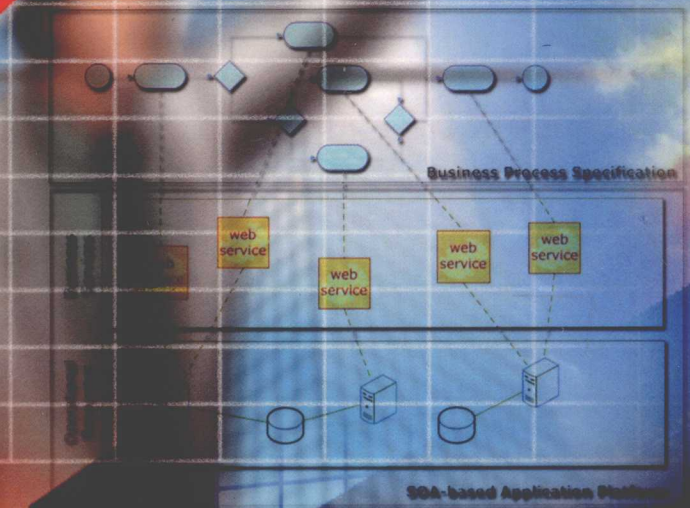
Business Process Modeling

Software Engineering, Analysis and Applications



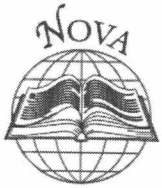
Jason A. Beckmann
Editor

Business Issues, Competition and Entrepreneurship

NOVA

BUSINESS PROCESS MODELING: SOFTWARE ENGINEERING, ANALYSIS AND APPLICATIONS

JASON A. BECKMANN
EDITOR



Nova Science Publishers, Inc.
New York

LIBRARY OF CONGRESS CATALOGING-IN-PUBLICATION DATA

Business process modeling : software engineering, analysis and applications
/ editors, Jason A. Beckmann.
p. cm.
Includes index.
ISBN 978-1-61209-344-4 (hbk.)
1. Management information systems. 2. Business--Data processing. 3. Workflow--Management. I. Beckmann, Jason A.
HD30.2.B8785 2010
658.4'034--dc22

2010051515

Published by Nova Science Publishers, Inc. † New York

BUSINESS ISSUES, COMPETITION AND ENTREPRENEURSHIP

BUSINESS PROCESS MODELING: SOFTWARE ENGINEERING, ANALYSIS AND APPLICATIONS

BUSINESS ISSUES, COMPETITION AND ENTREPRENEURSHIP

Additional books in this series can be found on Nova's website under the Series tab.

Additional E-books in this series can be found on Nova's website under the E-books tab.

PREFACE

Business process modeling (BPM) is the activity of representing processes of an enterprise so that the current process may be analyzed and improved. BPM is typically performed by business analysts and managers who are seeking to improve process efficiency and quality. This book presents current research in the study of business process modeling, including BPM and automation with general and domain specific languages; conceptualizing, analyzing and communicating the business model and context-aware methods for process modeling.

Chapter 1 - Business Process Management (BPM) is an holistic approach for describing, analyzing, executing, managing and improving large enterprise business processes, which can be seen as collections of related tasks executed to accomplish well-defined goals.

This chapter focuses on the description and analysis of business processes. In particular, the chapter introduces a notation for the description of a business process in terms of both functional and non-functional properties. Such a description is then used to carry out the predictive analysis of the business process behaviour. The chapter specifically addresses the performance and reliability prediction of a business process by use of a joint measure known as performability.

In the BPM context, the Business Process Modeling Notation (BPMN) is the de-facto standard for the high-level description of business processes. Unfortunately BPMN does not support the characterization of the business process in terms of non-functional properties such as performance and reliability. To overcome such limitation, this chapter introduces PyBPMN (Performability-enabled BPMN), a lightweight BPMN extension for the specification of performability properties.

The proposed extension is based on an approach that exploits principles and standards introduced by MDA (Model Driven Architecture). In particular, the BPMN extension is carried out by first specifying the BPMN metamodel and then obtaining the PyBPMN metamodel by adding the metaclasses that define the specific performance and reliability characteristics.

The tasks that define a business process can be carried out either by human operators or by automated software services. This chapter specifically focuses on fully automated business processes that are defined and executed as orchestrations of software services. In this respect, PyBPMN can be used to describe the performance and reliability properties of both a business process and its constituent services.

This chapter also introduces a model-driven method that makes use of PyBPMN to predict, at design time, the performability of a business process, either to select the configuration of services that provide the best level of performability or to check if a given configuration satisfies the overall requirements of the business process. The proposed method can be fully automated, thus allowing business analysts to carry out the performability prediction with no extra effort and without being required to own specific skills of performability theory, as shown by use of an example case study.

Chapter 2 - A business model is a sustainable way of doing business. Here sustainability stresses the ambition to survive even harsh business landscapes and create profits in the long run. Whether profits are retained by the shareholders or distributed in some degree to a broader mass of stakeholders is not the focus here. Rather, it is the point of this paper to illustrate how one may go about conceptualizing, analyzing and communicating the business model of a company. A business model describes the coherence in the strategic choices which makes possible the handling of the processes and relations which create value on both the operational, tactical and strategic levels in the organization. The business model is the platform which connects resources, processes and the supply of a service, which results in the fact that the company is profitable in the long term. Conceptualizing the business model is therefore concerned with identifying this platform, while analyzing it is concerned with gaining an understanding of precisely which levers of control are apt to deliver the value proposition of the company. Finally, communicating the business model is concerned with identifying the most important performance measures, both absolute and relative measures, and relating them to the overall value creation story.

Chapter 3 - The role of business process models, as models in general, has been considerably changed from describing scenarios (contemplative models)

towards actual coordinating activity execution (productive models) and from technical expert privilege to domain expert routine task.

Therefore, the challenge today is to reconcile two requirements: i) provide high level formalisms to domain experts, for the definition of models, and ii) execute these models, by mapping the abstract activities on various pieces of code and implementation artifacts. Automation of business processes, which is a pillar of nowadays frameworks evolvability, requires filling the gap between high level formalisms and execution. How this is performed is the topic of our chapter, which analyzes two approaches: the former based on large, general modeling languages, adopted as standards, and the latter based on small and composable domain specific languages. Examples of architectures driven by these process languages are presented, and solutions applied for the necessary transformations between different layers of abstraction are described.

Chapter 4 - This chapter proposes a view of business processes as designed artefacts that are ontologically no different than artefacts in domains such as mechanical and software engineering. This view distinguishes three concerns for designing processes: architecture, implementation and adaptation. The authors show that current process modelling approaches conflate these aspects, often leading to high complexity and inflexibility of the resulting process models. The authors use a generalisation of the "feature" concept in engineering design, represented using the function-behaviour-structure (FBS) ontology, as the basis of a new approach to concisely specifying business process architectures that allow for more process flexibility.

Chapter 5 - Recent studies have started to explore context-awareness as a driver in the design of adaptable business processes. The emerging challenge of identifying and considering contextual drivers in the environment of a business process are well understood, however, typical methods used in business process modeling do not yet consider this additional contextual information in their process designs. In this chapter, the authors describe our research towards innovative and advanced process modeling methods that include mechanisms to incorporate relevant contextual drivers and their impacts on business processes in process design models. The authors report on our ongoing work with an Australian insurance provider and describe the design science the authors employed to develop these innovative and useful artifacts as part of a context-aware method framework. The authors discuss the utility of these artifacts in an application in the claims handling process at the case organization.

Chapter 6 - Visual business process representation languages such as BPMN are a useful tool for specification of business processes. However,

practical verification and execution of Business Process Models is a challenging task. One solution to this problem is integration of business processes with business rules, which provides a flexible runtime environment. This chapter concerns Business Process Models as a visual inference specification method for modularized rulebases. To provide the background for this approach, selected analysis and execution methods for Business Processes, such as BPEL and BPMN tools, are presented. Business Processes can be supported with Business Rules as executable logic. Rule-Based Systems have well-established methods for verification and optimization. This chapter presents selected rule-based solutions, such as Drools and XTT2 – a novel visual rule specification that provides formalized analysis – as well as their integration with BPMN as a visual method for inference specification. The proposed BPMN+XTT2 solution combines flexible business processmodeling provided by BPMN with verification and execution features of XTT2.

Chapter 7 - Business process modeling is an important stage in Business Process Management (BPM). There are many tools and methodologies to do such modeling. However, few of them can provide an optimal planning as well as the performance evaluation in a versatile environment. Since the globalized economics changes very fast, the process dynamics for each firm also changes day by day. Our approach can provide companies a helpful toolkit to manage such situation while keeping the ability to predict the performance of each business process. Due to IT advances in Enterprise Resource Planning Systems (ERPS), more and more companies adapted Service-Oriented Architecture (SOA) as the main infrastructure of their core business operations. So, process is the key identity of business activity to be monitored. However, how to capacitate (man or machine) as well as to evaluate the performance of such process, especially in a volatile environment, so that the business goal can be fulfilled is still unknow n to most of the mangers. This chapter proposes an integrated method to do both optimal capacity planning and performance evaluation for the companies facing people floating as well as system failures while keeping the capability to predict the performance of such process. So, the planning decision can be verified. A numerical example is illustrated for the proposed method.

CONTENTS

In: Business Process Modeling
Editor: Jason A. Beckmann

ISBN: 978-1-61209-344-4

Chapter 1

PERFORMABILITY-ORIENTED DESCRIPTION AND ANALYSIS OF BUSINESS PROCESSES

Paolo Bocciarelli and Andrea D'Ambrogio*
University of Rome "Tor Vergata", Rome, Italy

ABSTRACT

Business Process Management (BPM) is an holistic approach for describing, analyzing, executing, managing and improving large enterprise business processes, which can be seen as collections of related tasks executed to accomplish well-defined goals.

This chapter focuses on the description and analysis of business processes. In particular, the chapter introduces a notation for the description of a business process in terms of both functional and non-functional properties. Such a description is then used to carry out the predictive analysis of the business process behaviour. The chapter specifically addresses the performance and reliability prediction of a business process by use of a joint measure known as performability.

In the BPM context, the Business Process Modeling Notation (BPMN) is the de-facto standard for the high-level description of business processes. Unfortunately BPMN does not support the characterization of the business process in terms of non-functional properties such as performance and reliability. To overcome such limitation, this chapter

* {bocciarelli, dambro}@info.uniroma2.it

introduces PyBPMN (Performability-enabled BPMN), a lightweight BPMN extension for the specification of performability properties.

The proposed extension is based on an approach that exploits principles and standards introduced by MDA (Model Driven Architecture). In particular, the BPMN extension is carried out by first specifying the BPMN metamodel and then obtaining the PyBPMN metamodel by adding the metaclasses that define the specific performance and reliability characteristics.

The tasks that define a business process can be carried out either by human operators or by automated software services. This chapter specifically focuses on fully automated business processes that are defined and executed as orchestrations of software services. In this respect, PyBPMN can be used to describe the performance and reliability properties of both a business process and its constituent services.

This chapter also introduces a model-driven method that makes use of PyBPMN to predict, at design time, the performability of a business process, either to select the configuration of services that provide the best level of performability or to check if a given configuration satisfies the overall requirements of the business process. The proposed method can be fully automated, thus allowing business analysts to carry out the performability prediction with no extra effort and without being required to own specific skills of performability theory, as shown by use of an example case study.

1. INTRODUCTION

Business Process Management (BPM) is an holistic approach for describing, analyzing, executing, managing and improving large enterprise business processes, which can be seen as collections of related tasks executed to accomplish well-defined goals [4][31].

The term BPM is generally referred to the several activities that define the business process lifecycle [36]. This chapter specifically addresses the description of a business process and the use of such description for analyzing and predicting the process behaviour.

In the BPM context, the Business Process Modeling Notation (BPMN) is a standard, promoted by the Business Process Management Initiative (BPMI), for the high-level representation of business processes [21]. The main objective of the BPMI effort is the definition of a notation that has to be easily readable for the different people involved in the business process automation, namely, business analysts and designers who specify the business process and technical developers who implement the specified process.

Unfortunately, BPMN does not support the characterization of the business process in terms of non functional properties, such as performance and reliability. BPMN descriptions do not contain neither the specification of overall performance and reliability constraints (e.g., the response time associated to the entire business process execution, the process reliability, etc.), nor the specification of task properties (e.g., the mean time to failure of a single process task). This means that BPMN can be effectively used to define only the functional properties of a business process, without providing the ability to specify non-functional properties.

To overcome such limitation, this chapter introduces PyBPMN (Performability-enabled BPMN), a lightweight BPMN extension that addresses the specification of performance and reliability properties of a business process. The term performability is used to denote a joint property that combines performance and reliability [17][38].

The proposed extension is carried out according to principles and standards introduced in the model-driven engineering field and is specifically founded on the Meta Object Facility (MOF) [25], the standard for specifying technology neutral metamodels, and the XML Metadata Interchange (XMI) [24], the standard for serializing models and metamodels into XML documents and schemas, respectively. MOF and XMI are at the heart of the Model Driven Architecture (MDA), the OMG's incarnation of model-driven engineering principles [23].

The BPMN metamodel [22] is extended by adding the metaclasses needed to describe the performance and reliability properties, in order to obtain the novel PyBPMN metamodel. Both the BPMN metamodel and the PyBPMN metamodel are defined by use of MOF constructs. The PyBPMN XML Schema is then derived from the PyBPMN metamodel by use of XMI-based transformation rules.

The tasks that define a business process can be carried out either by human operators or by automated software services. This chapter addresses fully automated business processes that are defined and executed as orchestrations of software services. In this respect, PyBPMN can be used to describe the performability properties of both a business process and its constituent services.

The ability to specify the performability properties in the description of a business process is essential to predict its performance and reliability at design time. Such a prediction activity can effectively support business analysts to early identify performance and reliability problems, in order to compare

different design alternatives and/or to select the best configuration of concrete services that implement the abstract tasks of a given abstract process.

The importance of a performability-based prediction activity is well known and widely recognized in the business process community, yet it is still far to be exploited in practice. This is due to several reasons, the most important one being its expensiveness in terms of costs and time. Traditionally, building a performability analysis model from a business process description requires a non negligible effort and significant skills in performance and reliability theory. Business analysts are usually not familiar with such theory and thus introducing a manual performability prediction activity is not convenient, from both cost and time-to-market points of view.

PyBPMN gives an essential contribution to produce business process specifications that can be mapped to performability analysis models by use of automated model transformations, thus overcoming the drawbacks of traditional model building activities.

In this respect, this chapter also introduces a model-driven method that exploits PyBPMN to predict, at design time, the performability of a business process, either to select the configuration of services that provide the best level of performability or to check if a given configuration satisfies the overall performability requirements of the business process.

The proposed method takes as input the BPMN specification of a business process and makes use of the extended PyBPMN specification to carry out the business process analysis that yields the performability prediction. The method is based on a set of model transformations first to obtain a model that gives the UML (Unified Modeling Language) representation of the business process and then to derive the performance and reliability models whose evaluation yields the performability prediction.

The translation of a BPMN-based description into a UML-based representation is essential to benefit from previous contributions [7][8][10] that provide model-driven methods for automatically building performance and reliability models, as detailed in Sections 0 and 0, respectively. The evaluation of such models yields the performance and reliability predictions that contribute to obtain the performability prediction of a business process.

In addition, the use of UML allows to take advantage of specific UML extensions, called profiles, that provide a standard approach for annotating UML models with domain-specific information. Specifically, the proposed method makes use of both the MARTE profile [27], for annotating performance and reliability information, and the SoaML profile [28], for

annotating information that characterize the SOA-based implementation of a business process.

All such transformations are specified by use of the QVT (Query/View/Transformation) language [29], which is provided by the OMG as the standard language for specifying model transformations that can be executed by a given QVT engine [13].

In order to define an integrated BMPN- and SOA-based approach for specifying, analyzing and implementing business processes, the model-driven method also includes an additional model transformation that is executed to generate the BPEL (Business Process Execution Language) code from the PyBPMN description, according to the algorithm proposed in [21][19].

The main advantage of the proposed model-driven method is that business analysts are allowed to automate the performability analysis of a given business process with no extra effort and without being required to own specific skills of performability theory, as shown by use of an example case study.

The rest of the chapter is organized as follows: Section 2 gives background concepts about BPM, to better frame the context of this chapter contribution. Section 4 describes PyBPMN, i.e., the proposed BPMN extension. Section 5 presents the method that exploits PyBPMN to enable the model-driven performability prediction of a business process, and, finally Section 6 gives an example application.

2. BPM Backgound Concepts

This section outlines a set of reference standards and technologies used in the BPM context, to better frame this chapter scope and contribution.

In particular, the next subsections illustrate the role of SOA-based technologies in the BPM context, the standard BPMN for the specification of business processes and the standard BPEL for the execution of business processes onto SOA-based execution platforms, respectively.

2.1 BPM and Service Oriented Architectures (SOAs)

In order to achieve the highest degree of agility and efficiency, BPM is supported by IT-based standards and technologies, such as Service Oriented Architectures (SOAs) [34] and Web Services [1]. The automated execution of

tasks within a business process, from the implementation perspective, is often founded on the Web Services technology. Web services represent just a set of technologies needed to invoke remote services, while SOA standards "enables the independent construction of services which can be combined to realize meaningful, higher level business process within the context of the enterprise", as stated in [37]. In other words a business process can be executed over an application platform defined in terms of a SOA by use of the Web Service implementation technology, as depicted in Figure 1, which separates the business process specification (upper part) from the implementation and execution onto the underlying application platform (bottom part).

The business process consists of a set of abstract tasks that are mapped (see dashed lines in Figure 1) to a set of services, denoted as *business services*, which provide the actual task operation. In a SOA context such business services are implemented as web services executed on top of a platform that provides the set of required operational resources.

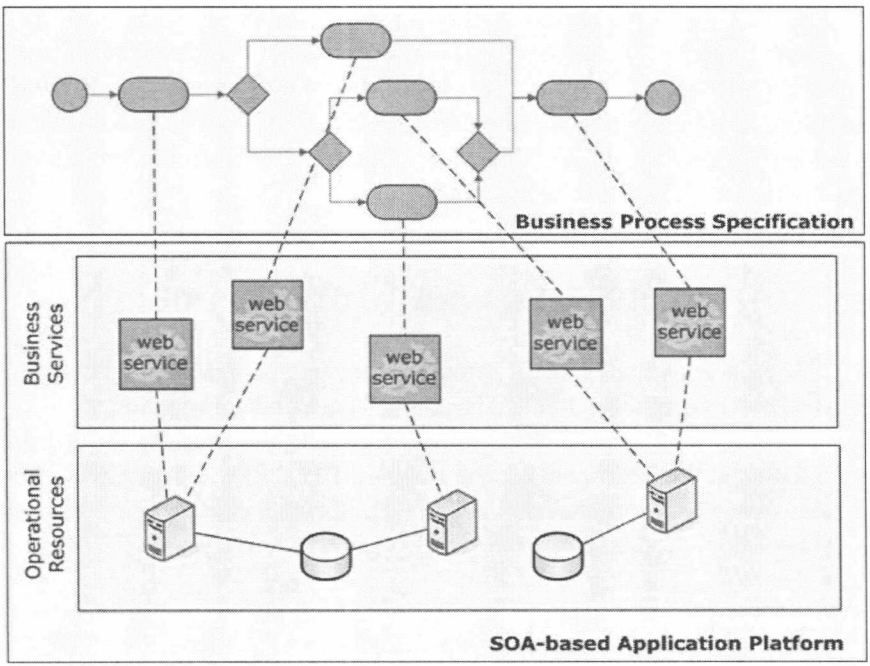


Figure 1. Business process specification and execution [37].