Formal Analysis of BPMN Models Using Event-B

Jeremy W. Bryans¹ and Wei Wei²

¹ School of Computing Science, Newcastle University, United Kingdom
Jeremy.Bryans@ncl.ac.uk
² SAP Research CEC Darmstadt, SAP AG, Bleichstr. 8, 64283 Darmstadt, Germany
wei01.wei@sap.com

Abstract. The use of business process models has gone far beyond documenta-
tion purposes. In the development of business applications, they can play the role of an artifact on which high level properties can be verified and design errors can be revealed in an effort to reduce overhead at later software development and diagnosis stages. This paper demonstrates how formal verification may add value to the specification, design and development of business process models in an industrial setting. The analysis of these models is achieved via an algorithmic translation from the de-facto standard business process modeling language BPMN to Event-B, a widely used formal language supported by the Rodin platform which offers a range of simulation and verification technologies.

Keywords: business process modelling, verification, BPMN, Event-B.

1 Introduction

Complex, large-scale business information systems are critical to the successful operation of many businesses, and SAP is a leading provider of such systems. Business process modeling has become increasingly important to the development of enterprise software applications [13]. Nowadays, business applications are usually built by integrating a broad range of highly configurable software components and services, which can be rapidly tailored to satisfy different and constantly changing business needs. Business process models are used to describe such integration scenarios and their work flows, facilitating an intuitive common understanding of the business logic between customers and developers. In addition to their use as documentation, business process models can also be simulated, analyzed and verified to reveal design errors at an early stage in software development. This promises to enhance the efficiency of reaching high-quality software solutions and can save substantial implementation and diagnosis costs which would otherwise be incurred at later development phases.

We wish to use formal methods to improve the quality of business process models within a software design process, and also aim to reduce the extra burden that formal methods induce on designers and developers. Within the context of the DEPLOY project¹, we choose the Event-B modeling formalism [1] and the

¹ www.deploy-project.eu
Rodin platform \cite{2} in our pursuit of these goals. The choice is also encouraged by our past successful experiences of using Event-B for describing and analyzing business applications \cite{5,6}. Event-B offers many indispensable features for analyzing business process models such as the ability to model data. The Rodin platform is empowered by a large number of plug-ins providing various analysis capabilities like specialized provers, model checking, and simulation.

This paper examines our recent work on the formal analysis of business process models using Event-B and Rodin, and discusses the impact of the analysis results on software design and development. We also investigate the potential to largely automate these analyses in order to pave the way for future industrial deployment. We designed an algorithmic translation from BPMN, the de-facto standard business process modeling language, to Event-B. The translation covers most of the commonly used BPMN features, also including features newly introduced in the proposed draft of the second version of the language \cite{15}. We also make the Event-B translation structurally faithful to the original BPMN model, which not only improves readability, but also enhances provability and analyzability.

Outline. In Section 2 we briefly introduce BPMN and Event-B, and in Section 3 sketch our translation from BPMN to Event-B. Sections 4 and 5 describe two case studies to illustrate how formal analysis is performed on the Event-B translations of BPMN models, and also discuss the possibility of automating these analysis procedures. Related work is discussed in Section 6 before we conclude in Section 7 with a discussion of our next steps. Due to space constraints, we have moved some of the Event-B code and discussion into appendices.

2 Background

BPMN. We introduce the Business Process Modeling Notation (BPNM) elements we use in this paper. We only show syntactic compositions here. The semantics of syntactic elements will be discussed later when we explain how they are translated.

A typical BPMN model consists of one or more pools, each representing a collaboration partner (such as FACTORY and WORKER in Figure 1). Each pool usually contains a top process. A process contains flow objects and the connections between them. Flow objects include events, gateways and activities. Events either throw or catch triggers and are represented as circles containing a marker indicating the kind of trigger. Gateways converge or diverge control flows and are represented as diamonds. An activity can be either an atomic task or a composite sub-process that contains an inner process. An activity can be a loop. Activities are graphically represented as rounded rectangles. A process may contain data items as process instance attributes. There are also data stores that are process-independent and globally accessible.

Two pools communicate with each other mainly by exchanging messages, which may carry data fields. Message flows are represented as directed dotted