining which includes algorithms for generating BPMN models based on event logs. Although mining tools were created to process imperfect data from IT systems, there exist several performance measures which can help to identify the optimal technique for complete logs [13].

### 3 Approach overview

Our research aims to determine how constraint programming may improve the process model generation and to discuss using the process mining for discovering BPMN models based on artificially generated logs.

The proposed method can simplify the process of knowledge acquisition from domain experts who may not be familiar with the appropriate business process modeling notation. Therefore, a user-friendly form of the input process description has been taken into consideration. One of the possible solutions for this problem is based on a spreadsheet specification of the process. However, in the case of process planning, there are no requirements for any ordering of tasks in the specification of the process. The specification includes:

- an unordered list of activities,
- task input and output data entities,
- maximum number of executions for each task,
- initial state of the process,
- a set of final states: one goal state and a number of error states.

Since business process models usually involve multiple participants, parts of the process specification can be created independently by each of the contributors. The reason of basing on a tabular specification is caused by the fact that during the phase of collecting process data the participants may not be aware of the interdependencies between activities performed by different actors of the process. Another idea behind supporting a tabular form of input data is the popularity of spreadsheet editors being accessible by users. This task is performed manually by filling a dedicated form or worksheet.

In the next step, all the files are gathered from the process participants and processed by an automated tool which generates a formal specification, as required by the constraint programming solver. Figure 1 shows a general illustration of the proposed semi-automated business process composition approach.

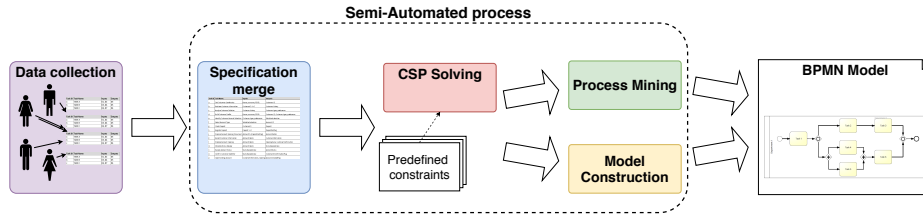


Fig. 1: Method overview (it partly uses the method presented in [14]).

According to our preliminary solution [14], a process model is composed by generating all admissible execution sequences of tasks, based on the input data and the set of predefined constraints. In the current phase of the research, we have included the possibility to represent complex flow structures such as loops and multiple final states. The generated workflow traces are then used as an input to one of the process mining algorithms that generates a workflow net

which is then translated into a process model. Another possible solution is based on constructing a process model by merging the generated workflow traces into an activity graph [15] and transforming it directly into a BPMN model.

## 4 Generating process traces

A business process can be described as a set of activities that produce a specific service or product. Therefore, in the initial phase of process composition, it is necessary to identify the tasks being executed within the workflow. Each task is assigned a set of triggering conditions and generates an effect of its execution. This input and output information can be represented as data entities, defined as variables of the primitive or complex data type.

The proposed approach can be used when a set of tasks and a set of data entities were identified. Then, it is necessary to define the dependencies between these sets. Table 1 presents an illustrative example of a task list which can be used for process composition. Data entities included in parentheses are considered as optional for the corresponding tasks.

Table 1: An example list of tasks and data entities. The maximum number of executions for each task is given in parentheses.

Task Id	Task Name	Task Inputs	Task Outputs
01	<i>Create Offer</i> (1)	RequestForOffer	InitialOffer
02	<i>Include Remarks</i> (1)	OfferReviewed	RemarksIncluded
03	<i>Review Offer</i> (2)	InitialOffer, (RemarksIncluded)	OfferReviewed
04	<i>Send Offer</i> (1)	OfferReviewed, (RemarksIncluded)	OfferSent

If  $m$  is the number of data entities in the modeled process, an  $m$ -dimensional vector has to be defined along with the process model. The initial state vector explicitly indicates which data entities are present before the process execution. The other structure to be defined is the final state matrix which reflects the possible combinations of data entities after the process end event. A process should contain exactly one goal state and several other final states which represent error and terminate end events. At the current stage of the research, the following constraints were proposed to ensure the correctness of the process:

1. The number of executions for each task should be lower than or equal to the value included in the specification.
2. The maximum length of the workflow is equal to the number of defined tasks multiplied by their number of executions.
3. Data entities required for the first executed task satisfy the initial state.
4. If the goal state is achieved, then the process ends.
5. The output data entities of the last task satisfy one of the final states.
6. A task can be executed when the current state satisfies the input conditions.
7. The presence of data entities can be changed only by a task execution.

The synthetic workflow log is generated using the Gecode solver with the assumption that each solution of the CSP is a different trace. Given the simple specification presented in Table 1 and a final state represented by data entity `OfferSent`, a synthetic log of two distinct traces was generated:

$$W_S = \{\{01, 03, 04\}, \{01, 03, 02, 03, 04\}\}. \quad (1)$$

In the next step, the list of generated solutions was converted to an XES file and processed in ProM environment using ILP miner. The result of process mining algorithm is shown in Figure 2.

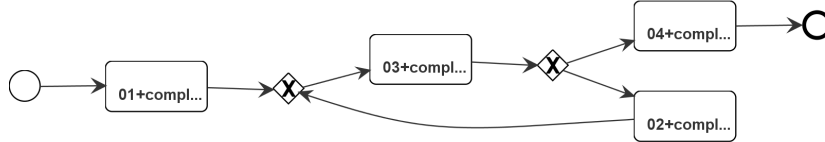


Fig. 2: The mined BPMN model representing the generated log.

## 5 Conclusion and Future Works

The purpose of this paper is to give an overview of the composition approach and to trigger a discussion on the concept of automated process modeling based on the existing approaches such as declarative languages, workflow trace generation, process mining and process constructing algorithms.

In this paper, we briefly discussed the concept of business process composition based on a partially structured specification. As a contrast to many existing process planning techniques, our method does not require the ordering relations of tasks being declared explicitly. Thus, in this approach, no knowledge of specific notation and modeling guidelines is needed to design business process models.

Since we based our method on different phases, it is more flexible and it may be controlled in different stages. For example, inconsistencies in the specification may be discovered before the final process model is generated, e.g. if the constraints are unsatisfiable. Every part of the method can be analyzed separately and replaced with another algorithm.

In the further development, we plan to evaluate other approaches such as Answer Set Programming and perform a comparative analysis of implementable techniques. In addition, we would like to analyze and evaluate the suitability of different process mining techniques for processing the generated event logs. These results are going to be compared with the dedicated algorithm for constructing BPMN models in terms of model fitness and the ability to discover complex workflow structures.

## References

1. van der Aalst, W.M.P.: Business Process Management: A Comprehensive Survey. *ISRN Software Engineering* **2013** (2013)
2. Skouradaki, M., Andrikopoulos, V., Leymann, F.: Representative BPMN 2.0 Process Models Generation from Recurring Structures. In: Proceedings of the 23rd IEEE International Conference on Web Services, IEEE (June 2016) 468–475
3. Meyer, H., Weske, M.: Automated service composition using heuristic search. In Dustdar, S., Fiadeiro, J.L., Sheth, A.P., eds.: *Business Process Management: 4th International Conference, BPM 2006, Vienna, Austria, September 5-7, 2006. Proceedings*, Berlin, Heidelberg, Springer Berlin Heidelberg (2006) 81–96
4. Weber, I.M.: *Semantic methods for execution-level business process modeling: Modeling Support through Process Verification and Service Composition*. Volume 40. Springer (2009)
5. Marrella, A.: What automated planning can do for business process management. In Teniente, E., Weidlich, M., eds.: *Business Process Management Workshops*, Cham, Springer (2018) 7–19
6. AlSedrani, A., Tourir, A.: Web service composition processes: A comparative study. *International Journal on Web Service Computing (IJWSC)* **7**(1) (2016)
7. Marrella, A., Lespérance, Y.: A planning approach to the automated synthesis of template-based process models. *Service Oriented Computing and Applications* **11**(4) (Dec 2017) 367–392
8. Heinrich, B., Schön, D.: Automated planning of process models: the construction of simple merges. In: *European Conference on Information Systems (ECIS), Research Papers*. (2016)
9. Barba, I., Del Valle, C., Weber, B., Jimenez, A.: Automatic generation of optimized business process models from constraint-based specifications. *International Journal of Cooperative Information Systems* **22**(02) (2013) 1350009
10. Mrasek, R., Mülle, J., Böhm, K.: Automatic generation of optimized process models from declarative specifications. In Zdravkovic, J., Kirikova, M., Johannesson, P., eds.: *Advanced Information Systems Engineering*, Cham, Springer (2015) 382–397
11. Barba Rodriguez, I.: *Constraint-based Planning and Scheduling Techniques for the Optimized Management of Business Processes*. Phd thesis, Universidad de Sevilla (2011)
12. De Giacomo, G., Maggi, F., Marella, A., Sardina, S.: Computing trace alignment against declarative process models through planning. In: *International Conference on Automated Planning and Scheduling (ICAPS 2016)*. (2016) 367–375
13. Buijs, J.C., Van Dongen, B.F., van Der Aalst, W.M.: On the role of fitness, precision, generalization and simplicity in process discovery. In: *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, Springer (2012) 305–322
14. Wiśniewski, P., Kluza, K., Ślażyński, M., Ligeża, A.: Constraint-based composition of business process models. In Teniente, E., Weidlich, M., eds.: *Business Process Management Workshops*, Cham, Springer International Publishing (2018) 133–141
15. Wiśniewski, P., Ligeża, A.: Constraint-based identification of complex gateway structures in business process models. In: *International Conference on Artificial Intelligence and Soft Computing*, Springer (2018) 788–798

This is a pre-print of the paper accepted for presentation at the  
6<sup>th</sup> Workshop on  
Declarative/Decision/Hybrid Mining and Modelling for Business Processes  
(DeHMiMoP 2018)

to appear in the  
BPM 2018 Workshop Proceedings  
Eds. Florian Daniel, Hamid Motahari, Michael Sheng  
published by Springer in the book series  
Lecture Notes in Business Information Processing (LNBIP)