

2009

The two cultures and the internet revolution

Thomas Kohlborn

Queensland University of Technology, t.kohlborn@qut.edu.au

Axel Korthaus

Queensland University of Technology, axel.korthaus@qut.edu.au

Taizan Chan

Queensland University of Technology, t.chan@qut.edu.au

Michael Rosemann

Queensland University of Technology, m.rosemann@qut.edu.au

Follow this and additional works at: <http://aisel.aisnet.org/ecis2009>

Recommended Citation

Kohlborn, Thomas; Korthaus, Axel; Chan, Taizan; and Rosemann, Michael, "The two cultures and the internet revolution" (2009).
ECIS 2009 Proceedings. 155.

<http://aisel.aisnet.org/ecis2009/155>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISEL). It has been accepted for inclusion in ECIS 2009 Proceedings by an authorized administrator of AIS Electronic Library (AISEL). For more information, please contact elibrary@aisnet.org.

SERVICE ANALYSIS – A CRITICAL ASSESSMENT OF THE STATE OF THE ART

Thomas Kohlborn, Business Process Management Group, Information Systems Program,
Faculty of Science and Technology, Queensland University of Technology, Brisbane,
Australia, t.kohlborn@qut.edu.au

Axel Korthaus, Business Process Management Group, Information Systems Program, Faculty
of Science and Technology, Queensland University of Technology, Brisbane, Australia,
axel.korthaus@qut.edu.au

Taizan Chan, Information Technology Cluster, Information Systems Program, Faculty of
Science and Technology, Queensland University of Technology, Brisbane, Australia,
t.chan@qut.edu.au

Michael Rosemann, Business Process Management Group, Information Systems Program,
Faculty of Science and Technology, Queensland University of Technology, Brisbane,
Australia, m.rosemann@qut.edu.au

Abstract

The service-orientation paradigm has not only become prevalent in the software systems domain in recent years, but is also increasingly applied on the business level to restructure organisational capabilities. In this paper, we present the results of an extensive literature review of 30 approaches related to service identification and analysis for both domains. Based on the consolidation of a superset of comparison criteria for service-oriented methodologies found in related literature, we compare and evaluate the different characteristics of service engineering methods with a focus on service analysis. Although a close business and IT alignment is regarded as one of the core beneficial promises of service-orientation, our analysis suggests that there is a lack of unified, comprehensive methodology for service identification and analysis integrating and addressing both domains. Thus, we discuss how our results can inform directions for future research in this area.

Keywords: business-IT alignment, service analysis, service identification, service orientation.

1 INTRODUCTION AND MOTIVATION

The emerging and accelerating trend towards business networks of collaborating business partners and the anticipation of entire service ecosystems comprising dedicated service marketplaces increase the need for enterprises to engage in service-orientated approaches to be able to gain the flexibility and agility required to succeed in this changing environment (Cherbakov et al. 2005, Sanz, Nayak and Becker 2006). It is no longer sufficient to apply the paradigm of service-orientation to the advancement of technical software architectures only. Instead, entire business architectures are about to be restructured from a services point of view in order to allow flexible and reconfigurable collaboration scenarios in business networks. While services in general can be seen as abstract resources that represent consumable capabilities (W3C 2004) offered by a service provider who performs specific actions on behalf of a service consumer at a certain point in time and place and through some channel (Dumas et al. 2001), to be more precise, we thus have to differentiate between the following two basic views of services:

- First, service-orientation on the business level enables organisations to expose and offer operations as *business services* to business partners in order to facilitate on-demand collaboration opportunities. A business service is the outcome of a specific “chunk of operation” that is performed by an organisation (Sanz, Nayak and Becker 2006). These services can have different levels of granularity ranging from comprehensive offerings (e.g. purchasing services) to fine granular services (e.g. address verifications) (Bieberstein et al. 2005).
- Second, to enable business services and thus support the agility of organisations, service-orientation on the technical level fosters the utilisation of *software services* and enables a close business and IT alignment (Cherbakov et al. 2005). Software services expose application functionalities that can be re-used and composed based on business needs. Hence, a software service supports the execution of a business service.

These two different views are both addressed by the Organization for the Advancement of Structured Information Standards (OASIS) in their definition of Service-Oriented Architecture (SOA) as a “paradigm for organising and utilising distributed capabilities that may be under the control of different ownership domains” (OASIS 2006). Since these capabilities can relate to business functions and to application functionalities as well, this definition supports a holistic SOA view.

As a consequence of the proliferation of the service idea on both the business and the software level, there is now a demand for service engineering methodologies that cover both business and software services and provide an integrated, holistic approach to ensure business and IT alignment and agility. Service engineering in general is still regarded as a research challenge in the literature about current SOA research roadmaps (e.g. Papazoglou et al. 2007, Kontogiannis 2007).

In our research, we aim at identifying an existing or developing a derived methodology for service analysis that accounts for the holistic view of service-orientation with its two domains of application, the business and the software domain. Service analysis is one of the earliest phases in a service engineering process, which covers the whole lifecycle of a service. It is of particular importance, as any errors made during this phase can flow through to and build up in the design and implementation phases, which results in increased cost due to necessary rework (Inaganti and Behara 2007). Not surprisingly, the two basic views of services introduced above are also reflected in the propositions about the scope of service analysis (see, for example, the different definitions by Marks and Bell 2006, p. 58, and Papazoglou and van den Heuvel 2006, p. 417). In this paper, we view service analysis as a comprehensive phase where the concept of service-orientation is applied to analyse the capabilities provided by an organisation and to consequently identify services that are currently supported and could be supported by IT. This notion includes the analysis of the impact of service-orientation on the

business level as well as on the technical level, which is congruent with the notion and scope of SOA introduced earlier.

As a first step towards achieving our higher research goal, the objective in this paper is to begin with a contemporary overview and comparison of existing major service analysis-related service engineering methods covering business and technical viewpoints in order to analyse their characteristics in detail and to evaluate to which extent these approaches are ready for the emerging requirements as stated above. Such a comparison and evaluation will not only point out possible gaps and necessary directions for research, it will also provide a general overview of the methods that can be chosen to start a SOA endeavour.

This paper is structured as follows. The second chapter will provide an overview of related work in the area of service-orientation methods comparison. Subsequently, we will present the criteria that have been used to compare and analyse extant approaches for service analysis and we will define the composition of our sample in the third chapter. The fourth chapter represents the actual comparison and the evaluation of the approaches regarding the selected criteria. Subsequently, we will discuss the findings and directions for further research. The final chapter will provide a conclusion.

2 RELATED WORK

The domain of service analysis is still rather young. Although the number of service-oriented methods has grown significantly, our literature review yielded only a rather narrow set of papers dedicated to the comparison of existing service analysis approaches. Moreover, these papers exclusively focus on the analysis of approaches related to software services and are very limited in regard to the number of methods compared. We essentially identified three papers on the topic (namely Klose, Knackstedt and Beverungen 2007, Kohlmann and Alt 2007 and Ramollari, Dranidis and Simons 2007) that provided sets of criteria for the comparison of service engineering methods with a focus on service analysis. For an overview of these criteria, the reader may refer to Table 1, which is presented and discussed in the next section.

Klose, Knackstedt and Beverungen (2007) not only compare nine existing approaches and, as a result, note the lack of a unified method for identifying services, but also present their own proposed method, thus adding to the set of existing methods. The objective of their comparison of different methods is limited to an overview of the width and depth of a few existing methods. Kohlmann and Alt (2007) compare six methods related to service analysis and modelling, particularly focusing on their criteria on the set of activities relevant for those methods. Finally, Ramollari, Dranidis and Simons (2007) compare ten methods, which are related to the development of SOAs in general. Thus, the analysed methodologies have typically a wider scope than just service analysis.

The authors of these publications conducted their method comparisons for different reasons, so that the employed criteria are not completely congruent. For example, Kohlmann and Alt (2007) seem to have chosen criteria that reflect the strengths of their own proposed method, while Ramollari, Dranidis and Simons (2007) provide a more general overview of service-oriented development methodologies ideally covering the complete life cycle of a service.

For this paper, we have consolidated the criteria already used in the three analysed related research publications based on a clustering process (see next section) to provide a comprehensive set of criteria for the comparison of service analysis approaches for both business and software services, thus catering to the underlying comprehensive notion of SOA and services as introduced previously.

3 SELECTION OF COMPARISON CRITERIA AND SAMPLE

3.1 Selecting the criteria

To be as neutral as possible and to increase the validity of our research by building on the existing body of knowledge, we decided to make the superset of the criteria used in the three analysed related research papers the basis for the set of criteria to be used in our more comprehensive method comparison. As there were overlaps in the original criteria, we had to consolidate the superset based on semantic similarities. This task was conducted independently by two different coders and resulted in clusters that were almost 90% identical. In a discussion process, the differences were debated and resolved, thus leading to Table 1, which not only shows all the criteria used in the three related papers, but also their grouping in the new clusters and the labels assigned to these clusters. These labels eventually represent the criteria that we use in our own comparison. In the following, we will describe our criteria, briefly explain the grouping of the criteria from the analysed papers as reflected in the rows of Table 1 and also define symbols and abbreviations used in our comparison overview in Table 2.

Publications / Description of used criteria	Klose, Knackstedt, Beverungen (2007)	Kohlmann, Alt (2007)	Ramollari, Dranidis, Simons (2007)
SOA concept	<ul style="list-style-type: none"> Employed SOA concept 		
Delivery strategy for SOA	<ul style="list-style-type: none"> Background and starting point 	<ul style="list-style-type: none"> Business- or technical-driven service identification 	<ul style="list-style-type: none"> Delivery strategy
Lifecycle coverage	<ul style="list-style-type: none"> Covering of SOA design phases 		<ul style="list-style-type: none"> Lifecycle coverage
Degree of prescription			<ul style="list-style-type: none"> Degree of prescription Process agility
Accessibility and validity	<ul style="list-style-type: none"> Documentation of the method 		<ul style="list-style-type: none"> Availability Industrial application
Adoption of existing processes/approaches/ techniques/notations	<ul style="list-style-type: none"> Proposal of IT criteria for service identification Application of process models for service identification 	<ul style="list-style-type: none"> Domain analysis Visualisation with service landscapes/maps Alignment with process model 	<ul style="list-style-type: none"> Adoption of existing processes / techniques / notations
Regard to stakeholders	<ul style="list-style-type: none"> Regard to stakeholders for service identification 		<ul style="list-style-type: none"> Supported roles
Service classification and clustering	<ul style="list-style-type: none"> Service hierarchies and classification scheme 	<ul style="list-style-type: none"> Examination of service cut Service specification Service clustering 	
Additional characteristics		<ul style="list-style-type: none"> Alignment with sourcing strategy 	

Table 1. Clustering of previously used criteria to compare service analysis-related methods

SOA concept: To analyse whether both service concepts (business and software services) are supported, we selected *SOA concept* as a criterion reflecting whether an approach's primary focus is on the derivation of business services (BS), software services (SS) or both (BS/SS). Klose, Knackstedt and Beverungen (2007) have utilised a similar criterion, but as their focus is restricted on software services only and, in addition, it is not clear what the foundation and the possible values of their criterion actually are, we abstracted from these specific values and only analyse if the SOA concept rather refers to business services or software services.

Delivery strategy for SOA: To address organisation-specific requirements (e.g. the need to leverage existing legacy systems), different *delivery strategies for a SOA* are conceivable. Hence, we need to analyse whether a particular approach supports the top-down strategy (T), where services are derived based on the analysis of business requirements (Erl 2005), the bottom-up strategy (B), which focuses on the derivation of services based on an analysis of legacy systems on an as-needed basis (Sneed

2006), or the meet-in-the-middle strategy (M) that combines the other two strategies (Inaganti and Behara 2007). The delivery strategy criterion was included to reflect which strategy an approach recommends to implement a SOA. To provide information about the particular starting point and focus of an approach, we included a textual comment in our analysis. Klose, Knackstedt and Beverungen (2007) call their related criterion ‘Background and starting point’, Kohlmann and Alt (2007) roughly differentiate between technical and business-driven identification and Ramollari, Dranidis and Simons (2007) just distinguish the three basic SOA delivery strategies mentioned above.

Lifecycle coverage: While some proposed SOA development approaches aim at offering support for the full *SOA lifecycle*, other approaches are more focused on a subset of the activities to be performed in the life of a SOA. This criterion has been derived by combining the criteria ‘Lifecycle coverage’ used by Ramollari, Dranidis and Simons (2007) and ‘Covering of SOA design phases’ as proposed by Klose, Knackstedt and Beverungen (2007). In our analysis, we use a trivalent scale (0, +, ++), with the following semantics: 0 stands for methods that focus on service identification and analysis only, while + represents methods with a service analysis and design focus and ++ finally marks more comprehensive approaches that include phases like implementation etc.

Degree of prescription: A service analysis method can be rather *prescriptive* and define a rigid, heavy-weight process with lots of details, or it can describe a more lightweight, flexible, less structured process that is adaptable and allows for an *agile* approach. This criterion is based on the ‘Degree of prescription’ criterion and the ‘Process agility’ criterion both introduced by Ramollari, Dranidis and Simons (2007), as we considered the two to be highly related. In our analysis, we use a trivalent scale (0, +, ++), with the following semantics: 0 stands for methods that are very lightweight, while + represents methods with a moderate degree of prescription and ++ marks highly prescriptive approaches.

Accessibility and validity: To be useful, a service analysis approach should not only be *well documented*, but the documentation must also be *accessible*, and the validity of the approach should be made clear. The documentation should provide many details, examples, ideas, case studies etc. to provide useful guidance in practice. Sometimes, methodologies proposed by vendors or industry players are *proprietary*. Typically, detailed knowledge about these approaches cannot be easily accessed, whereas non-proprietary approaches are openly available. Moreover, this criterion captures whether an approach has been *validated* or illustrated by presenting real case studies, whether it uses fictitious examples only or even remains on a purely theoretical level without any examples at all. Klose, Knackstedt and Beverungen (2007) (‘Documentation of the method’) as well as Ramollari, Dranidis and Simons (2007) (‘Availability’ and ‘Industrial application’) apply corresponding criteria for their comparisons. In our analysis, we used a three-valued tuple to describe the documentation (textual comment), the availability (trivalent scale with 0 standing for a proprietary, not openly available method, + representing a method that is at least partially documented for public use, e.g. in the form of papers about single activities that constitute the method, and ++ denotes a fully open method) and the validation, e.g. in form of industry case studies etc. (textual comment).

Adoption of existing processes/approaches/techniques/notations: The proposed service analysis methodologies might utilise already *existing techniques, procedures and notations* that can serve as a foundation for the approach. Klose, Knackstedt and Beverungen (2007), for example, use their ‘Application of process models for service identification’ criterion to explicitly analyse if methods consider process models as input. Moreover, they introduce the ‘Proposal of IT criteria for service identification’ criterion to examine whether existing IT criteria (e.g. design principles, such as high cohesion and loose coupling) are proposed for service identification. Under the umbrella of their ‘Adoption of existing processes/techniques/notation’ criterion, Ramollari, Dranidis and Simons (2007) similarly analyse if methods propose reusing proven existing processes, techniques and standardised notations. Also, a number of criteria used by Kohlmann and Alt (2007) can be assigned to this cluster. For example, they analyse whether an approach comprises a ‘Domain analysis’, whether there is an ‘Alignment with process models’ and whether the emerging SOA architecture is illustrated using a

‘Visualisation with service landscapes/maps’. In our own comparison, we describe the results for this criterion in the form of a textual comment to cope with the variety of possible result values.

Regard to stakeholders: As service analysis methods should address the *requirements of potential stakeholders* regarding services, Klose, Knackstedt and Beverungen (2007) (‘Regard to stakeholders for service analysis’) as well as Ramollari, Dranidis and Simons (2007) (‘Supported roles’) include respective criteria in their analyses. This criterion particularly reflects if the perspective of the service consumer is included or if solely the perspective of the service provider is addressed. To be flexible in our analysis, we use a textual comment to describe the value of this criterion.

Service classification and clustering: The criterion *service classification and clustering* describes if the method distinguishes different kinds of services. As the approaches do not use the same terminology for service types, we will only compare the approaches based on the number of different service types as an indication of the level of detail of each approach, similar to Klose, Knackstedt and Beverungen (2007) with their ‘Service hierarchies and classification scheme’ criterion. Kohlmann and Alt (2007) address this aspect implicitly by using the criteria ‘Service specification’ and ‘Examination of service cut’. They also introduce a ‘Service clustering’ criterion, which can be seen as a grouping mechanism above the level of individual services. In our analysis, we use a simple trivalent scale (0, +, ++) to indicate if there is only one single service concept in the method (0), if the method distinguishes between different service types but does not provide details (+) or if the method includes a detailed definition of the different service types (++) .

Additional characteristics: This criterion is a placeholder for any other important characteristics of the analysed methods that seem important enough to be pointed out. For example, Kohlmann’s and Alt’s (2007) analysis if the respective methods postulate alignment with the sourcing strategy can be an aspect that is reflected here among others. We use a textual comment again to express aspects of interest in this category in a flexible way.

Having detailed the different criteria we employed to compare the various service engineering methods with a service analysis focus, the next paragraph will explain the selection process that was used to compile the sample of methods to be analysed.

3.2 Selecting the service analysis-related service engineering methods

After having defined the criteria for the comparison of extant approaches, we had to identify those prominent service analysis-related methods that were the most appropriate candidates for our analysis. We did not aim at a representative sample but rather at a comprehensive mix of approaches reflecting the broad spectrum of different characteristics. To this end, we made sure to not only include academic work, but also approaches developed by the largest providers of packaged business applications such as IBM, Microsoft and SAP (Genovese 2007). We ensured that the sample included both top-down and bottom-up approaches. Another requirement was that each approach to be included explicitly referred to the concept of service-orientation. The sources of the methods are manifold as they range from journals (e.g. Papazoglou and van den Heuvel 2006) to conferences (e.g. Klose, Knackstedt and Beverungen 2007), books (e.g. Erl 2005) and white papers (e.g. SAP 2005). The final sample of analysed extant approaches can be found in the first column of Table 2, which also shows the characteristics of each approach with regard to the selected criteria. The ordering of the list of approaches in Table 2 allows the visual identification of different classes of approaches according to the underlying SOA concept (BS, BS/SS, and SS), which are separated by thicker lines in the table. For the category of approaches that are only concerned with software services (SS), subclasses can be identified depending on the delivery strategy for SOA (T, T/M, M, M/B, or B) separated by dashed lines.

4 METHOD COMPARISON AND EVALUATION

The comparison of the 30 extant service engineering methods as reflected in Table 2 was conducted independently by two coders, whereby the second coder restricted the analysis to a random control sample. The results of the coding process of the control sample were consistent with the results of the original coder. In the following, we describe our observations that resulted from the analysis of the 30 methods for each criterion used in the comparison process. The criterion “Additional characteristics” will not be elaborated on as it is assumed that the description provided in Table 2 is self-explanatory.

SOA concept: SOAs containing primarily business services are less prevalent than SOAs for IT infrastructure. Jones (2006), OASIS (2005) and Sehmi and Schwegler (2006a) propose approaches that do not directly apply to the concept of a business service. Nonetheless, the underlying concepts can be adopted for the identification of business services. Flaxer and Nigam (2004) and IBM (2005) explicitly define business services, but a detailed approach for the identification of these services is missing. Kaabi, Souveyet and Rolland (2004) identify business services based on goal-modelling, which can then be supported by software services.

Approaches	SOA concept	Delivery strategy for SOA	Lifecycle coverage	Degree of prescription	Utility	Adoptions of existing techniques/processes	Regards to stakeholders	Service classification & clustering	Additional characteristics
	{BS/SS/BS+SS}	{T/M/B}, {text}	{0/+/>++}	{0/+/>++}	{{text}, {0/+/>++}, {text}}	{text}	{text}	{0/+/>++}	{text}
Jones (2006)	BS	T, domain decomposition	0	+	examples, 0, n.a.	Decomposition	External stakeholders analysis	++	n.a.
IBM (2005)	BS	T, component decomposition	+	+	examples, +, applied in industry	Decomposition	Customer part of analysis	0	Focus on business components
Flaxer, Nigam (2004)	BS	T, business entities	0	+	examples, +, case studies	n.a.	n.a.	0	Focus on business components
Sehmi, Schwegler (2006a, b)	BS+SS	T, capability decomposition	++	+	examples, +, applied in industry	Decomposition	External stakeholder capabilities	0	n.a.
OASIS (2005)	BS+SS	T, domain decomposition	++	++	examples, 0, n.a.	Related to Jones (2006)	External stakeholders as part of analysis	++	Draft version
Kaabi et al. (2004)	BS+SS	M, goal modelling/IT analysis	+	+	example, 0, n.a.	n.a.	Stakeholder involvement	+	n.a.
Sewing (2006)	SS	T, process models	0	++	examples, 0, focus groups	n.a.	Stakeholder involvement	0	Suitability analysis
Bell (2008)	SS	T, attributes analysis	+	++	examples, 0, n.a.	Own visualisation	Service consumer addressed	+	Focus on service modelling
Adamopoulos (2002)	SS	T, use cases	++	++	case study, 0, case study	Use cases	Consumer as part of use cases	0	n.a.
Kim, Doh (2007)	SS	T, use cases	+	++	example, 0, n.a.	Task tree generation	Consumer as part of use cases	0	n.a.
Kohlmann, Alt (2007)	SS	T/M, process models	+	++	case study, 0, case study	Visualisation with service maps	n.a.	++	Sourcing
Erl (2005)	SS	T/M, process models, entities	++	++	examples, 0, n.a.	Decomposition	n.a.	++	very detailed
Gold-Bernstein, Ruh (2004)	SS	T/M, business events/IT	0	++	examples, 0, n.a.	Use cases	n.a.	+	n.a.
Sun (2006)	SS	T/M, use cases	++	0	n.a., ++, applied in industry	n.a.	Consumer as part of use cases	0	Extensive approach, but no details
Klose et al. (2007)	SS	M, process models/IT	0	++	case study, 0, case study	Design principles	Analysis of takeover and visibility potential	++	n.a.
Erradi et al. (2007)	SS	M, process models/IT	+	0	case study, 0, case study	n.a.	n.a.	++	n.a.
SAP (2005)	SS	M, processes and solution maps/IT	+	+	examples, ++, applied in industry	n.a.	Implicitly by defining process interfaces	++	multiple drivers and principles
Quartel, Dijkman, van Sinderen (2004)	SS	M, processes and application portfolio analysis	0	+	examples, 0, n.a.	n.a.	n.a.	+	Own language
Allen (2007)	SS	M, business requirements/IT analysis	++	0	n.a., +, n.a.	n.a.	n.a.	0	Extensive approach, but no details
Zimmermann, Krogdahl, Gee (2004)	SS	M, domains and processes/IT analysis	++	0	example, ++, n.a.	BPM, OOAD, EA	n.a.	+	n.a.
Stojanovic et al. (2004)	SS	M, use cases/IT	0	0	n.a., 0, n.a.	UML, CBD	n.a.	+	n.a.
Marks, Bell (2006)	SS	M, multiple starting points	+	+	examples, 0, n.a.	n.a.	n.a.	+	comprehensive
Chang, Kim (2007)	SS	M, processes/IT	+	++	case study, 0, case study	OOAD, CBD, SOAD	n.a.	0	n.a.
Arsanjani (2004)	SS	M, domain analysis/IT analysis	++	+	n.a., ++, applied in industry	goal modelling	n.a.	1	n.a.
Papazoglou et al. (2006)	SS	M, process models/IT	++	+	examples, 0, n.a.	n.a.	Service consumer addressed	1	n.a.
Chen et al. (2005)	SS	M/B, domain analysis/source code	+	+	case study, 0, case study	feature analysis	n.a.	0	n.a.
Zhang, Liu, Yang (2005)	SS	B, source code	+	+	case study, 0, case study	OAR, RWSL, SADL/UML	n.a.	+	Invasive legacy transformation
Nadhan (2004)	SS	B, IT analysis	+	0	few examples, 0, n.a.	n.a.	n.a.	+	n.a.
Rahmani et al. (2006)	SS	B, class diagrams	0	+	example, 0, n.a.	MDA, class diagrams, use cases	n.a.	0	n.a.
Sneed (2006)	SS	B, source code	+	+	case study, 0, case study	n.a.	n.a.	0	n.a.

Table 2. Comparison of the 30 approaches

All other approaches focus on the derivation of software services, although the term business service is used to distinguish between services that encapsulate business logic and services that encapsulate application logic.

Delivery strategy for SOA: The delivery strategy for SOA is partially dependent on the underlying SOA concept. Approaches, which address the analysis of business services, postulate a top-down strategy for the delivery of services. Regarding the scope and depth of these approaches, none of them provides a detailed description of how to include existing (legacy) systems into the analysis. For example, Sehmi and Schwegler (2006a) propose a pure top-down approach that describes how a business model can be implemented using software services. As their method has been incorporated in Microsoft's "Motion Methodology" (Merrifield and Tobey 2006), all details are not openly available. The starting points for the business service analysis vary widely. Jones (2006) and OASIS (2005) postulate a method that does not necessarily rely on any models or documentations, but on the collaborative analysis of the business of an organisation. IBM (2005) addresses business services as provisions of business components, whereas Flaxer and Nigam (2004) propose to analyse business entities to identify business components and business services subsequently. A bottom-up strategy for business services could not be identified.

Approaches addressing the derivation of software services postulate one of the three described delivery strategies. The top-down approach is supposed to derive a high quality SOA that is built on well-designed services and service compositions. However, depending on the size of the company and on the scope of the SOA initiative, a top-down strategy may consume significant resources, such as money and time, without showing an immediate outcome, since the upfront analysis has to be conducted before actually deriving services (Erl 2005; Gold-Bernstein and Ruh 2004).

Contrarily, a pure bottom-up strategy to deliver software services typically comprises activities that analyse existing legacy systems in order to define fine-grained services that can be linked to business processes and business requirements (Sneed 2006). Hereby, one can distinguish between two types of bottom-up analyses. Non-invasive legacy approaches encompass methods that do not change the structure of the legacy code (Nadhan 2004; Al Belushi and Baghdai 2007). They propose to build wrappers around the functionalities and components of the legacy system, so that they can be used in a service-oriented environment. Invasive legacy approaches aim at self-contained software services that encapsulate the functionalities provided by the legacy systems by restructuring the respective legacy code (Chen et al. 2005; Sneed 2006; Zhang, Liu and Yang 2005).

An interesting point is that most approaches postulate a meet-in-the middle strategy that takes into account business requirements as well as existing legacy systems to combine the advantages of both strategies. Thus, the advantages of a high quality SOA have to be weighed against reality constraints applied by the legacy systems. Arsanjani (2004) as well as Zimmermann, Krogh and Gee (2004) describe what an overarching approach could look like, but they fail to go into detail, as their approaches are proprietary.

Lifecycle coverage: Regarding the lifecycle coverage, it is obvious that the scopes of the methods vary widely. While certain methods specifically focus on the analysis phase (e.g. Klose, Knackstedt and Beverungen 2007), others also address the service design phase (e.g. Kohlmann and Alt 2007) or even address the complete lifecycle (e.g. Erl 2005, Papazoglou and van den Heuvel 2006). However, as a unified, standardised life cycle for services or SOA is not prevalent in literature, authors addressing the life cycle of a service propose such a life cycle in the course of their publication.

Degree of prescription: Some of the approaches do not provide any structured guideline or process to derive services. These approaches give general suggestions about what to do, but do not provide information on how it should be done (e.g. Arsanjani 2004). Thus, they can be used or must be used in a flexible manner, as detailed application steps are missing. Nonetheless, most of the analysed approaches provide some kind of procedural model to identify business services or software services. The steps or phases for the identification are very approach-specific. For example, Rahmani et al. (2006) propose an Model Driven Architecture (MDA) approach that focuses on the derivation of three

specific models (three phases) to identify services. Jones (2006), on the other hand, proposes four steps within his identification framework that can be used to identify services.

Availability and validity: There is a strong correlation between the documentation of the method within the respective publication and the application in practice. Typically, if the method has been applied in practice by conducting a case study, the case study will be described within the paper. If the method has not been applied in practice, typically just basic examples are presented.

Adoption of existing techniques / processes: Different methods relate to already existing concepts. Zimmermann, Krogdahl and Gee (2004) compare service-orientation to component- and object-orientation, as does Chang and Kim (2007), for example. Some methods specifically relate to UML, such as Stojanovic et al. (2004) to model services, which is widely applied in industry. Other methods utilise very specific languages or techniques that are tailored to their respective needs, e.g. Zhang, Liu and Yang (2005) or Quartel, Dijkman and van Sinderen (2004).

Regard to stakeholder: Stakeholders are addressed differently by the analysed methods. Some do not consider stakeholders at all (e.g. Arsanjani 2004), whereas others address stakeholders during the identification of services (e.g. Jones 2006, Sehmi and Schwegler 2006a, Kaabi et al. 2004 and Klose, Knackstedt and Beverungen 2007). However, the way the stakeholders are addressed varies amongst these methods. For example, Jones (2006) analyses the way in which external stakeholders interact with the services of an organisation, whereas Klose, Knackstedt and Beverungen (2007) analyse the stakeholder involvement in the service delivery process by examining the takeover and visibility potential of different process steps.

Service classification: Different authors propose different classification schemes based on the scope of their proposed approaches. Some approaches only provide guidelines to derive services in general, others distinguish between basic types of services and a few provide a classification scheme with descriptions of the objectives of each service. In the case of business services, no classification scheme based on the analysed approaches could be identified. For example, Arsanjani (2004) as well as Sehmi and Schwegler (2006a) propose decomposition approaches that can be used to identify services, but no classification for services is provided. If the SOA concept of software service is addressed by the approaches, different levels of details are observable. For example, Sneed (2006) and Zhang, Liu and Yang (2005) do not provide any classification and thus propose approaches that generally identify software services. Zimmermann, Krogdahl and Gee (2004) provide a rough differentiation between services, whereas Erl (2005), Klose, Knackstedt and Beverungen (2007), Kohlmann and Alt (2007) and SAP (2005) offer more detailed classification schemes. As the naming of the different services is not standardised, a proliferation of homonyms and synonyms can be encountered.

5 DISCUSSION

The number of publications that broach the issue of service engineering including service analysis has increased in recent years, but to the best of the authors' knowledge, a comprehensive analysis of the coverage and specific characteristics has not been sufficiently conducted yet, particularly not against the background of evaluating the fitness of extant methods in regard to requirements resulting from the trend towards an integrated application of the service paradigm in organisations on both the business and the technical level.

Our research has yielded a framework of criteria that was used to describe the characteristics of a significant sample of service analysis-related methods in detail. One benefit of this framework of criteria and the method comparison is that they support the process of method selection by an organisation, as they facilitate the definition of a set of desired method properties based on individual requirements and the subsequent evaluation of the methods against these desired characteristics. The second benefit of our results is that they are suitable to suggest ways how to address identified shortcomings. For example, based on our analysis we conjecture that there is currently a lack of methods that meet the requirement of a holistic service analysis method - the requirement that

motivated our research in the first place. A consolidated method should comprise business services as well as technical services. Thus, a comprehensive method needs to provide guidelines about how to identify business services that can subsequently be supported by software services. In the following, we discuss first suggestions how the results of our analysis can inform the design of such an improved method. Investigating the overlaps and complementary aspects of the analysed methods, one can apply the concepts behind method engineering (Odell 1996) to outline what a consolidated, comprehensive method may look like.

For example, we identified several commonalities and complementary aspects of the analysed methods (e.g. common starting points, different phases supported etc.). An earlier version of the method published by Jones (2006) has been used to serve as the foundation for OASIS (2005). These approaches can be complemented with the method proposed by Sehmi and Schwegler (2006a) and IBM (2005), thereby creating the foundation for the identification and analysis of business services. As processes are one of the underlying elements of a business service as defined by Sanz, Nayak and Becker (2006), methods proposing to use process models as a starting point for service analysis can be utilised. As our analysis pointed out, the majority of methods actually utilise process models for the analysis of services. For example, the methods proposed by Klose, Knackstedt and Beverungen (2007), Kim and Doh (2007) and Kohlmann and Alt (2007) could extend a detailed comprehensive method as the one developed by Erl (2005), as the different steps to derive services are complementary. Other methods use compatible starting points that could potentially be converted to be complementary to the process-driven approaches. For example, a use case corresponds to the activities and goals of an elementary business process and hence, the concepts behind the identification of services of use case-oriented approaches can be consolidated with the approaches focussing on process decomposition. Zimmermann, Krogdahl and Gee (2004) point out, however, that successful service analysis and design requires a focus on processes first, as it is necessary to analyse more than one system at a time to derive adequate services. The approach by Gold-Bernstein and Ruh (2004) can be adapted to business process models, as they propose that a table with events occurring during the operations of an organisation be used as the foundation for deriving services. These events can also be represented within a business process if they are brought in a sequential and logical order. Other approaches build upon one another and thus overlap to a certain extend.

Once the software services are identified, one needs to consider the actual application landscape that provides the needed functionality, as proposed by Erl (2005) and Inaganti and Behara (2007). The actual implementation might utilise methods for invasive or non-invasive code restructuring (Zhang, Liu and Yang 2005 and Nadhan 2004, respectively). Further research will need to show how a detailed consolidated method can be developed and how applicable such a method will be in practice.

6 CONCLUSION

In this paper, we have compiled a framework of comparison criteria that was used to provide an overview and analysis of 30 extant service analysis-related service engineering methods that can be used as a starting point for the journey into service-orientation. The comparison presented is the first step towards filling research gaps as proposed in recent research road maps of SOA and motivated in the introduction of this paper. The analysis supports the conjecture that there is currently a lack of a comprehensive service analysis method that comprises the identification and analysis of services on both the business and the technical level, thus ensuring a high level of business and IT alignment and agility. Our research has pointed out initial considerations for the consolidation of existing methods to achieve this goal through further research.

References

- Adamopoulos, D., Pavlou, G. and Papandreou, C. (2002). Advanced service creation using distributed object technology. *Communications Magazine, IEEE*, vol. 40, no. 3, 146-154.
- Al Belushi, W. and Baghdadi, Y. (2007). An approach to wrap legacy applications into web services. *ICSSSM '07*, 1-6.
- Allen, P. (2007). The service oriented process. http://www.cbdiforum.com/secure/interact/2007-02/service_oriented_process.php, accessed on 01.04.2008.
- Arsanjani, A. (2004). Service-oriented modeling and architecture. How to identify, specify, and realize services for your SOA. <http://www.ibm.com/developerworks/library/ws-soa-design1/>, accessed on 01.04.2009.
- Bell, M. (2008). *Service-oriented modeling. Service analysis, design and architecture*. Hoboken, NJ: John Wiley & Sons.
- Bieberstein, N., Bose, S., Walker, L. and Lynch, A. (2005). Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals. *IBM Systems Journal*, vol. 44, no. 4, pp. 691-708.
- Chang, S. H. and Kim, S. D. (2007). A systematic approach to service-oriented analysis and design. In *PROFES 2007*, 374-388.
- Chen, F., Li, S. and Chu, W. C.-C. (2005). Feature analysis for service-oriented reengineering. In *APSEC '05*, 201-208.
- Cherbakov, L., Galambos, G., Harishankar, R., Kalyana, S., and Rackham, G. (2005). Impact of service orientation at the business level. *IBM Systems Journal*, 44 (4), 653-668.
- Dumas, M., O'Sullivan, J., Heravizadeh, M., Edmond, D. and Hofstede, A. T. (2001). Towards a semantic framework for service description. In *IFIP TC2/WG2.6*, R. Meersman, K. Aberer, and T. S. Dillon, Eds., vol. 239. Hong Kong: Kluwer, B.V., 277-291.
- Erl, T. (2005). *Service-Oriented Architecture. Concepts, Technology, and Design*. 4th ed. Upper Saddle River, NJ.: Prentice Hall.
- Erradi, A., Kulkarni, N. N. and Maheshwari, P. (2007). Service design process for reusable services: Financial services case study. In *ICSOC '07*, 606-617.
- Flaxer, D. and Nigam, A. (2004). Realizing business components, business operations and business services. In *CEC-EAST '04*. Washington, DC, USA: IEEE Computer Society, 328-332.
- Genovese, Y. (2007). Evaluating IBM, Microsoft, Oracle and SAP Business Application SOA Strategies. Gartner Research, G00150421.
- Gold-Bernstein, B. and Ruh, W. (2004). *Enterprise Integration: The Essential Guide to Integration Solutions*. Redwood City, CA: Addison Wesley Longman Publishing Co., Inc.
- IBM (2005). Component business models. <http://www-935.ibm.com/services/us/imc/pdf/g510-6163-component-business-models.pdf>, accessed on 01.04.2009.
- Inaganti, S. and Behara, G. K. (2007). Service identification: BPM and SOA handshake. *BPTrends*, vol. 3, 1-12.
- Jones, S. (2006). *Enterprise SOA Adoption Strategies. Using SOA to Deliver IT to the Business*. C4Media.
- Kaabi, R. S., Souveyet, C. and Rolland, C. (2004). Eliciting service composition in a goal driven manner. In *ICSOC '04*. New York, NY, USA: ACM, 308-315.
- Kim, Y. and Doh, K.-G. (2007). The service modeling process based on use case refactoring. In *BIS '07*, 108-120.
- Klose, K., Knackstedt, R., and Beverungen, D. (2007). Identification of services - A stakeholder-based approach to SOA development and its application in the area of production planning. In *ECIS '07*, H. Oesterle, J. Schelp, and R. Winter, eds., St. Gallen, 1802-1814.
- Kohlmann, F. and Alt, R. (2007). Business-driven service modelling - A methodological approach from the finance industry. In *BPSC '07*, Leipzig, 180-193.

- Kontogiannis, K., Lewis, G. A., Smith, D. B., Litoiu, M., Muller, H., Schuster, S. and Stroulia, E. (2007). *The Landscape of Service-Oriented Systems: A Research Perspective*. SDSA '07, IEEE Computer Society, 1.
- Marks, E. A. and Bell, M., (2006). *Service-Oriented Architecture. A Planning and Implementation Guide for Business and Technology*. Hoboken, NJ: John Wiley & Sons.
- Merrifield, R. and Tobey, J. (2006). *Motion Lite: A Rapid Application of the Business Architecture Techniques Used by Microsoft Motion*. Microsoft. <http://msdn2.microsoft.com/en-us/library/bb736727.aspx>, accessed on 01.04.2009.
- Nadhan E. (2004). Seven steps to a service-oriented evolution. *Business Integration Journal*, vol. 1, 41-44.
- Nayak, N., Nigam, A., Sanz, J., Marston, D. and Flaxer, D. (2006). Concepts for service-oriented business thinking. In *SCC'06*. Washington, DC, USA: IEEE Computer Society, 357-364.
- OASIS (2005). SOA blueprints. <http://www.oasis-open.org/committees/download.php/15965/05-12-00000.001.doc>, accessed on 01.04.2009.
- OASIS (2006). Reference model for service oriented architecture 1.0. <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf>, accessed on 01.04.2009.
- Odell, J. J. (1996). *A primer to method engineering*. IFIP TC8, WG8.1/8.2, Chapman & Hall, Ltd., 1996, 1-7.
- Papazoglou, M. P. and van den Heuvel, W.-J. (2006). Service-oriented design and development methodology. *International Journal of Web Engineering and Technology (IJWET)*, 2 (4), 412-442.
- Papazoglou, M. P., Traverso, P., Dustdar, S. and Leymann, F. (2007). *Service-Oriented Computing: State of the Art and Research Challenges*. *COMPUTER*, 40, 38-45
- Quartel, D., Dijkman, R. and van Sinderen, M., (2004). Methodological support for service-oriented design with ISDL. In *ICSOC'04*. New York, NY, USA: ACM, 1-10.
- Rahmani, A. T., Rafe, V., Sedighian, S. and Abbaspour, A. (2006). An MDA based modeling and design of service oriented architecture. In *ICCS'06*, 578-585.
- Ramollari, E., Dranidis, D. and Simons, A. J. H. (2007). A survey of service oriented development methodologies. Paper presented at the 2nd European Young Researchers Workshop on Service Oriented Computing, Leicester, UK.
- Sanz, J., Nayak, N. and Becker, V. (2006). Business services as a modeling approach for smart business networks. IBM Research Division Almaden Research Center, Tech. Rep. RJ10381 (A0606-001).
- SAP (2005). *Enterprise service design guide*. http://www.sap.com/platform/netweaver/pdf/BWP_ES_Design_Guide.pdf, accessed on 01.04.2009.
- Sehmi, A. and Schwegler, B. (2006a). Service-oriented modeling for connected systems - part 1. *The Architecture Journal*, vol. 7, 33-41.
- Sehmi, A. and Schwegler, B. (2006b). Service-oriented modeling for connected systems - part 2. *The Architecture Journal*, vol. 8, 35-39.
- Sewing, J.-H., Rosemann, M. and Dumas, M. (2006). Process-oriented assessment of web services. *International Journal of E-Business Research*, 2 (1), 19-44.
- Sneed, H. M. (2006). Integrating legacy software into a service oriented architecture. In *CSMR'06*. Washington, DC, USA: IEEE Computer Society, 3-14.
- Stojanovic, Z., Dahanayake, A. and Sol, H. (2004). Modeling and design of service-oriented architecture. In *SMC'04*, vol. 5, 4147-4152.
- Sun (2006). SOA RQ methodology. A pragmatic approach. http://www.sun.com/products/soa/soa_methodology.pdf, accessed on 01.04.2009.
- W3C (2004). *Web services glossary*. <http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211>, accessed on 01.04.2009.
- Zhang, Z., Liu, R. and Yang, H. (2005). Service identification and packaging in service oriented reengineering. In *SEKE'05*, Taipei, Taiwan, 620-625.
- Zimmermann, O., Krogdahl, P. and Gee, C. (2004). Elements of service oriented analysis and design. An interdisciplinary modeling approach for SOA projects. <http://www-128.ibm.com/developerworks/webservices/library/ws-soad1>, accessed on 01.04.2009.