Integrated Modeling of Distributed Software Systems and Workflow Applications

Mathias Weske, Guido Wirtz
Westfälische Wilhelms-Universität Münster, Germany
weske@helios.uni-muenster.de, guidow@math.uni-muenster.de

The design of distributed software systems on the one hand and of workflow management applications on the other hand share a variety of common issues, for instance the modeling of activities, control flow, data flow, and resources. Despite this fact, there appears to be little interaction between the research communities involved. This Minitrack tries to remedy this situation by providing a forum for the discussion of novel approaches to model workflow-aware distributed software systems as well as the design and introduction of workflow management applications, which make use of techniques and tools used in distributed software systems design.

The recent growth in deployment of complex distributed applications has created the need for integrating distributed systems requirements into design techniques. Explicit or potential concurrency, the need for synchronization, resource usage, allocation, and distribution have to be modeled adequately in order to meet these requirements. On the other hand, workflow technology and real-life workflow applications are highly relevant for the software engineering community, since workflow management has created languages, techniques, and tools to model, simulate, monitor, and control the execution of business processes in complex technical and organizational environments.

To achieve the abovementioned goals, the Minitrack program consists of six papers, which are organized in two sessions entitled “Distributed Application Frameworks” and “Advanced Workflow Modeling”. The former addresses issues in the areas of distributed production management systems, system integration and workflow management application introduction; the latter details the modeling and analysis of ad-hoc and adaptive workflows.

In the first session, Bastos and Ruiz propose an object-oriented approach to model dynamic resource allocation in production systems, aiming at improving the allocation of different kinds of resources to enterprise activities. Their approach uses the CIMOSA industry standard for production systems and the Workflow Reference Model proposed by the Workflow Management Coalition and combines them in a new framework. Kotov focuses on the integration aspect of distributed workflow applications. In particular, the design and architecture of a system factory to facilitate the integration of multiple heterogeneous systems is proposed. Göbel and Schwarzer report on the introduction of workflow technology in a multi-national insurance company. The chances and opportunities, as well as the obstacles and risks of this technology are investigated. The report of the conceptual issues and practical experiences is guided by different phases, among which are business process re-engineering, workflow product evaluation, and the introduction of workflow management applications.

In the second session, Huth, Erdmann, and Nastansky address the relationship between ad-hoc workflows and structured workflows, the latter of which are predefined, while the former are not. In particular, the authors show that ad-hoc workflows can be regarded as representations of process knowledge of workflow participants. Under certain conditions, this knowledge can be used to create structured workflows, which afterwards can be executed in a routine fashion. Purvis, Purvis, and Lemalu address adaptive workflows based on a translation approach. Petri Nets are used on the conceptual level, and workflows are implemented by translating Petri Nets to Java classes, using an extension of a commercial software tool. In the proposed approach, adaptation of workflows are implemented by changing Petri Nets, creating new Java templates, and migrating objects to these new templates. In a graph-theoretic approach Basu and Blanning propose methods for analyzing workflows, with the aim of improving the effectiveness of workflows. In particular, attributed meta graphs are used to analyze workflows that have tasks with temporal constraints. Using the proposed approach, time-critical workflow execution paths can be identified to assist workflow modelers in designing more effective workflows.

The work presented shows that techniques and lessons learned from both areas are of mutual benefit, but serious problems have been identified, too. Among them are the consistent incorporation of all aspects relevant to workflow and practically usable methods to combine structure and behavior modeling for distributed software systems.