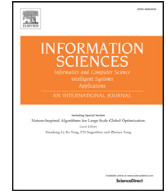


Contents lists available at [ScienceDirect](#)

Information Sciences

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

Formal verification of complex business processes based on high-level Petri nets



Ahmed Kheldoun^a, Kamel Barkaoui^{b,*}, Malika Ioualalen^a

^aMOVEP, Computer Science Department, USTHB, Algiers, Algeria

^bCEDRIC-CNAM, 292 Rue Saint-Martin 75141, Cedex 03 Paris, France

ARTICLE INFO

Article history:

Received 23 October 2015

Revised 28 November 2016

Accepted 31 December 2016

Available online 31 December 2016

Keywords:

Business process modelling

BPMN

RECATNets

Conditional rewriting logic

Maude language and tool

ABSTRACT

The Business Process Modeling Notation (BPMN) has been widely used as a tool for business process modeling. However, BPMN suffers from a lack of standard formal semantics. This weakness can lead to inconsistencies, ambiguities, and incompletenesses within the developed models. In this paper, we propose a formal semantics of BPMN using recursive ECATNets. Owing to this formalism, a large set of BPMN features such as cancellation, multiple instantiation of subprocesses and exception handling can be covered while taking into account the data flow aspect. The benefits and usefulness of this modelling are illustrated through three examples. Moreover, since recursive ECATNets semantics is defined in terms of conditional rewriting logic, one can use the Maude LTL model checker to verify several behavioral properties related to BPMN models. The work presented in this document has been automated through the development of a first prototype.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

1.1. Overview and motivation

The standard Business Process Modeling Notation (BPMN) [19] has been established as the de-facto standard for modeling business processes. It provides a standard notation easily understandable that supports the business process management while being able to represent complex processes semantics. Nevertheless, despite the various advantages of BPMN, it suffers from a lack of formal semantics which can lead to inconsistencies, ambiguities, and incompletenesses within the developed models. Furthermore, BPMN brings additional features drawn from a range of sources including Workflow Patterns [22] which are able to define: (1) subprocesses that may be executed multiple times concurrently; and (2) subprocesses that may be interrupted as a result of exceptions. These features increase the types of semantic errors that can be found in BPMN models. As a result, many researchers proposed formal methods to build formal description and verification models of business processes. However, one of the weaknesses of these proposals is their lack of support for modeling complex BPMN business processes involving exception, cancellation and multiple instantiation of subprocesses. For that, we need an expressive modeling formalism that allows, on one hand, to specify their dynamic structure, and on the other hand, to check the control-flow correctness of these business processes while taking into account their data flow aspect.

* Corresponding author.

E-mail address: kamel.barkaoui@cnam.fr (K. Barkaoui).

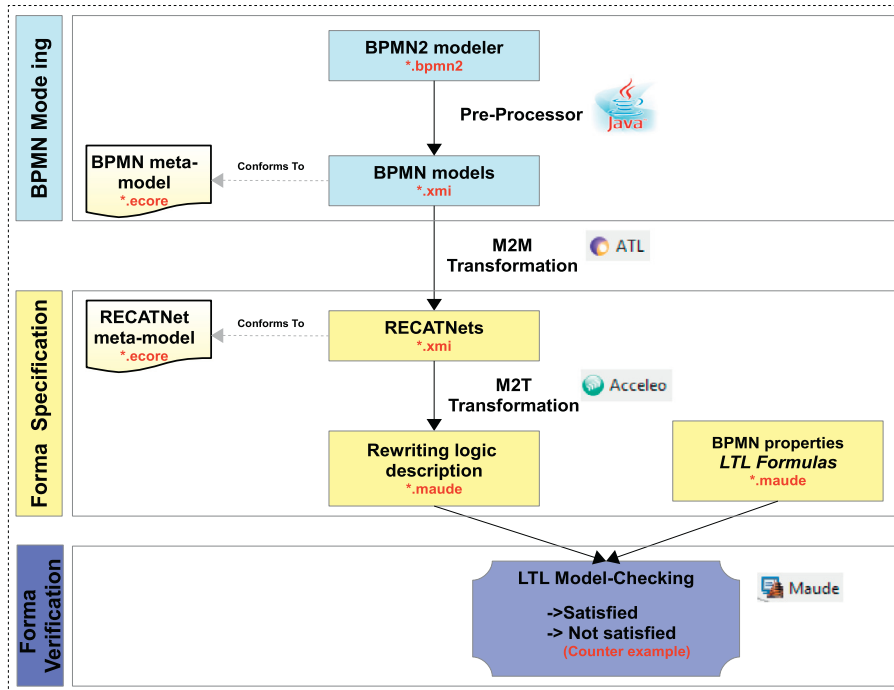


Fig. 1. Overview of the proposed approach.

In this paper, we propose the use of Recursive ECATNets [3] to cope with the formal verification of complex BPMN models. The Recursive ECATNets model offers practical mechanisms for handling the most advanced BPMN constructs (involving exception, cancellation and multiple instances). Since Recursive ECATNets semantics can be expressed in terms of conditional rewriting logic [17], we can use the Maude LTL model checker [8] to investigate several behavioral properties of business processes.

1.2. Proposed approach

The proposed approach for the verification of BPMN models is based mainly on Meta-modeling and Model Transformations. It is achieved automatically into four steps : (1) transformation of BPMN models created by a graphical editor like eclipse BPMN2 Modeler (URL: <http://eclipse.org/bpmn2-modeler/>) into XML Metadata Interchange (XMI) files conforming to our proposed BPMN meta-model, (2) transformation of obtained BPMN XMI files into RECATNets using the ATLAS Transformation Language (ATL) [13] where two meta-models are defined for BPMN and RECATNet (URL: <http://recatnets.cnam.fr/>), (3) transformation of obtained RECATNets into rewriting logic description using the transformation tool Acceleo (URL: <http://www.eclipse.org/acceleo/>), (4) checking the properties of business processes expressed as rewrite theories by using the Maude LTL model checker. This is summarized in Fig. 1.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 provides an overview about BPMN and Recursive ECATNets. Section 4 presents the mapping rules from BPMN to Recursive ECATNet. Section 5 shows two examples for the proposed mapping. Section 6 presents the formal semantics of the mapping rules. Section 7 presents the RECATNet semantics in terms of rewriting logic. Section 8 presents the developed prototype. Section 9 presents an overview of the verification process using the Maude LTL model checker. Finally, Section 10 concludes and gives some further research directions.

2. Related work

Many researchers have tried to deal with formal modeling and verification of business processes using BPMN models. Morimoto [18] presents an extended survey of the existing verification techniques of BPMN diagrams and compares them with each other with respect to motivations and methods. Nevertheless, none of the cited works take into account the following key features of BPMN : (1) cancellation of subprocesses; (2) parallel multi-instance subprocesses; and (3) exception handling in the context of subprocesses that are executed multiple times concurrently.

Petri nets often are a topic in verification of business processes using BPMN models. Dijkman et al. [6] propose a mapping from a core set of BPMN to labelled Petri nets. This output is represented in the PNML language [4] and can subsequently be used to verify BPMN processes by using the open source tool WofBPEL [20]. The proposed mapping for exception handling is

Download English Version:

<https://daneshyari.com/en/article/4944724>

Download Persian Version:

<https://daneshyari.com/article/4944724>

[Daneshyari.com](https://daneshyari.com)