

Lecture Notes in Business Information Processing

40

Series Editors

Wil van der Aalst

Eindhoven Technical University, The Netherlands

John Mylopoulos

University of Trento, Italy

Norman M. Sadeh

Carnegie Mellon University, Pittsburgh, PA, USA

Michael J. Shaw

University of Illinois, Urbana-Champaign, IL, USA

Clemens Szyperski

Microsoft Research, Redmond, WA, USA

Ingo M. Weber

Semantic Methods for Execution-level Business Modeling

Modeling Support
Through Process Verification
and Service Composition



Springer

Author

Ingo M. Weber

University of New South Wales

School of Computer Science and Engineering

Sydney, NSW 2052, Australia

E-mail: ingo.weber@cse.unsw.edu.au; imweber@imweber.de

Dissertation, angenommen von der Fakultät für Wirtschaftswissenschaften
der Universität Karlsruhe (TH).

Tag der mündlichen Prüfung: 8.6.2009

Referent: Prof. Dr. Rudi Studer

Korreferentin: Prof. Dr. Birgitta König-Ries

Library of Congress Control Number: 2009938724

ACM Computing Classification (1998): H.3.5, J.1, D.2.12, I.2.4

ISSN 1865-1348

ISBN-10 3-642-05084-0 Springer Berlin Heidelberg New York

ISBN-13 978-3-642-05084-8 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

springer.com

© Springer-Verlag Berlin Heidelberg 2009

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12780020 06/3180 5 4 3 2 1 0

Foreword

Enterprises today act in an environment of continuously accelerating change. Since nearly all of an enterprise's activities are supported through IT, the adaptation of IT applications is necessary in a similarly quick fashion. Hereby Service-Oriented Architectures (SOA) and business processes play an important role, which thus have to be either re-designed or adapted accordingly.

This is where the work of Ingo Weber adds to the field, by creating new approaches for the development and adaptation of business processes, focused on an implementation-oriented point of view. Key questions are raised by the verification of specific properties of the processes as well as by the automatic composition of processes out of pre-defined components. The presented work hereby combines formal approaches with real evaluations in an impressive manner.

The work is structured in seven chapters, as follows. The introductory chapter puts forward the motivation for the study, the research questions, and an overview of the work and its central contributions. Subsequently, chap. 2 is concerned with the foundations of the work, where three areas are discussed: business process management, service-oriented computing, and semantic technologies.

Chapter 3 develops a detailed requirement analysis, presenting a total of 19 requirements. Since the breadth of the requirements is too large, the subsequently developed conceptual approach does not cover all of them. The developed approach distinguishes itself primarily by enriching process models with semantic annotations: this way, so-called annotated process models are created, which allow the specification of preconditions and postconditions. For the approach, a component-based architecture is introduced and complemented with a methodology giving targeted instructions on how to systematically use the offered components.

Chapter 4 introduces one of the central technical contributions of the presented work: the verification of annotated process models. Hereby, methods are developed which allow checking of the semantic annotations for consistency – in particular four verification tasks are addressed: (1) “effect conflict”: are there activities which can be executed in parallel and whose postconditions are inconsistent; (2) “precondition conflict”: are there parallel activities such that the postcondition of one is inconsistent with the precondition of the other; (3) “reachability”: are there activities which are never reached in any process execution; (4) “executability”: are there activities whose preconditions are not fulfilled albeit they are scheduled for execution; In a first step toward the solution, so-called annotated process graphs are formally defined in terms of syntax and execution semantics. Building on top of these definitions, the above-mentioned verification tasks are formalized, the computational complexity is analyzed, and algorithms for tractable cases of the above tasks are developed.

In chap. 5, new solutions for the composition of services are developed. Hereby, the typical trade-off between expressivity of the formalisms and the efficiency of the solution algorithms is taken into account. Substantial contributions of the developed approaches are according heuristics which make use of the domain knowledge presented in the ontologies, as well as methods for approximate reasoning. The developed solution approaches are first presented in an overview and then detailed step by step. Among others, the Web Service Composition problem is formalized for this purpose, and complexity results are introduced which depend on the expressivity of the formalisms for modeling domain knowledge.

Chapter 6 addresses the evaluation of the developed approaches. Five different prototypical scenarios are presented and illustrated with screen shots. In addition, empirical evaluations for the composition of services are discussed. The chapter ends with the presentation of evaluations based on several case studies.

The work concludes with a short summary and an outlook in chap. 7. Elaborate appendices include detailed proofs to the theorems formulated throughout the work.

The work presented by Ingo Weber contains original methodical contributions for the modeling and formal analysis of implementation-oriented process models. As such, the work impresses with the combination of strictly formally substantiated solution approaches and evaluations in practical application scenarios.

Preface

The pace of change in the current economical world increases continuously. With respect to information technology (IT), there are currently two main trends with the goal of enabling organizations to adapt to changing circumstances. On the one hand, software applications are being altered such that they offer Web services interfaces. This way, the business logic and the functionality become accessible from the outside in a standardized way. On the other hand, organizations increasingly apply methods from business process modeling. The vision is to create executable process models in very short time frames, which orchestrate the usage of Web services, and thus to enable the swift implementation of changed processes. However, the software tools offered today have difficulties to fulfill the demands that arise from the combination of the two technologies.

This monograph aims at the advancement of scientific knowledge on the implementation of business processes in organizations' IT systems. For this purpose, the monograph investigates if and how modelers of processes at the execution level can be supported using semantic technologies. Execution-level processes abstract from the technical details of Web services. Thus, they remain comprehensible for non-technicians. At the same time they reflect which Web services are available. We first analyze the requirements of process modeling support at this level. On this basis, we construct a conceptual framework supporting the majority of the requirements. For two of the components from the framework there were no sufficient solutions available. Therefore, we design detailed solutions for these: Firstly, we investigate the verification of annotated process models. As early as at design time, this type of verification can unveil conflicts arising from various sources such as an incorrect ordering of activities in a process model – e.g., when two activities are scheduled as parallel, but their parallel execution would lead to undefined results. Secondly, we investigate the problem of automatically composing semantically annotated Web services. This is relevant for finding an activity implementation made up of Web services. For both approaches we investigate trade-offs between expressivity and scalability, with an emphasis on scalability. Several prototypical

implementations enable the evaluation of the above components with respect to process models from practice.

The monograph was originally written as a PhD thesis with the Universität Karlsruhe (TH) while I worked at SAP Research in Karlsruhe, Germany. This environment allowed me to interact with many great researchers from academia, in particular from the AIFB institute, Universität Karlsruhe (TH), the IPE group at the Forschungszentrum Informatik (FZI), and several publicly funded projects, especially the EU-funded Integrated Project *SUPER*. People that were particularly influential in the creation of this monograph are: Anupriya Ankolekar, Sudhir Agarwal, Nenad Stojanovic, Tomasz Kaczmarek, James Scicluna, and Wil van der Aalst.

Special thanks go to my PhD supervisor Prof. Rudi Studer, who gave me the freedom to pursue my visions, yet guided me with insightful criticism and encouragement. I am also very grateful to Prof. Birgitta König-Ries, the second reviewer of my thesis, for her time and highly valuable feedback. Further I want to thank the other two members of my committee for their support, Prof. Andreas Oberweis and Prof. Christof Weinhardt.

My project leader, Christian Brelage, as well as my managers at SAP, Orestis Terzidis and Elmar Dorner, were extremely supportive by giving me substantial freedom in my work and by allowing me to participate in many international conferences and workshops. My research visit to SAP Research in Brisbane in early 2008 was also very fruitful, allowing me to collaborate with Alistair Barros, Marek Kowalkiewicz, Uwe Kylau, and Julien Vayssière, as well as with Shazia Sadiq and Guido Governatori from the University of Queensland, and Jan Mendling, at the time with Queensland University of Technology.

Further influential feedback came from Michael Altenhofen, Daniel Oberle, Roger Kilian-Kehr, Wasim Sadiq, Matthias Born, Ivan Markovic, Christian Drumm, and many more colleagues and students at SAP Research and from SAP product development groups. Jörg Hoffmann deserves special mention; as co-author on many of my publications he showed me how to apply formal methods to applied problems, and worked with me on many of the topics presented in this monograph.

Finally, I would like to thank my family, friends, and my partner, Susanne, for their patience and endurance during this intense time. Without their support, it would not have been possible to complete this work.

Sydney, Australia
August 2009

Ingo M. Weber

Contents

Part I Foundations

1	Introduction	3
1.1	Motivation and Overview	3
1.2	Hypotheses	6
1.3	Methodology	7
1.4	Details of the Solution Approach	9
1.4.1	Verification of Annotated Process Models	9
1.4.2	Composition of Execution-level Artifacts	10
1.5	Structure of This Book	10
2	Background	13
2.1	Business Process Management	13
2.1.1	The Path to Business Process Management	14
2.1.2	Definitions	17
2.1.3	Overview and Methodology for BPM	18
2.1.4	Process Formalisms	25
2.1.5	Workflow Patterns	27
2.1.6	Current Challenges in BPM	27
2.2	Service-Oriented Computing	28
2.2.1	Service-Oriented Architecture	28
2.2.2	Web Services	29
2.2.3	Service-Oriented Design Principles	31
2.2.4	An Orchestration Language: WSBPEL	32
2.3	Semantic Technologies	34
2.3.1	Ontologies and Reasoning	36
2.3.2	Frame Logic	37
2.3.3	Description Logics	38
2.3.4	Semantic Web Services	39
2.3.5	Semantic Business Process Management	42
2.4	Chapter Summary	44

Part II Modeling Support through Verification and Composition

3	Requirements Analysis and Conceptual Framework	47
3.1	The Problem of Supporting Composition in BPM.	47
3.2	Requirements for Composition Support in Business Process Modeling.	50
3.2.1	Modeling Extension Requirements	51
3.2.2	Fundamental Requirements	53
3.2.3	Non-functional Requirements	56
3.2.4	Situational Requirements	59
3.3	Requirements Discussion	62
3.4	Conceptual Framework Overview	62
3.4.1	Semantics of Annotated Process Models	63
3.4.2	Annotating Preconditions and Effects	65
3.4.3	Refined Problem Description	65
3.5	High-Level Architecture and Components	67
3.5.1	Component Overview and Explanation	67
3.5.2	Discovery	69
3.5.3	Task Composition	71
3.5.4	Auto-completion	73
3.5.5	Data Mediation	75
3.5.6	Validation	77
3.6	Methodological Guidance for the Semantic BPM Life-Cycle	79
3.6.1	Overview	80
3.6.2	The Modeling Phase	82
3.6.3	The Configuration Phase	85
3.6.4	Discussion of the Methodology	88
3.7	Automatic Process Composition Algorithm	89
3.8	Related Work	91
3.8.1	Related Work for the Requirements Analysis	92
3.8.2	Related Work for the Conceptual Framework	93
3.9	Summary and Discussion	94
4	Verification of Annotated Process Models	97
4.1	Overview	98
4.2	Annotated Process Graphs	103
4.2.1	Process Graphs	103
4.2.2	Semantic Annotations.	108
4.3	Verification Tasks	115
4.4	Checking Precondition and Effect Conflicts	117
4.5	Computational Hardness of Executability and Reachability Checking	124
4.6	Polynomial-Time Executability Checking for Basic Processes	130
4.7	Resolving Inconsistencies	138
4.7.1	Executability	139
4.7.2	Precondition and Effect Conflicts	140

4.8	Related Work	141
4.8.1	Petri Nets	142
4.8.2	Model Checking	144
4.8.3	Beyond Control-Flow	144
4.9	Chapter Conclusion and Discussion of Open Questions	146
5	Task Composition	149
5.1	Overview	150
5.2	Design and Main Algorithms	154
5.2.1	Overall Arrangement	155
5.2.2	Forward Search	158
5.2.3	A Non-deterministic Search Algorithm	161
5.3	Formalizing Web Service Composition	165
5.3.1	<i>WSC</i> Syntax	165
5.3.2	<i>WSC</i> Semantics	166
5.4	Update Reasoning	168
5.4.1	Basic Aspects of the Search	169
5.4.2	General and Horn Background Theory	169
5.4.3	Binary Background Theory	171
5.4.4	Approximate Update Reasoning	172
5.5	Heuristic Function	174
5.5.1	Main Approach of the Heuristic Function	175
5.5.2	Building an Approximated Composition Graph	176
5.5.3	Extracting an Approximated Solution	181
5.6	Peripherals	184
5.6.1	Information Gathering	184
5.6.2	Discovery	188
5.7	Related Work	192
5.7.1	AI Planning	193
5.7.2	Update Problems	194
5.7.3	Web Service Composition	197
5.8	Chapter Conclusion	200

Part III Finale

6	Evaluation	203
6.1	Evaluation Overview	203
6.2	Prototypes	207
6.2.1	Back-End Components	207
6.2.2	Integrated Prototypes	208
6.3	Empirical Results for Task Composition	213
6.3.1	Performance with Deterministic Simulated Services	214
6.3.2	Performance with Non-deterministic Real-World Services	217
6.4	Process Case Studies	218

6.4.1	General Observations	218
6.4.2	Overview of the Process Case Studies	219
6.4.3	Findings	221
6.5	Chapter Summary	224
7	Conclusions and Outlook	227
7.1	Summary	227
7.2	Open Problems	229
7.3	Future Research Topics	230
7.4	Reflection	230
Appendix:		
A	Proofs	233
A.1	Proofs for the Verification Technique	233
A.1.1	Correctness of M-Propagation	233
A.1.2	Binary Theories Can Be Compiled Away	240
A.1.3	Correctness of I-Propagation	242
A.1.4	I-Propagation Can Be Used for Executability Checking ..	249
A.1.5	Complexity Results	251
A.2	Proofs for Task Composition	260
References		271
Index		291