Information and Software Technology 57 (2015) 571–594

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Generating optimized configurable business process models in scenarios subject to uncertainty

Andrés Jiménez-Ramírez [a,*], Barbara Weber [b], Irene Barba [a], Carmelo Del Valle [a]

[a] University of Seville, Dpto. Lenguajes y Sistemas Informáticos, Spain
[b] University of Innsbruck, Department of Computer Science, Austria

ARTICLE INFO

Article history:
Received 14 October 2013
Received in revised form 8 June 2014
Accepted 9 June 2014
Available online 2 July 2014

Keywords:
Configurable business process models
Constraint-based business process models
Flexibility
Robustness
Planning and scheduling
Constraint programming

ABSTRACT

Context: The quality of business process models (i.e., software artifacts that capture the relations between the organizational units of a business) is essential for enhancing the management of business processes. However, such modeling is typically carried out manually. This is already challenging and time consuming when (1) input uncertainty exists, (2) activities are related, and (3) resource allocation has to be considered. When including optimization requirements regarding flexibility and robustness it becomes even more complicated potentially resulting into non-optimized models, errors, and lack of flexibility.
Objective: To facilitate the human work and to improve the resulting models in scenarios subject to uncertainty, we propose a software-supported approach for automatically creating configurable business process models from declarative specifications considering all the aforementioned requirements.
Method: First, the scenario is modeled through a declarative language which allows the analysts to specify its variability and uncertainty. Thereafter, a set of optimized enactment plans (each one representing a potential execution alternative) are generated from such a model considering the input uncertainty. Finally, to deal with this uncertainty during run-time, a flexible configurable business process model is created from these plans.
Results: To validate the proposed approach, we conduct a case study based on a real business which is subject to uncertainty. Results indicate that our approach improves the actual performance of the business and that the generated models support most of the uncertainty inherent to the business.
Conclusions: The proposed approach automatically selects the best part of the variability of a declarative specification. Unlike existing approaches, our approach considers input uncertainty, the optimization of multiple objective functions, as well as the resource and the control-flow perspectives. However, our approach also presents a few limitations: (1) it is focused on the control-flow and the data perspective is only partially addressed and (2) model attributes need to be estimated.

1. Introduction

A Business Process (BP) can be defined as a set of activities which are performed in coordination in an organization to achieve a business goal [93]. These activities can be manual activities, other BPs, or even pieces of software. Nowadays, in order to support BPs, BP Management (BPM) embraces methods, techniques, and software to design, enact, control, and analyze operational processes involving humans, organizations, applications, and other sources of information [90]. Such management generally follows a strict methodology to ensure the quality of the information systems which are created. Typically, the traditional BPM life cycle [93] includes four phases, i.e., process design & analysis (i.e., a design of the BP is created following the requirements), system configuration (i.e., the software defined in the BP design is implemented), process enactment (i.e., the software is executed following the BP design) and evaluation (i.e., monitoring information or logs are analyzed to look for design improvements) [93].

The quality of a BP design has a great influence on all the phases of the BPM life cycle and it is essential for BP improvement, which has been ranked as the number one priority for top management by the 2010 Gartner survey [36].

* Corresponding author. Tel.: +34 606772495.
E-mail addresses: ajramirez@us.es (A. Jiménez-Ramírez), barbara.weber@uibk.ac.at (B. Weber), irenebr@us.es (I. Barba), carmelo@us.es (C. Del Valle).

http://dx.doi.org/10.1016/j.infsof.2014.06.006
0950-5849/

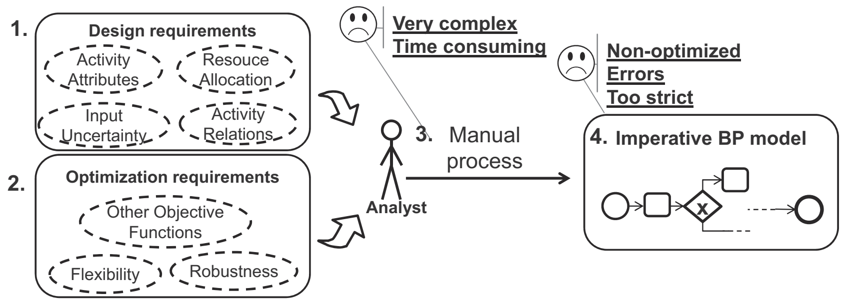

Fig. 1. Motivation overview.

1.1. Problem statement

In the process design & analysis phase the BP models are typically specified by hand using imperative languages like EPC or BPMN [12]. This way, a precise activity sequence which establishes how a given set of activities has to be performed is defined. Such a sequence typically includes temporal relations between activities or even dependencies with input data. Typically, such activities are related to a set of attributes (e.g., duration and cost) which need to be estimated. Furthermore, in many real scenarios such estimates might be subject to input uncertainty (e.g., non-punctual clients imply uncertainty in their arrival times) [84]. In addition, the BP performance can be greatly influenced by how the different resources are assigned to each activity that has to be performed [45,81]. Therefore, regarding such scenarios and motivated by the case study described in the paper, when designing a BP model, analysts have to face certain design requirements (cf. Fig. 1 (1)), such as:

1. Dealing with activity attributes and their estimated values.
2. Managing the input uncertainty which exists in many real scenarios [84] in which providing a range of possible values for a BP property is most reliable that providing an exact value which may be difficult to know. For example, the arrival time of clients can be considered uncertain due to unpunctual clients.
3. Dealing with relations between the activities, i.e., control-flow as well as temporal and data constraints of the BP.[1]
4. Considering resource allocation.

Since uncertain scenarios are considered, managing such input uncertainty becomes necessary. For this, flexibility and robustness are proposed since we consider this is the best way to properly address the considered uncertainty. The situation is further complicated if the aforementioned design requirements have to be addressed along with optimizing some (potentially) conflicting objective functions (e.g., time and cost). Such optimization requirements (cf. Fig. 1 (2)) can be summarized as follows:

1. Flexibility, i.e., the capability to adapt to input uncertainty [34,66,78]. For this, designed models should consider different execution alternatives to support such uncertain scenarios [93].
2. Robustness, i.e., the capability to withstand the uncertainty to some extent [18,20,27]. For his, BP models should be designed to avoid making unnecessary adaptations which typically are costly.
3. Other objective functions are commonly considered since the BP design usually involves a trade-off between different quality dimensions which may be in conflict or be opposed [67].

This task of creating a BP design can form a very complex problem and be very time consuming (cf. Fig. 1 (3)). For this, methods and tools for supporting analysts during the BP design are becoming more and more important. Moreover, the resulting models may be non-optimized, potentially contain errors, and might be too strict [29,58,94] (cf. Fig. 1 (4)).

In such context, there exist some proposals for generating BP models or that could be extended in such direction (cf. [29,40,47,53,60,63,64,75,94]). These proposals are based on generating a single execution plan which fulfills all the BP constraints starting with a constraint-based specification. This plan could be, in turn, used for the generation of an imperative model. As a major drawback of existing proposals, considering only one single execution plan unnecessarily restricts the flexibility of the resulting imperative model. Thus, to the best of our knowledge, the existing proposals are not sufficient to address all the previously mentioned requirements, e.g., dealing with the flexibility needs of existing BPs [66].

1.2. Contribution

In order to facilitate the human work which is involved in the process design & analysis phase and to improve the resulting imperative BP models we propose a method for automatically creating configurable BP models (i.e, a modeling artifact that captures a family of process models in an integrated manner) [22] from declarative specifications [29] (cf. Fig. 2). The proposed approach considers all the aforementioned requirements which have to be considered when creating a suitable BP model, i.e., activity attributes, resource allocation, input uncertainty, relation between activities, optimization of several objective functions, as well as flexibility and robustness issues, and is detailed in the following.

Declarative models are typically easier to specify and less time-consuming than imperative models in scenarios where high variability is required [29]. Therefore, we propose to use a declarative specification as starting point of the proposed approach. For this, the Declare language[2] [63] is used as basis, since it allows the specification of BP activities together with the constraints which must be satisfied for correct BP enactment and for the goal to be achieved. We extend Declare in order to widen its design flexibility by considering stochastic values for modeling the uncertainty of the scenario (as required in the considered problems, cf. Section 1.1), resulting in the SDeclare language. To be more precise, with the proposed extension, some properties of a BP (such as activity attributes, data and temporal constraints, and resource availability) can be expressed

[1] Note that the considered scenarios are focused on the control-flow and the resource perspectives of the BPs and the data perspective is only partially considered.

[2] Declare is one of the most referenced and used declarative BP languages in the context of BPM.