

Chapter 5

Improving the Quality and Cost-Effectiveness of Process-Oriented, Service-Driven Applications: Techniques for Enriching Business Process Models

Thomas Bauer

Neu-Ulm University of Applied Sciences, Germany

Stephan Buchwald

T-Systems International GmbH, Germany

Manfred Reichert

University of Ulm, Germany

ABSTRACT

A key objective of any Service-driven architectural approach is to improve the alignment between business and information technology (IT). Business process management, service composition, and service orchestration, play major roles in achieving this goal. In particular, they allow for the process-aware integration of business actors, business data, and business services. To optimize business-IT alignment and to achieve high business value, the business processes implemented in process-aware information systems (PAISs) must be defined by domain experts, and not by members of the IT department. In current practice, however, the information relevant for process execution is usually not captured at the required level of detail in business process models. In turn, this requires costly interactions between IT departments and domain experts during process implementation. To improve this situation, required execution information should be captured at a sufficient level of detail during business process design (front-loading). As another drawback, existing methods and tools for business process design do not consider available Service-oriented Architecture (SOA) artifacts such as technical service descriptions

DOI: 10.4018/978-1-4666-4193-8.ch005

during process design (*look-ahead*). Both *front-loading* and *look-ahead* are not adequately supported by existing business process modeling tools. In particular, for many process aspects, appropriate techniques for specifying them at a sufficient level of detail during business process design are missing. This chapter presents techniques for enabling *front-loading* and *look-ahead* for selected process aspects and investigates how executable process models can be derived from business process models when enriched with additional information.

INTRODUCTION

Business process management (BPM) and process-aware information systems (PAISs) have become integral parts of enterprise computing and are used to support business processes at the operational level (Mutschler, Reichert, & Bumiller, 2008; Reichert & Weber, 2012; Weske, 2007). As opposed to function- and data-centric information systems, process logic is strictly separated from application code, relying on executable process models that provide the explicit schemes for process execution (Weber, Rinderle, & Reichert, 2007; Weber, Reichert, & Rinderle-Ma, 2008). This enables a *separation of concerns*, which is a well-established principle in computer science to increase maintainability and to reduce costs of change.

A particular challenge for process-aware information systems (PAIS) is *enterprise application integration*, i.e., to link the atomic activities of an executable process model with business functions implemented in heterogeneous, distributed application systems.

In this context, the emergence of Service-oriented architectures (SOA) and well-defined service principles (Erl, 2005) foster *process-centric* application integration in the large scale (Weber & Reichert, 2012). A Service-driven approach provides tools for encapsulating heterogeneous application functions as services with standardized interfaces. These services then can be composed and orchestrated in a process-aware manner based on a run-time component called *process engine*. Altogether, SOA enables enterprise application integration at a high level of abstraction, reducing

the need for realizing application-to-application bridges prevalent in current practice (e.g. based on message exchange or remote procedure calls). In particular, any process activity may retrieve data from an application system (e.g. by invoking a corresponding service) and temporarily store it within the process engine. In turn, this data can be consumed, changed, or complemented when executing subsequent process activities (e.g. human tasks).

Another fundamental goal of a Service-driven approach is to improve *business-IT alignment* (Chen, 2008). In particular, a PAIS should meet the needs of the stakeholders involved in the business processes it implements and not reflect only the design decisions made by the IT department. A variety of issues related to business-IT alignment have been investigated in the ENPROSO (*Enhanced Process Management through Service Orientation*) project (Buchwald, Bauer, & Reichert, 2012; Buchwald, 2012). In particular, ENPROSO revealed that all aspects relevant for process automation should be defined at a sufficient level of detail during the design of business process models. Furthermore, existing artifacts should be reused in this context if available. With *front-loading* and *look-ahead*, this chapter presents two fundamental techniques for achieving these goals.

Front-loading enables the capturing of relevant aspects for process automation at a sufficient level of detail during business process modeling. For a *business process model*, it is not sufficient to only specify the activities, the rough control flow between these activities, and the abstract data objects or roles associated with them. Additionally,

29 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/improving-quality-cost-effectiveness-process/77947

Related Content

Pre-Vaccination and Quarantine Approach for Defense Against Worms Propagation of Malicious Objects in Wireless Sensor Networks

Rudra Pratap Ojha, Pramod Kumar Srivastava and Goutam Sanyal (2018). *International Journal of Information System Modeling and Design* (pp. 1-20).

www.irma-international.org/article/pre-vaccination-and-quarantine-approach-for-defense-against-worms-propagation-of-malicious-objects-in-wireless-sensor-networks/208637

Towards Knowledge Management to Support Decision Making for Software Process Development

Edrisi Muñoz and Elisabeth Capón-García (2014). *Agile Estimation Techniques and Innovative Approaches to Software Process Improvement* (pp. 161-179).

www.irma-international.org/chapter/towards-knowledge-management-to-support-decision-making-for-software-process-development/100277

Iterative Knowledge Based Embedded Systems Development Framework

Goh Kiah Mok, Benny Tjahjono and Ding Wei (2011). *Reconfigurable Embedded Control Systems: Applications for Flexibility and Agility* (pp. 542-566).

www.irma-international.org/chapter/iterative-knowledge-based-embedded-systems/50442

Analysis of ANSI RBAC Support in EJB

Wesam Darwish and Konstantin Beznosov (2011). *International Journal of Secure Software Engineering* (pp. 25-52).

www.irma-international.org/article/analysis-ansi-rbac-support-ejb/55268

CbSSDF: A Two-Layer Conceptual Graph Approach to Web Services Description and Composition – A Scenario Based Solution Analysis and Comparison with OWL-S

Xiaofeng Du, Malcolm Munro and William Song (2013). *Frameworks for Developing Efficient Information Systems: Models, Theory, and Practice* (pp. 123-145).

www.irma-international.org/chapter/cbssdf-two-layer-conceptual-graph/76621