Lecture Notes in Computer Science

The LNCS series reports state-of-the-art results in computer science research, development, and education, at a high level and in both printed and electronic form. Enjoying tight cooperation with the R&D community, with numerous individuals, as well as with prestigious organizations and societies, LNCS has grown into the most comprehensive computer science research forum available.

The scope of LNCS, including its subseries LNAI and LNBI, spans the whole range of computer science and information technology including interdisciplinary topics in a variety of application fields. The type of material published traditionally includes

- proceedings (published in time for the respective conference)
- post-proceedings (consisting of thoroughly revised final full papers)
- research monographs (which may be based on outstanding PhD work, research projects, technical reports, etc.)

More recently, several color-cover sublines have been added featuring, beyond a collection of papers, various added-value components; these sublines include

- tutorials (textbook-like monographs or collections of lectures given at advanced courses)
- state-of-the-art surveys (offering complete and mediated coverage of a topic)
- hot topics (introducing emergent topics to the broader community)

In parallel to the printed book, each new volume is published electronically in LNCS Online.

Detailed information on LNCS can be found at
www.springer.com/lncs

Proposals for publication should be sent to
LNCS Editorial, Tiergartenstr. 17, 69121 Heidelberg, Germany
E-mail: lncs@springer.com

ISSN 0302-9743

ISBN 978-3-642-21639-8

Lecture Notes in Computer Science
LNCS LNAI LNBI

springer.com

Mouratidis • Rolland (Eds.)

LNCS 6741

Advanced Information Systems Engineering

CAiSE'11

CAiSE 2011

Haralambos Mouratidis
Colette Rolland (Eds.)

Advanced Information Systems Engineering

**23rd International Conference, CAiSE 2011
London, UK, June 2011
Proceedings**

CAiSE'11

Springer

Lecture Notes in Computer Science 6741

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Haralambos Mouratidis Colette Rolland (Eds.)

Advanced Information Systems Engineering

23rd International Conference, CAiSE 2011
London, UK, June 20-24, 2011
Proceedings

Springer

Volume Editors

Haralambos Mouratidis
University of East London
School of Computing, IT and Engineering
Docklands Campus, 4/6 University Way, E16 2RD London, UK
E-mail: H.Mouratidis@uel.ac.uk

Colette Rolland
Université Paris1 Panthéon Sorbonne
CRI
90 Rue de Tolbiac, 75013 Paris, France
E-mail: rolland@univ-paris1.fr

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-21639-8 e-ISBN 978-3-642-21640-4
DOI 10.1007/978-3-642-21640-4
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011928907

CR Subject Classification (1998): H.4, H.3, D.2, C.2, J.1, I.2

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

A warm welcome to the proceedings of the 23rd International Conference on Advanced Information Systems Engineering (CAiSE 2011)! The CAiSE series of conferences started in 1989 with the objective to provide a forum for the exchange of experience, research results, ideas and prototypes in the field of information systems engineering. Twenty-two years later, CAiSE has established itself as a leading venue in the information systems area for presenting and exchanging results of emerging methods and technologies that facilitate innovation and create business opportunities.

CAiSE 2011, held in London during June 20–24, 2011 continued this tradition. The theme of CAiSE 2011 was "Information Systems Olympics: Information Systems in a Diverse World." This year's CAiSE conference theme was linked to the coming London Olympic and Paralympic Games 2012, two international multi-sport events that bring together athletes from all continents to celebrate sporting excellence but also human diversity. Diversity is an important concept for modern information systems. Information systems are diverse by nature ranging from basic systems to complex ones and from small to large. The process of constructing such systems is also diverse ranging from ad-hoc methods to structured and formal methods. Diversity is also present among information systems developers, from novice to experienced. Moreover, the wide acceptance of information systems and their usage in almost every aspect of human life has also introduced diversity among users. Users are both novice and experienced and they demonstrate differences related to race, ethnicity, gender, socio-economic status, age, physical abilities, religious beliefs, and so on. It is therefore the responsibility of the information systems engineering community to engineer information systems that operate in such a diverse world.

CAiSE 2011 received 320 submissions, the largest number ever received in the CAiSE conference series. Most of the submissions came from Germany, Spain, Italy, France and China. Following an extensive review process, which included a Program Committee/Program Board meeting during February 13–14, 2011 in London, 42 submissions were accepted as full papers and 5 as short papers. Accepted papers addressed a large variety of issues related to the conference and were organized into ten themes: Requirements, Adaptation and Evolution, Model Transformation, Conceptual Design, Domain-Specific Languages, Case Studies and Experiences, Mining and Matching, Service and Management, Validation and Quality, Business Process Modeling. The program of the conference was also supplemented by a number of tutorials, 11 workshops, a Doctoral Consortium, the CAiSE Forum, and two working conferences. Two keynote speeches were delivered as part of the conference program. Anthony Finkelstein talked about "Open Challenges at the Boundaries Software Engineering and Information Systems," while Dimitrios Beis talked about "Information Systems for the

Olympics Games." Moreover, a panel discussed issues related to "Green and Sustainable Information Systems."

The organization and successful running of a large conference such as CAiSE would not be possible without the valuable help and time of a large number of people. As editors of this volume, we would like to express our gratitude to the Program Committee members, additional reviewers and the Program Board members for their valuable support in selecting the papers for the scientific program of the conference; to the authors of the papers for sending their work to CAiSE; to the presenters of the papers; and to the participants of the conference for their contribution. We also thank our sponsors and the General Chair and Chairs of the various CAiSE 2011 committees for their assistance in creating an exciting scientific program. We would also like to thank the local Organizing Committee at the University of East London for their hospitality and the organization of the social events of the conference.

March 2011

Colette Rolland
Haralambos Mouratidis

Organization

Advisory Committee

Arne Sølvberg	Norwegian University of Science and Technology, Norway
Janis Bubenko Jr.	Royal Institute of Technology, Sweden
Colette Rolland	Université Paris 1 Panthéon Sorbonne, France

General Chair

Pericles Loucopoulos	Loughborough University, UK

Program Chairs

Haralambos Mouratidis	University of East London, UK
Colette Rolland	Université Paris 1 Panthéon Sorbonne, France

Local Arrangements Chairs

Elias Pimenidis	University of East London, UK
Miltos Petridis	University of Greenwich, UK

Workshop and Tutorial Chairs

Oscar Pastor	Valencia University of Technology, Spain
Camille Salinesi	Université Paris 1 Panthéon Sorbonne, France

Forum Chair

Selmin Nurcan	Université Paris 1 Panthéon Sorbonne, France

Panel Chair

Barbara Pernici	Politecnico di Milano, Italy

Doctoral Consortium Chairs

Michel Léonard	Université de Genève, Switzerland
Bernhard Thalheim	Christian Albrechts University Kiel, Germany
Cornelia Boldyreff	University of East London, UK

Publication Chairs

Jolita Ralyté	Université de Genève, Switzerland
David Preston	University of East London, UK

Publicity Chairs

Rebecca Deneckere	Université Paris 1 Panthéon Sorbonne, France
Jaelson Castro	Universidade Federal de Pernambuco, Brazil
Leszek Maciaszek	Macquarie University, Australia
Kecheng Liu	University of Reading, UK
Keng Siau	University of Nebraska-Lincoln, USA

Finance Chair

Mohammad Dastbaz	University of East London, UK

Webmasters

Michalis Pavlidis	University of East London, UK
Sambhu Singh	University of East London, UK

Program Committee Board

Marco Bajec, Slovenia
Nacer Boudjilida, France
Eric Dubois, Luxembourg
Xavier Franch, Spain
Marina Jirotka, UK
Moira Norrie, Switzerland
Barbara Pernici, Italy
Klaus Pohl, Germany
Jolita Ralyté, Switzerland
Camille Salinesi, France
Janis Stirna, Sweden
Roel Wieringa, The Netherlands

Program Committee

Wil van der Aalst, The Netherlands
Peggy Aravantinou, Greece
Pär Ågerfalk, Sweden
Hans Akkermans, The Netherlands
Antonia Albani, The Netherlands
Daniel Amyot, Canada
Paris Avgeriou, The Netherlands
Luciano Baresi, Italy
Ahmad Barfourosh, Iran
Zohra Bellahsene, France
Boalem Benatallah, Australia
Giuseppe Berio, France
Mokrane Bouzeghoub, France
Silvana Castano, Italy
Jaelson Castro, Brazil
Corine Cauvet, France
Donna Champion, UK
Vasilis Chrisikopoulos, Greece
Ioanna Constantiou, Denmark
Panos Constantopoulos, Greece

Valeria De Antonellis, Italy
Joerg Evermann, Canada
João Falcão a Cunha, Portugal
Paolo Falcarin, UK
Mariagrazia Fugini, Italy
Paolo Giorgini, Italy
Stefanos Gritzalis, Greece
Remigijus Gustas, Sweden
Terry Halpin, USA
Willem-Jan van den Heuvel, The Netherlands
Patrick Heymans, Belgium
Jane Huang, USA
Matthias Jarke, Germany
Paul Johannesson, Sweden
Christos Kalloniatis, Greece
Dimitris Karagiannis, Austria
Panagiotis Karras, Singapore
Evangelia Kavakli, Greece
Zoubida Kedad, France
Marite Kirikova, Latvia
Naoufel Kraiem, Tunisia
John Krogstie, Norway
Wilfried Lemahieu, Belgium
Michel Leonard, Switzerland
Panos Louridas, Greece
Kalle Lyytinen, USA
Raimundas Matulevicius, Estonia
Jan Mendling, Germany
Isabelle Mirbel, France
John Mylopoulos, Canada
Selmin Nuncan, France
Andreas Oberweis, Germany
Antoni Olive, Spain
Andreas Opdahl, Norway
Mike Papazoglou, The Netherlands
Jeffrey Parsons, Canada
Oscar Pastor, Spain
Anne Persson, Sweden
Michael Petit, Belgium
Yves Pigneur, Switzerland
Elias Pimenidis, UK
Geert Poels, Belgium
Naveen Prakash, India
Erik Proper, The Netherlands
Sudha Ram, USA
Ruth Raventos, Spain
Manfred Reichert, Germany
Stephan Reiff-Marganiec, UK
Bill Robinson, USA
Michael Rosemann, Australia
Gustavo Rossi, Argentina
Matti Rossi, Finland
Motoshi Saeki, Japan
Christos Schizas, Cyprus
Keng Siau, USA
Monique Snoeck, Belgium
Pnina Soffer, Israel
Carine Souveyet, France
Arnon Sturm, Israel
Kenji Taguchi, Japan
David Taniar, Australia
Ernest Teniente, Spain
Bernhard Thalheim, Germany
Aphrodite Tsalgatidou, Greece
Irene Vanderfeesten, The Netherlands
Olegas Vasilecas, Lithuania
Yannis Vassiliou, Greece
Yair Wand, Canada
Hans Weigand, The Netherlands
Mathias Weske, Germany
Jon Whittle, UK
Carson Woo, Canada
Eric Yu, Canada
Konstantinos Zografos, Greece

Additional Referees

Alberto Abelló
David Aguilera-Moncusi
Saeed Ahmadi-Behnam
Naved Ahmed
Reza Akbarinia
Fernanda Alencar
Raian Ali
Christos Anagnostopoulos
Birger Andersson
Vasilios Andrikopoulos
Ion Androutsopoulos
Luca Ardito
George Athanasopoulos
Ahmed Awad
Daniele Barone
Saeed Ahmadi Behnam
Maria Bergholtz
Maxime Bernaert
Devis Bianchini
Riccardo Bonazzi
Boris Brandherm
Glenn J. Browne
Stephan Buchwald
Andrea Capiluppi
Amit Chopra
Remi Coletta
Ajantha Dahanayake
Fabiano Dalpiaz
Rébecca Deneckère
Olfa Djebbi
Vicky Dritsou
Fabien Duchateau
Rami Eid-Sabbagh
Golnaz Elahi
Amal Elgammal
Thibault Estier
Alfio Ferrara
Kunihiko Fujita
Matthias Galster
Dimitris Gavrilis
Andrew Gemino
Sepideh Ghanavati
Emmanuel Giakoumakis
Bas van Gils
Daniela Grigori
Irit Hadar
Stijn Hoppenbrouwers
Ela Hunt
Shareeful Islam
Lei Jiang
Rim Kaabi
Diana Kalibatiene
Christos Kalloniatis
Maya Kaner
Haki Kazem
Takashi Kitamura
David Knuplesch
Spyros Kokolakis
Jens Kolb
Takafumi Komoto
Panos Kourouthanassis
Eleni Koutrouli
Vera Kuenzle
Ales Kumer
Matthias Kunze
Andreas Lanz
Alexei Lapouchnian
Dejan Lavbic
Evaldas Lebedys
Francesco Lelli
Zhan Liu
Mathias Lohrmann
Linh Thao Ly
Alexander Lübbe
Manolis Maragoudakis
Michele Melchiori
Slim Mesfar
Marco Mesiti
Alexandre Métrailler
Wolfgang Molnar
Geert Monsieur
Stefano Montanelli
Gunter Mussbacher
Wanda Opprecht
Sami Ouali
Michael Pantazoglou

Michael Parkin
Adamantia Pateli
Michalis Pavlidis
Raul Mazo Pena
João Pimentel
Eric Platon
Viara Popova
Alireza Pourshahid
Ruediger Pryss
Ricardo Ramos
Jan Recker
Evangelos Rekleitis
Oscar Romero
Christoph Rosenkranz
Ulysse Rosselet
Khalid Saleem
Camille Salinesi
Emanuel Santos
Sihem Ben Sassi
Ricardo Seguel
Azalia Shamsaei
Omri Shiv
Patricio Silva
David Simms
Jonas Sjöström
Aidas Smaizys
Sergey Smirnov
Sergejus Sosunovas
Kilian Stoffel
Klaas-Jan Stol
Lovro Subelj
Yehia Taher
Ilias P. Tatsiopoulos
B. Veeresh Thummadi
Chouki Tibermacine
Julian Tiedeken
Dan Tofan
Federico Tomassetti
Justas Trinkunas
Christina Tsagkani
Aggeliki Tsohou
Dimitrios Tsoumakos
Oktay Turetken
Gaia Varese
Yannis Velegrakis
Antonio Vetró
Kaja Vidmar
Antonio Villegas
Emmanuel Waller
Matthias Weidlich
Michael Wufka
Iyad Zikra
Slavko Zitnik
Aljaz Zrnec

Table of Contents

Session 3: Model Transformation 1

Session 4: Conceptual Design 1

Session 5: Conceptual Design 2

Session 6: Domain Specific Languages

Session 7: Case Studies and Experiences

Session 8: Model Transformation 2

Session 9: Mining and Matching

Session 10: Business Process Modelling

Session 11: Validation and Quality

Session 12: Service and Management 1

Session 13: Service and Management 2

Session 14

Business Process Service Oriented Methodology (BPSOM) with Service Generation in SoaML

Andrea Delgado[1], Francisco Ruiz[2], Ignacio García-Rodríguez de Guzmán[2], and Mario Piattini[2]

[1] Computer Science Institute, Faculty of Engineering, University of the Republica
Julio Herrera y Reissig 565
CP 11300, Montevideo, Uruguay
[2] Alarcos Research Group, Information Tech & Systems Dep., University of Castilla-La Mancha
Paseo de la Universidad No.4
CP 13071, Ciudad Real, España
adelgado@fing.edu.uy,
{francisco.ruizg,ignacio.grodriguez,mario.piattini}@uclm.es

Abstract. Carrying out business processes by means of software services helps to close the business–systems gap, by introducing an intermediate layer between business process definition and software systems, thus permitting not only better independence, but also more traceability between them. Despite the fact that technologies have matured to support this new reality, there is a lack of methodologies and notations, although some have been proposed to guide service development with different visions of service design and implementation. Service modeling is the basis for, among other things, the automation of several development steps by means of the model-driven development paradigm. The SoaML standard is a major step towards service modeling in UML. In this paper we extend our Business Process Service Oriented Methodology (BPSOM) for service development from business processes by integrating two main aspects: service modeling using SoaML and QVT transformations to obtain SoaML service models from BPMN BP models.

Keywords: Business Process Management (BPSOM), Service Oriented Computing (SOC), Model Driven Development (MDD), BPMN, SoaML.

1 Introduction

The modeling of business process as the means to show explicitly how organizations carry out their business has gained importance in recent years. Although the business area has several mature techniques with which to manage its business processes, based on the Business Process Management (BPM) [1][2] paradigm, the software area has recently been integrating this vision into software development, supported by the Service Oriented Computing (SOC) [3] paradigm. Carrying out business processes by means of software services based on a Service Oriented Architecture (SOA) [4][5] style, helps to close the business-system gap which has come about as a result of the differences between business and software area visions of the organization. The

H. Mouratidis and C. Rolland (Eds.): CAiSE 2011, LNCS 6741, pp. 672–680, 2011.

Model Driven Development (MDD) [6] paradigm, along with Model Driven Architecture (MDA) [7] have an important role to play. They allow correspondences between models to be defined, since they are key development artifacts, permitting the generation of code in different technologies. Although technologies have matured to support this new reality, few methodologies have been proposed to guide the service development process. The Service Oriented Architecture Modeling Language (SoaML) [8], recently defined by OMG, is a major step towards the modeling of services using UML and specific service stereotypes.

The standardized framework MINERVA [9] we have defined aims to support the Business Process (BP) lifecycle [1] by applying service-oriented and model-driven paradigms to business processes; it can be viewed on-line in [10]. The Business Process Service Oriented Methodology (BPSOM) [11] integrated in MINERVA provides the methodological guide with which to develop services from business processes. This paper extends the definition of BPSOM shown in [11] by integrating two new key aspects: the use of the SoaML standard for service modeling, and transformations using the Query/Views/ Transformations (QVT) [12] language, to generate SoaML service models, when possible, from business process models in Business Process Modeling Notation (BPMN) [13].

The remainder of the paper is organized as follows: BPSOM is presented in Section 2, along with the use of BPMN for BP modeling and SoaML for service modeling. In Section 3 service generation from business process is presented, related work is described in Section 4, and conclusions and future work are in Section 5.

2 BPSOM Definition

BPSOM has been defined for integration into the existing software development process used in the organization, with the aim of reusing existing knowledge, by adding only specific elements for service oriented development from business processes. Fig. 1 shows the definition of BPSOM and its use within the base process.

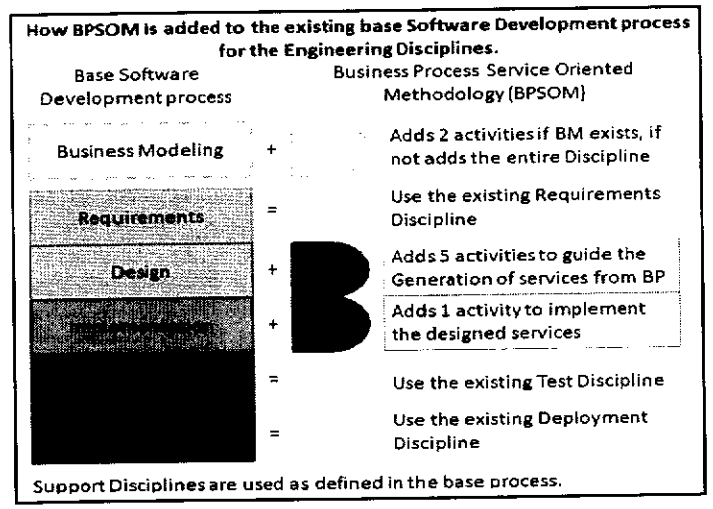

Fig. 1. How BPSOM is added to the existing software development process

We have based our work on the broad idea of methodology as outlined in [14]: a set of methods or techniques related, along with a process model and a set of deliverables, metrics, tools and management guidelines (including roles and organization team work). The definition of BPSOM began in 2005 and its disciplines, activities, roles and artifacts are detailed in [11]. In this paper we focus on the extension of BPSOM by adding service modeling in SoaML, along with the automatic generation of SoaML service models from BPMN BP models through QVT transformations, first described in [15]. The key aspect is to show where and how to use SoaML diagrams in BPSOM, as the standard provides their description, but no guide for using them. Transformations help obtaining some of the SoaML models automatically, providing support to activities, although human intervention is needed.

2.1 BPMN Use in the BPSOM Business Modeling Discipline

There is a great variety of notations for business process modeling [16], although in recent years, BPMN has emerged as the one preferred. Business people can use it to model business processes by themselves and then pass it to the software area.

BM1 – Assess the target organization. This activity aims to involve the project team in the organization for which the development is being carried out. The participating roles in this activity are the Business Analyst (from the business area), the Analyst and the Architect (from the software project team). The OMG Business Motivation Model (BMM) [17] can also be used for modeling goals and information which can be linked to SoaML services.

BM2 – Identify Business Processes. This is one of the key activities in the development of services from BP, since it is the main input needed to understand and describe BP in the organization. We use BPMN to specify them, which provides elements such as swimlanes (pool, lane), flow objects (activities, gateways), connecting objects (sequence, message) and artifacts (group, data) to model BP. Fig. 2 shows the "Patient Admission and Registration for Major Ambulatory Surgery (MAS)" BP from the Ciudad Real General Hospital project on which we are working, adapted to be used as example.

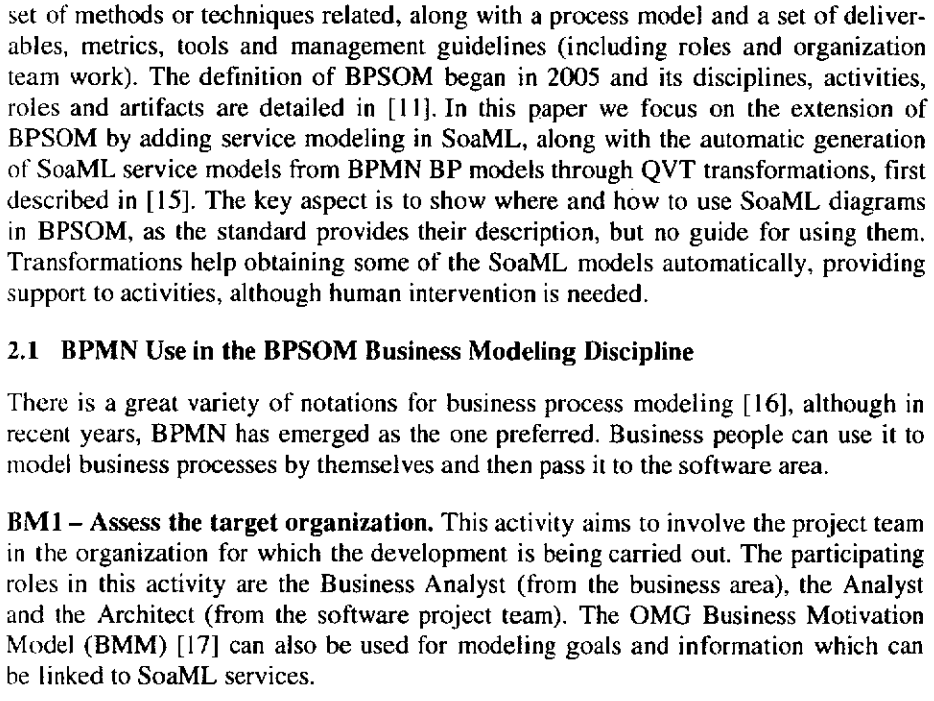

Fig. 2. "Patient Admission and Registration for MAS" Business Process in BPMN

2.2 SoaML Use in BPSOM Design Discipline

There is also a wide choice of notations for service modeling [16], UML being the one preferred. The SoaML profile extends UML by adding specific elements for service modeling, and will therefore soon be adopted by the community. It provides several stereotypes with which to specify services (contract, interfaces, operations, parameters) and the service architecture for the business process.

D1 – Identify and categorize services. This activity aims to identify the services needed to perform the business process under development and it is a key one in our approach. One of the main inputs of this activity is the BP model specified previously. The use of SoaML implies defining the Service Architecture (SA) which specifies the participants, contracts for the services and the roles they play as provider or consumer. Fig. 3 shows the SoaML SA for the example.

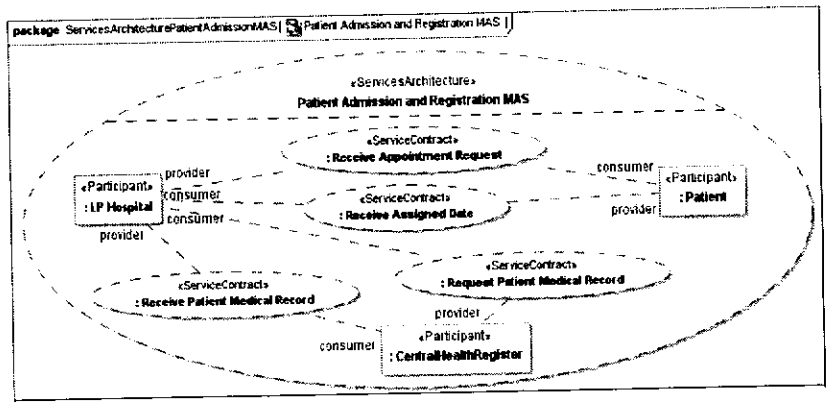

Fig. 3. SoaML ServicesArchitecture diagram for business process in Fig. 2

Services that the organization needs to provide to other parties and services that the organization has to consume from other parties are identified, based on the messages exchanged, each party being defined by a pool. To identify the services to support the business process, we look at each message exchanged between the pools (participants), setting the activity type to "ServiceTask" when we define it as a service. The ones that present incoming messages will be providers and those with outgoing

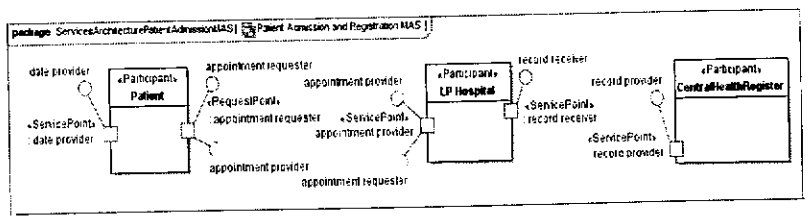

Fig. 4. Service and Request Ports for Participants

messages to a service task will be consumers. Services will be assigned to the participant corresponding to the pool containing each activity, and the associated Service Contract will hold all service information. Fig. 4 displays some Service and Request Ports (formerly Points) for the services defined, showing the bidirectional and unidirectional pattern of communication that can be defined.

D2 – Specify services. The specification of services corresponds to the definition of all the information needed, including the associated Service Contract with interfaces, operations, input and output parameters, among others. The information related to the in and out messages must be specified, indicating the parameters and data to be exchanged between the parties. The choreography defined by the Service Contract must also be specified, based on the interaction between participants. Once all this information has been considered, the most important parts of the ServiceContract can be generated and this can then be completed by the Architect or developers, who will also have to give the implementation details. Fig. 5 presents the ServiceContract definition and its choreography for the "ReceiveAppointmentRequest" service.

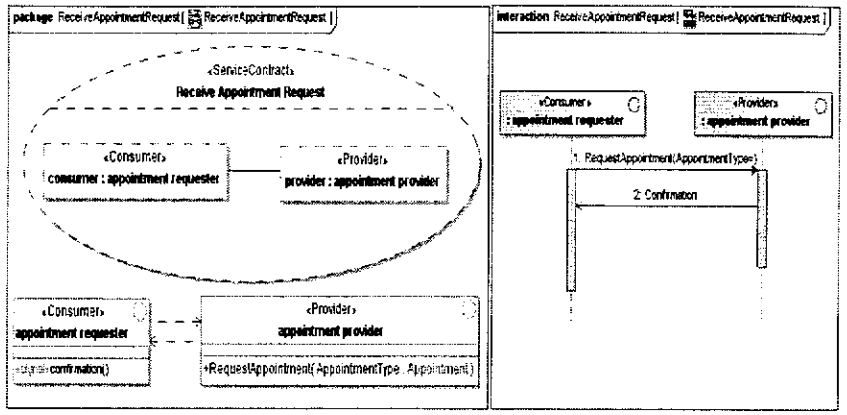

Fig. 5. ServiceContract and Choreography for the "ReceiveAppointmentRequest" service

D3 – Investigate existing services. The principal goal of this activity is to reuse the organization's existing services, as far as possible. To do so, a central Service Catalogue is defined, which has to be searched in each service development project. In the SoaML component diagram an adapter or wrapper has to be defined to relate the design service to its existing implementation, linking them in activity D4.

D4 – Assign components to services. The components that will implement the services generated must be defined and shown in the components diagram. For each service, a component with which to implement it has to be defined. SoaML provides the participant component with which to define the implementation of participants and services, defining new components to be generated and, if one exists, defining adapters or wrappers to use it.

D5 – Define services interaction. Service interaction can be defined as the orchestration or choreography of services, [4][5], as is done for business processes. This insight is provided by a sequence diagram showing all the services, or by various diagrams showing subsets of services for different sub-processes in the BP. That activity has no corresponding diagram in SoaML, so it is shown by a UML sequence diagram.

3 From BPMN Models to SoaML Models and Beyond

The BPSOM methodological and automated guide is used to derive and generate services from BP models, thus constituting the basis for its implementation. BPSOM defines how to derive services from BP in a conceptual manner. It identifies the participants involved and the services they provide and request, along with the associated contracts and interfaces, parameters, and the messages exchanged, using the SoaML standard. The automation in BPSOM focuses on the generation of services from BP by means of QVT transformations defined between the SoaML and the BPMN metamodels. We follow the MDA approach based completely on the use of OMG standards. The BPMN BP model constitutes the CIM, and the SoaML service model the PIM, which can then be used to generate code, using MDA engines. The QVT transformations are based on a defined ontology [18] which relates BP models to service models, conceptualizing their elements and relationships. Fig. 6 shows an overview of the relationship between the BP in Fig. 2 and ServicesArchitecture in Fig. 3. The QVT transformations code itself is not shown here, as it can be seen in [15].

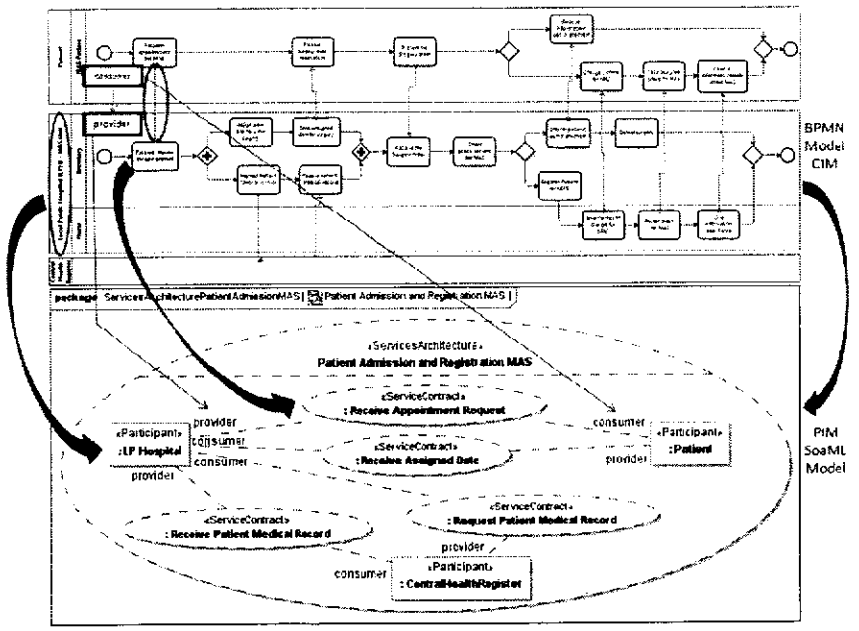

Fig. 6. BPMN to SoaML QVT transformations definitions for the example

We use the Eclipse environment in MINERVA, integrating several plug-ins to support BPSOM definitions, including MediniQVT as the QVT engine. The BPMN BP model is created by business and software people in a BPMN modeler which exports the model. It is then loaded into Eclipse and marked by the Architect with information to apply the QVT transformations, obtaining participants and its ports. To generate the code, the MDA engine needs all the SoaML diagrams, completed later by developers. For the example, we have integrated the MagicDraw Cameo SOA+ and ModelPro Eclipse plug-ins, which can be downloaded from [19] with the BPMN BP and SoaML services models, QVT transformations and input and output XMI files.

6 Related Work

We carried out a systematic review regarding the application of SOC and MDD paradigms to BP, presented in [16]. To the best of our knowledge, there is no other work that relates BPMN models directly to SoaML models the way we do. Regarding the methodological approach, BPSOM has been defined over the same period as other proposals shown in [11][16]. Nevertheless, it is worth mentioning [20], which defines a methodology for service development focusing on WS, the survey of methodologies presented in [21], as well as a consolidated methodology for defining business and software services, the SOMA plug-in for the RUP [22], which, as ours does, adds activities, but to RUP, and Shape [23] ,which also uses SoaML, but with different guides and no generation. For the model driven approach it is worth mentioning [24], which defines guidelines and transformations from one model to other, [25] proposing a method for service composition with a process to model generation, metamodels and artifacts to be obtained, adding in [26] a value model for deriving services using ATL [27].This is also used in [28], in which models, metamodels and transformations are defined, moving from collaborative BP to a SOA model, generating BPEL. Our proposal differs from these in several ways: firstly, BPSOM can be added to any existing base software development process, thus promoting reuse and making it easier to adopt. Secondly, QVT transformations are integrated in the development environment, obtaining the models from which to generate code. Thirdly, the conceptual and automatic guide is fully integrated in BPSOM. Finally, MINERVA framework integrates existing standards, promoting standardization of development.

7 Conclusions and Future Work

BPSOM has been defined to guide service development from business processes, integrated into MINERVA framework for continuous BP improvement. Its contributions are as follows: it allows the reuse of existing knowledge in the developing organization, by using the base software development process, adding specific elements for service development. The use of the SoaML standard to model services supports the definition of meaningful elements in specifying services from BP, in both a conceptual and an automatic way. Finally, we have defined QVT transformations from the BPMN metamodel to the SoaML metamodel that can be executed in the Eclipse environment, obtaining an initial definition of service models. These QVT transformations were defined for previous versions of BPMN and SoaML, so we are updating

and completing them using the BPMN 2.0 and SoaML beta2 standards recently released by OMG. There are few implementations of SoaML, so we are developing our own to show the service models graphically. From these diagrams, code can be generated using existing MDA engines. We are working on case studies at the Ciudad Real General Hospital to validate the proposal.

Acknowledgments. This work has been partially funded by the Agencia Nacional de Investigación e Innovación (ANII,Uruguay), ALTAMIRA project (Junta de Comunidades de Castilla-La Mancha, Spain, F. Soc. Europeo, PII2I09-0106-2463), PEGASO/MAGO project (Ministerio Ciencia e Innovacion MICINN, Spain, FEDER, TIN2009-13718-C02-01) and INGENIOSO project (Junta de Comunidades de Castilla-La Mancha, Spain, PEII11-0025-9533).

References

1. Weske, M.: BPM Concepts, Languages, Architectures. Springer, Heidelberg (2007)
2. Smith, H., Fingar, P.: Business Process Management:The third wave. Meghan-Kieffer, Tampa (2003)
3. Papazoglou, M., Traverso, P., Dustdar, S., Leymann, F.: Service-Oriented Computing: State of the Art and Research Challenge. IEEE Computer Society, Los Alamitos (2003)
4. Krafzig, D., Banke, K., Slama, D.: Enterprise, SOA, Best Practices. Prentice-Hall, Englewood Cliffs (2005)
5. Erl, T.: SOA: Concepts, Technology, and Design. Prentice-Hall, Englewood Cliffs (2005)
6. Mellor, S., Clark, A., Futagami, T.: Model Driven Development. IEEE Comp.Society, Los Alamitos (2003)
7. Object Management Group (OMG), Model Driven Architecture, MDA (2003)
8. Object Management Group (OMG), SOA Modeling Language, SoaML (2009)
9. Delgado, A., Ruiz, F., García-Rodríguez de Guzmán, I., Piattini, M.: MINERVA: Model drIveN and sErvice oRiented framework for the continuous business process improVement and relAted tools. In: Dan, A., Gittler, F., Toumani, F. (eds.) ICSOC/ServiceWave 2009. LNCS, vol. 6275, pp. 456–466. Springer, Heidelberg (2010)
10. Delgado, A.: MINERVA framework (2010), http://alarcos.esi.uclm.es/MINERVA/
11. Delgado, A., Ruiz, F., García - Rodríguez de Guzmán, I., and Piattini, M.: Towards a Service-Oriented and Model-Driven framework with business processes as first-class citizens. In: 2nd International Conference on BP and Services Computing, BPSC 2009 (2009)
12. Object Management Group (OMG), Query/Views/Transformations, QVT (2008)
13. Object Management Group (OMG), Business Process Modeling Notation, BPMN (2009)
14. Graham, I., Henderson-Sellers, B., Younessi, H.: The OPEN Process Specification. ACM Press, Addison-Wesley(1997)
15. Delgado, A., García - Rodríguez de Guzmán, I., Ruiz, F., Piattini, M.: From BPMN business process models to SoaML service models: a transformation-driven approach. In: 2nd Int.Conf. on Software Tech. and Engineering (ICSTE 2010), San Juan de Puerto Rico (October 2010)
16. Delgado, A., Ruiz, F., García-Rodríguez de Guzmán, I., Piattini, M.: Application of service-oriented computing and model-driven development paradigms to BP: a systematic review. In: 5th Int. Conf. on SW and Data Technologies (ICSOFT 2010), Athens (2010)
17. Object Management Group (OMG), Business Motivation Model, BMM (2010)
18. Delgado, A., Ruiz, F., García-Rodríguez de Guzmán, I., Piattini, M.: Towards an ontology for SO modeling supporting BP. In: 4th. Int. Conf. on Research Challenges IS, RCIS 2010 (2010)

19. Delgado, A.: BPSOM methodology example (2010), http://alarcos.esi.uclm.es/MINERVA/BPSOM/BPSOMexample.zip
20. Papazoglou, M., van den Heuvel, W.: Service-oriented design and development methodology. Int. J. Web Engineering and Technology 2(4), 412–462 (2006)
21. Kohlborn, T., Korthaus, A., Chan, T., Rosemann, M.: Identification and Analysis of Business and SE Services- A Consolidated Approach. IEEE Transactions on Services Comp. (2009)
22. IBM-SOMA, http://www.ibm.com/developerworks/rational/downloads/06/rmc_soma/
23. Stollberg, M., et al.: A Customizable Methodology for the MDE of Service-based System Landscapes. In: 4th Workshop on Modeling, Design, and Analysis for the Service Cloud (MDA4ServiceCloud 2010), with ECMFA 2010, Paris (June 2010)
24. Herold, S., Rausch, A., Bosl, A., Ebell, J., Linsmeier, C., Peters, D.: A Seamless Modeling Approach for service-oriented IS. In: 5th Int. Conf. on IT:New Generations, ITNG 2008 (2008)
25. de Castro, V., Marcos, E., López Sanz, M.: A model driven method for service composition modelling: a case study. Int. J. Web Engineering and Technology 2(4) (2006)
26. de Castro, V., Vara Mesa, J.M., Herrmann, E., Marcos, E.: A Model Driven Approach for the Alignment of Business and Information Systems Models (2008)
27. Jouault, F., Kurtev, I.: Transforming models with ATL (ATLAS Transformation Language). In: Bruel, J.-M. (ed.) MoDELS 2005. LNCS, vol. 3844, pp. 128–138. Springer, Heidelberg (2006)
28. Touzi, J., Benaben, F., Pingaud, H., Lorré, J.P.: A model-driven approach for collaborative service-oriented architecture design. Int. Journal of Prod. Economics 121(1) (2009)