

# Chapter 16

## Tool Based Integration of Requirements Modeling and Validation into Business Process Modeling

**Sven Feja**

*Christian-Albrechts-University of Kiel, Germany*

**Sören Witt**

*Christian-Albrechts-University of Kiel, Germany*

**Andreas Speck**

*Christian-Albrechts-University of Kiel, Germany*

### **ABSTRACT**

*Business process models (BPM) are widely used for specification of software systems, as the basis for model driven software development. Hence, it is crucial to ensure that these BPMs fulfill the requirements they have to comply with. These requirements may originate from various domains. Many may be considered non-functional requirements. They are affecting privacy, security, as well as compliance or economic aspects. In order to avoid error-prone manual checking, automated checking techniques should be applied wherever possible. This requires expressing requirements in a formal manner. The common textual representations for such formal requirements are not well accepted in the modeling domain, since they are settled on a lower level of abstraction, compared to BPMs. In this chapter, the authors present the Business Application Modeler (BAM), which integrates formal requirement specification and automated checking with process modeling. On the one hand BAM supports different notations for process modeling. On the other hand a graphical notation, called G-CTL, for the formal specification of requirements is provided. G-CTL is based on temporal logic, and statements are expressed on the level of abstraction of the graphical process models. Furthermore BAM provides the ability to define selective views on process models. This allows complex domain specific annotations of processes as well as the assignment of responsibilities regarding functional domains. Moreover, BAM integrates into common requirements engineering processes.*

DOI: 10.4018/978-1-4666-4301-7.ch016

## INTRODUCTION

Business process models (BPMs) are well known and accepted in the domain of business application development due to their easy-to-use graphical notation. They are meaningful and comprehensible for software engineers as well as for those, who are familiar with the intended function of the software and not with its development (e.g. management or decision-makers). The purpose of BPMs ranges from documentation, over requirement specification for systems to an early stage of model driven software development (MDSO).

In any case, trust in the validity of BPMs is crucial. Our notion of valid BPMs comprises the level of correct syntax and the level of correct semantics as well, as the more abstract level of business interest. In this contribution we focus on the latter, since syntax and basic semantics are well elaborated for most notations of BPMs. The mentioned level of business interest is related to many aspects. Examples are the wide area of compliance, security, privacy or business rules. Among others, these concerns are essential for business process orchestration, collaboration between enterprises, e-commerce or e-government. If processes fulfill according requirements, it increases the confidence of all participants into the developed systems and processes.

In order to ensure that processes fulfill specific requirement, these need to be specified first. Various kinds of requirements are often expressed as natural language statements. The main advantage of natural language specification is that it is easily comprehensible by every involved stakeholder. On the downside natural language requirements are very ambiguous and the semantic is not clearly defined. Therefore it is not possible to process them automatically. However, the manual checking of the processes is error-prone, expensive and not reliable (Wang, Hidvegi, Baily & Whinston, 2000). Thus, the checking of processes should be performed automatically wherever possible.

Hence, an automatically processable specification of requirements is needed. This can be achieved by the use of formally specified requirements. Next to the obvious advantage of formal specification languages they are often neglected by many groups of the stakeholders. Since formally specified requirements seem incomprehensible or even frightening to them, O'Regan (2006) calls this behavior a cultural shock. Therefore, requirements should be specified in a way that they are comprehensible by every or at least most the involved stakeholders.

Moreover, the formal expression of requirements is presumed by automated checking. This is a textual representation in contrast to the graphical representation of BPMs. Hence, there is a gap between the requirements and the more abstract, graphical representation of process models.

In this paper we present the *Business Application Modeler* (BAM), which is implemented as plug-in for the Eclipse IDE. Basically, BAM is a modeling tool for BPMs. In order to be highly flexible the modeling capabilities of BAM are based on a generic meta meta model which allows BAM to support various modeling notations. In this contribution the presented example processes will only be modeled in the *Event-driven Process Chain* (EPC) notation.

Based on the process models, BAM allows the formal specification of requirements in a graphical representation—the *graphical validation rules*. A major reason for the graphical approach, we propose for expressing the rules is to raise the acceptance in the business modeling community. As denoted above, BPMs aim to be comprehensible for management and decision-makers as well, as for the engineers who implement systems. This should apply for the rules, which express the requirements, as well.

In BAM, a requirement is expressed by a single rule or a set of rules, to which the processes have to comply. As BAM is a modeling tool for BPMs the expressible requirements reflect specifiable issues of BPMs.

23 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/tool-based-integration-requirements-modeling/77711](http://www.igi-global.com/chapter/tool-based-integration-requirements-modeling/77711)

## Related Content

---

### A Survey, Design and Analysis of IoT Security and QoS Challenges

M Kesavanand J Prabhu (2018). *International Journal of Information System Modeling and Design* (pp. 48-66).

[www.irma-international.org/article/a-survey-design-and-analysis-of-iot-security-and-qos-challenges/218171](http://www.irma-international.org/article/a-survey-design-and-analysis-of-iot-security-and-qos-challenges/218171)

### An Operational Semantics of Real-Time Process Algebra (RTPA)

Yingxu Wang and Cyprian F. Ngolah (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 3340-3360).

[www.irma-international.org/chapter/operational-semantics-real-time-process/29565](http://www.irma-international.org/chapter/operational-semantics-real-time-process/29565)

### Could the Work System Method Embrace Systems Concepts More Fully?

Steven Alter (2010). *Emerging Systems Approaches in Information Technologies: Concepts, Theories, and Applications* (pp. 214-226).

[www.irma-international.org/chapter/could-work-system-method-embrace/38182](http://www.irma-international.org/chapter/could-work-system-method-embrace/38182)

### Mining Users' Intents from Logs

Ghazaleh Khodabandelou, Charlotte Hug and Camille Salinesi (2015). *International Journal of Information System Modeling and Design* (pp. 43-71).

[www.irma-international.org/article/mining-users-intents-from-logs/126306](http://www.irma-international.org/article/mining-users-intents-from-logs/126306)

### Implementing Background Net with Knowware System for Personalized Keyword Support

Yuan Chen, Liya Ding, Sio-Long Lo and Dickson K.W. Chiu (2013). *Mobile and Web Innovations in Systems and Service-Oriented Engineering* (pp. 120-133).

[www.irma-international.org/chapter/implementing-background-net-knowware-system/71994](http://www.irma-international.org/chapter/implementing-background-net-knowware-system/71994)