Integrated Service Engineering Workbench: Service Engineering for Digital Ecosystems

Gregor Scheithauer
Siemens Corporate Technology
Information & Communication
Otto-Hahn-Ring 6
81739 Munich, Germany
gregor.scheithauer.ext @siemens.com

Konrad Voigt
SAP Research
CEC Dresden
Chemnitzer Str. 48
01187 Dresden, Germany
konrad.voigt@sap.com

Veli Bicer
FZI Forschungszentrum Informatik
Information Process Engineering
Haid-und-Neu-Straße 10-14
76131 Karlsruhe, Germany
bicer@fzi.de

Matthias Heinrich
SAP Research
CEC Dresden
Chemnitzer Str. 48
01187 Dresden, Germany
matthias.heinrich@sap.com

Anja Strunk
TU Dresden
Computer Networks
Nöthnitzer Str. 46
01187 Dresden, Germany
anja.strunk@tu-dresden.de

Matthias Winkler
SAP Research
CEC Dresden
Chemnitzer Str. 48
01187 Dresden, Germany
Matthias.winkler@sap.com

ABSTRACT
The evolution of service-oriented architectures towards digital ecosystems comprehends a number of challenges. According to Papazoglou et al., one challenge is to develop services in a dynamic environment with high uncertainties, and in collaboration with other companies. The Integrated Service Engineering (ISE) Workbench is a step in this direction in that it supports a model-driven approach and different stakeholders. This paper briefly introduces the ISE Framework, presents the ISE Workbench and its functionality, and proposes a demonstration of a conducted case study in the IT outsourcing domain.

Categories and Subject Descriptors
H.4 **[Information Systems Applications]:** Miscellaneous; D.2.8 **[Software Engineering]:** Miscellaneous;

General Terms
Documentation, Design, Economics.

Keywords
Service engineering, business modeling, BPM, MDA, case study.

1. INTRODUCTION
Tertiarisation describes a structural change in developed countries concerning the sectoral composition. Countries shift from an industry economy toward a service economy. Sources of this change include Globalization, technological change, and an in

MEDES 2009, October 27-30, 2009, Lyon, France.

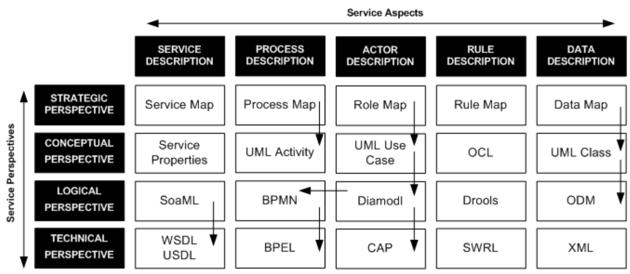

Figure 1: ISE Framework (cf. [6]).

-creasing demand for services [12]. Considering this trend, it becomes clear that services and the service economy play an important role in today's and tomorrow's business. In line with this trend, digital ecosystems [3] emerge, such as eBay, Google Base, Amazon.com, SalesForce.com, and SAP Business by Design. Such market places allow *to trade* services between different legal bodies.

One of many challenges in this domain is to engineer services [11]. Existing software engineering methodologies are inappropriate for service-oriented design due to highly dynamic environments, high uncertainties, distributed control of processes, and many different stakeholders design time, which holds also true for digital ecosystems and its peculiarities.

Kett et al. [6] took these new developments into consideration and developed the Integrated Service Engineering (ISE) Framework in order to address these new challenges and to embrace the concept of model-driven design. The ISE Workbench is an implementation of the ISE Framework in that it integrates various notations for different modeling purposes and offers the technology to derive deployable software artifacts. The tool is a joint effort of the Theseus/TEXO research project [1], which aims at realizing the Internet of Services (IoS).

The remainder of this paper is structured as follows: section 2 introduces the ISE Framework. Section 3 presents the ISE Workbench

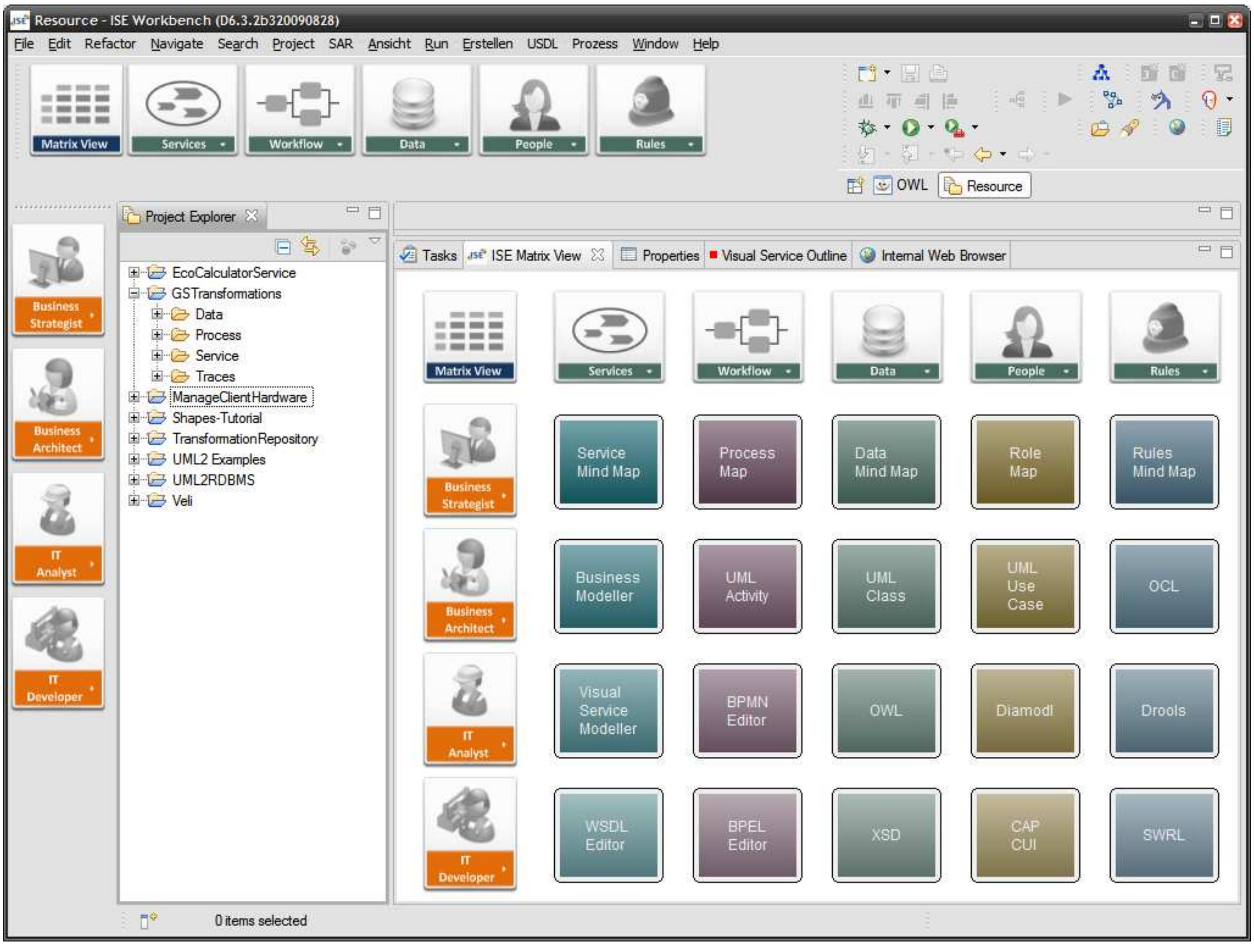

Figure 2: The ISE Workbench.

with its functionality and model transformation capabilities. Following this, section 4 proposes a demonstration on the basis of a case study in the IT outsourcing domain. Finally, section 5 concludes this work and discusses future work.

2. THE ISE FRAMEWORK

Based on a state-of-the-art study of existing frameworks, Kett et al. [6] argued that existing frameworks for service engineering either address the business perspective or the technical perspective. To overcome the gap between these approaches they introduced the ISE Framework, which Figure 1 shows. The framework builds on the Zachman framework [13] and a service engineering methodology for service products [4]. The vertical axis shows four perspectives of the engineering process and is named *service perspectives*. Each perspective relates to a specific role with appropriate skills and offers different sets of tools and methods. It also implies the chronology of the framework for they are linked to phases of the service engineering process. The horizontal axis shows five different *descriptions of a service*. Each description is valid for each perspective. Each intersection in the matrix is placeholder for a meta model, a notation, and activities, which are appropriate for the respective perspective and the modeling aspect.

2.1 Service Perspectives

Business strategists pick up new service ideas and focus on requirement analysis in the *strategic perspective*. Kett et al. [6] depicted a basic underlying model for this perspective: the Business Model Ontology (BMO) [10]. Eventually, a decision is made whether to implement a new service or not. The *conceptual perspective* focuses on operationalizing and implementation of strategic artifacts from the owner's perspective. The final artifact is a service design which is neither technical nor platform-specific. Conceptual artifacts are transformed into formal models during the *logical perspective* by IT analysts. This perspective offers a bridge between service design and technical service implementation. Finally, the IT developer transforms the logical artifacts into platform-dependent software artifacts, e.g., BPEL [2] and WSDL [5], etc., during the *technical perspective*.

2.2 Service Aspects

The *service description* embodies services' value proposition toward potential customers. This includes functional, financial, legal, marketing, and quality of service properties as well as other meta data for service proposition, discovery, selection, contracting, and monitoring. The *process description* addresses services' behavioral aspect, which includes core capabilities and sequence flows. The *actor description* offers means to model and to refine human resources, and to assign tasks. Intangible assets, terms, and concepts as well as their relationships are defined in the *data de-*

scription. The *rule description* addresses organizational rules. These are defined to elicit and formalize domain knowledge to guide services' behavior.

3. THE ISE WORKBENCH

The Integrated Service Engineering (ISE) Workbench implements the ISE Framework (cf. Figure 2) and supports an interdisciplinary structured service engineering process to develop services that can be traded over the Internet. The work on the workbench started in April 2008 and is a prototype, which is still under development. Developers add new features as well as improve existing ones. For example, the business rule aspect is not implemented, yet. The ISE Workbench builds on Eclipse's Rich Client Platform (RCP), which allows an integration of existing tools as well as offers a platform for novel tool development. The workbench embodies a total number of 20 editors in order to model the five service aspects for each of the four perspectives. OMG's Query View Transformation (QVT) specification is the basis for model transformation implementation, e.g. BPMN [9] to BPEL [2]. The remainder of this section elaborated on its main functionality and notations, model transformations, and deployment.

3.1 Main Functionality & Notations

In order to support the ISE Framework with its 20 intersections, available notations were analyzed. Figure 1 depicts the resulting 20 modeling notations. This set of notations is only one possible selection. For each chosen notation, a suitable editor was integrated into the workbench to design all service aspects from different angles. The *strategic perspective* uses the mind map notation to elicit the information depicted by Kett et al. The *conceptual perspective* employs mostly the UML diagrams [7], a semi-formal graphical notation. Whereas, the *logical perspective* makes use of formal notations, the *technical perspective* applies formal languages, such as BPEL [2] and WSDL [5].

Next to existing notations, new languages were developed, where necessary. The *service property* notation is a domain-specific language and describes services from a provider's perspective in a non-technical fashion and includes information about capabilities, price & payment, delivery channels, rating, legal aspects, and provider details in order to facilitate service discovery. The *Universal Service Description Language (USDL)* is a XML specification that holds facts about business information, operational information, and technical information related to the service. The *Canonical Abstract Prototypes* (*CAP)* editor provides an abstract description of a user interface structure. Finally, the *service archive (SAR)* is an XML schema and denotes how to bundle technical models for deployment.

3.2 Model Transformations

The ISE Workbench offers model transformation for flexibility, speed, and accuracy in design. Each editor relies on a meta model designed with the Eclipse Modeling Framework (EMF). A model transformation relates elements within two meta models, e.g., an UML swim lane relates to a BPMN pool.

The ISE Workbench shows how these modeling notations and languages correlate and how they are used interchangeably for service engineering. The implementation of transformation scripts are codified with QVT.

Two types of transformation exist. On the one hand, there are transformations between service aspects, e.g., BPMN and Diamodl; on the other hand, there are transformations between service perspectives, e.g., SoaML [8] and WSDL. The arrows in Figure 1 depict available transformations. Current and future works address further transformation implementations.

3.3 Deployment

The ISE Workbench offers deployment capabilities for seamless service execution. The service archive (SAR) is an XML schema and denotes how to bundle technical models. After service design with the ISE Workbench, the tool generates a SAR file and deploys it on a service runtime environment.

4. DEMONSTRATION

This section illustrates the proposed demonstration. The demonstration consists of three parts: (1) a brief introduction of the ISE Workbench, (2) the live-implementation of a business service, and (3) an illustration of the transformation implementation. The demonstration shows a running version of the ISE Workbench to explain the tool, the implementation of the scenario, and the execution of model transformations. The demonstration concludes with a general discussion about the tool's strength and weaknesses to further improve the ISE Workbench.

A real-world business service forms the basis for demonstrating the ISE Workbench. A multi-national company owns a business service, namely the *Manage Client Hardware*, which allows outsourcing the purchase and the maintenance of computer hardware e.g., a desktop PC.

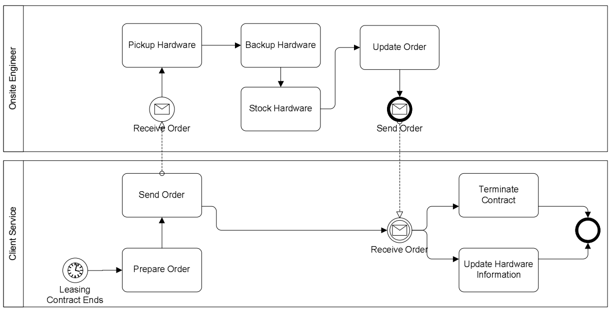

Figure 3: End of a Leasing Contract.

One scenario in this case study is the end of a leasing contract, which Figure 3 shows[1]. This event triggers a client service to prepare and to send an order to an onsite engineer in order to handle hardware at customers' sites. On-site engineers pickup the hardware and perform backups. They then put the hardware in one of the stock locations. Finally, onsite engineers update information in the order message, which comprises hardware status in form of a journal as well as hardware locations. The order message is then sent back to a client service, who uses the order information to terminate the contract and to update hardware information, which concludes the service.

Firstly, the demonstration addresses the *strategic perspective* using the mind map editors to scope information for each of the

[1] It was necessary to modify the scenario for publication. The scenario's scope and its complexity remain the same, nevertheless.

service aspects. Following this, the presenter generates UML diagram skeletons from the process mind map and the role mind map during the *conceptual perspective*. He eventually adds further information and details business processes, data entities, service description, and use cases. This concludes the design phase and starts the implementation phase. During the *logical perspective,* for each conceptual diagram, formal models including BPMN and OWL are generated. These diagrams are completed with formal information including message exchange. Lastly, the workbench transforms these models into interpretable languages such as BPEL and WSDL.

5. CONCLUSION & FUTURE WORK

This paper introduced the Integrated Service Engineering (ISE) Workbench, which implements the ISE Framework [6] and supports the whole engineering process of tradable services. This paper shows the tool's main functionality and its transformation capabilities. Additionally, a demonstration was proposed. Future work includes further development of model transformations as well as the integration of the *rule* service aspect. Additionally, it is envisioned to pursue further case studies to improve the modeling experience as well as to gather requirements from different business service domains.

6. ACKNOWLEDGMENTS

The project was funded by means of the German Federal Ministry of Economy and Technology under the promotional reference '01MQ07012'. The authors take the responsibility for the contents.

7. REFERENCES

[1] Theseus TEXO – Business Webs in the Internet of Services. http://www.theseus-programm.de/en-us/theseus-application-scenarios/texo/default.aspx. last accessed: 2009-03-05.

[2] Alexandre Alves, Assaf Arkin, Sid Askary, Charlton Barreto, Ben Bloch, Francisco Curbera, Mark Ford, Yaron Goland, Alejandro Guízar, Neelakantan Kartha, and Canyang Kevin Liu. Specification: Business Process Execution Language for Web Services version 2.0, January 2007.

[3] Alistair P. Barros and Marlon Dumas. The Rise of Web Service Ecosystems. *IT Professional*, 8(5):31–37, 2006.

[4] Hans-Jorg Bullinger, Klaus-Peter Fahnrich, and Thomas Meiren. Service Engineering – Methodical Development of new Service Products. *International Journal of Production Economics*, 85(3):275–287, September 2003.

[5] Roberto Chinnici, Jean-Jacques Moreau, Arthur Ryman, and Sanjiva Weerawarana. Specification: Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. W3C Recommendation, 6 2007.

[6] Holger Kett, Konrad Voigt, Gregor Scheithauer, and Jorge Cardoso. Service Engineering in Business Ecosystems. In *Proceedings of the XVIII. International RESER Conference*, Stuttgart, Germany, September, 25 - 26 2008.

[7] Object Management Group (OMG). Specification: Unified Modeling Lanugage (UML) version 2.0. http://www.omg.org/docs/formal/05-07-04.pdf, July 2007.

[8] Object Management Group (OMG). Specification: UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms, v 1.1. http://www.omg.org/technology/documents/formal/QoS_FT.htm, April 2008.

[9] Object Management Group (OMG). Specification: Business Process Modeling Notation (BPMN), Version 2.0. http://www.omg.org/docs/dtc/09-08-14.pdf, Aug 2009.

[10] Alexander Osterwalder. *The Business Model Ontology: A Proposition in a Design Science Approach*. PhD thesis, Universite de Lausanne Ecole des Hautes Etudes Commerciales, 2004.

[11] Mike P. Papazoglou, Paolo Traverso, Schahram Dustdar, and Frank Leymann. Service-oriented computing: a research roadmap. *Int. J. Cooperative Inf. Syst.*, 17(2):223–255, 2008.

[12] Michael Peneder, Serguei Kaniovski, and Bernhard Dachs. What Follows Tertiarisation? Structural Change and the Role of Knowledge-based Services. *The Service Industries Journal*, 23 Issue 2(146):47–66, March 2003.

[13] J. A. Zachman. A Framework for Information Systems Architecture. *IBM Systems Journal*, 26(3):276–292, 1987.