Supporting Customizable Business Process Models Using Graph Transformation Rules

Verena Geist, Software Competence Center Hagenberg, Hagenberg, Austria
Christa Illibauer, Software Competence Center Hagenberg, Hagenberg, Austria
Christine Natschläger, Software Competence Center Hagenberg, Hagenberg, Austria
Robert Hutter, Prologics IT GmbH, Linz, Austria

ABSTRACT

Business Process customization is an active research area in the process management field, dealing with variations/commonalities among processes of a given process family and runtime adaptations of single process instances. Many theoretical approaches have been suggested in the last years; however, practical implementations are rare and limited in their functionality. In this article, a new approach is proposed for capturing customizable process models based on well-known graph transformation techniques and with focus on practical aspects like definition of variation points, linking and propagation of changes, visual highlighting of differences in process variants, and dynamically selecting a specific variant at runtime. The suggested concepts are discussed within case studies, comprising different graph transformation systems for generating process variants supporting (a) variability by restriction, (b) variability by restriction and by extension, and (c) runtime adaptations due to the executing actor. The overall approach is being implemented in the FireStart BPM suite.

KEYWORDS

Actor-Based Adaptations, Business Process Management, Flexibility by Underspecification, Graph Transformation, Process Customization, Runtime Adaptations, Variability Modelling, Variant Management

INTRODUCTION

Process variability is related to flexible business process management (BPM), one of the most active research areas (Reichert & Weber, 2012). Whereas process variability is concerned with design- and customization-time decisions and deals with differences and commonalities among processes of a given process family, process flexibility is concerned with runtime decisions and affects only one particular process instance (La Rosa, van der Aalst, Dumas, & Milani, 2013). Customizable process models include both process variability and flexibility. Typical fields of application are, e.g., production processes, invoice processes, delivery processes, and dynamic processes that have to be adapted, e.g. to a certain actor, at runtime.

There is a large number of theoretical approaches to process variability, focusing on variability by restriction and/or by extension, and flexible business processes. However, many experienced researchers assess the tool support for analysing and constructing customizable process models as limited and not adequate (Reichert & Weber, 2012; La Rosa et al., 2013). In addition, exponential growth makes modelling and maintaining process variants a complex endeavour in practice. Thus,
there is need for establishing a sound method for explicitly supporting customization of business processes in a way that fosters industrial applicability.

In this article, a new approach for capturing customizable business process models based on graph transformation techniques is proposed. The approach is based on the concept for variability modelling in (Natschläger, Geist, Illibauer, & Hutter, 2016) and extended by runtime adaptations to process instances, addressing flexibility by underspecification. Important aspects of the variability concept are the individual definition of adaptable and blocked elements as well as linking and propagation of changes. In addition, completing an incomplete specification, as it may be done for activities, which depend on the executing actors and thus must be adopted (i.e. pre-specified and linked) at runtime, is regarded as a further form of customization. A further advantage is the application of graph transformation rules to deal with the exponentially increasing number of variants. Instead of manually defining all variants, a few rules specify concrete variations and the corresponding variants are automatically generated.

The article is structured as follows. In the following section, an overview of related work on graph transformation, business process variability and flexibility modelling, and tool support for capturing customizable business process models is provided. In the main parts of this work, the authors first present the general approach for modelling business process variants based on key requirements and actor-based adaptations to business process instances and then discuss the proposed concepts within case studies using graph transformation techniques. The paper concludes with a summary on the findings and a discussion on future work.

STATE-OF-THE-ART

This section provides an overview of the state-of-the-art concerning graph transformation, process variability and flexibility (in particular, flexibility by underspecification due to actor-based adaptations) modelling, and the current support of business process customization in BPM suites.

Graph Transformation

Research on graph transformations started around the 1970s; the main idea of graph transformation is the rule-based modification of a graph \( G \), where each application of a rule leads to a graph transformation step. So the transformation of a graph \( L \) (left-hand side) to a graph \( R \) (right-hand side) is based on a rule \( r \) (also called production). Applying the rule \( r = (L, R) \) means finding a match of \( L \) in the source graph \( G \) and replacing \( L \) by \( R \), leading to the target graph of the graph transformation. The application of rules is restricted by application conditions (AC), which can either be positive or negative. A negative application condition (NAC) is satisfied if it is not part of a match of \( L \) in \( G \), whereas a positive application condition (PAC) is satisfied if it is part of the match of \( L \) in \( G \) (Ehrig, Ehrig, Prange, & Taentzer, 2006).

Different approaches for graph transformation have been proposed like node label replacement, hyperedge replacement, or the algebraic approach (Ehrig et al., 2006, p. 10). The algebraic approach, which was initiated by Ehrg, Pfender, and Schneider in 1973, is applied for the suggested concepts. Algebraic graph transformation is supported by a tool for attributed graph grammar systems (called AGG). AGG provides a graphical editor and can be used for specifying graph grammars with a start graph or for typed attributed graph transformations. In addition, AGG offers analysis techniques as for consistency checking, critical pair analysis, and termination evaluation (Ehrig et al., 2006). In previous work, AGG was applied to implement an algebraic graph transformation from standard BPMN to Deontic BPMN diagrams and to prove that this is a trusted model transformation (Natschläger, Kossak, & Schewe, 2015). In this work, AGG is used to implement and evaluate the suggested customization concepts.
A Critical Systems View of Power-Ethics Interactions in Information Systems Evaluation
www.igi-global.com/chapter/critical-systems-view-power-ethics/38184?camid=4v1a

Exploring the Use of Social Media to Advance K12 Science Education
www.igi-global.com/article/exploring-the-use-of-social-media-to-advance-k12-science-education/177885?camid=4v1a