A Comparison of SOA Methodologies Analysis & Design Phases

Sandra SVANIDZAITĖ
Institute of Mathematics and Informatics, Vilnius University

Abstract. Service oriented computing is a new software engineering paradigm that represents a shift in software engineering and raises the abstraction level by grouping common business process functionality and exposing it as a service. SOA allows a rapid and low-cost application development through service composition. Existing widely used methodologies designed to support object-oriented development such as RUP or agile cannot be reused for SOA without any adaptation. As a consequence, new methodologies that address all the principles and patterns of SOA are required to ensure effective SOA application development. This paper aims to present a state-of-the-art of the most widely known SOA methodologies describing their solutions proposed for SOA analysis & design phases. The characteristics according to which these methodologies are compared are discussed. The results of comparison are provided.

Keywords. SOA, SOA analysis, SOA design, SOMA, SOAF

Introduction

Most SOA methodologies propose to divide SOA development lifecycle into six phases: Service-oriented analysis, Service-oriented design, Service development/construction, Service testing, Service deployment/transition, Service administration/management. The first two phases are the most important ones because the success of SOA development mainly depends on them. Technology and standards, such as BPM, BPEL, WSDL, EA, OOAD are important to develop SOAs, but it has been widely recognized that they are not sufficient on their own. Just by applying a Web service layer on top of legacy applications or components does not guarantee true SOA properties, such as business alignment, flexibility, loose coupling, and reusability. Instead, a systematic and comprehensive SOA analysis & design methodology is required [1]. A number of SOA methodologies such as IBM RUP/SOMA, SOAF, SOUP, methodology by Thomas Erl and methodology by Michael Papazoglou has been proposed to ensure successful SOA development. A number of SOA methodology surveys have already been performed but they treat them from a general point of view without providing any in-depth analysis of properties of these methodologies aiming at SOA analysis & design phases. This paper contributes to outlining the drawbacks and benefits of proposed SOA methodologies and focuses on SOA analysis & design phases by providing in-depth analysis and a comparison according to characteristics specified.

1. Characteristics of SOA Methodologies Analysis & Design Phases

In order to evaluate analysis & design phases in SOA methodologies we have defined characteristics that will be used to perform a comparison. The characteristics proposed for evaluation are as follows [1, 2, 3, 4, 5]:

SOA analysis & design strategy: Three strategies (top-down, bottom-up and meet-in-the-middle) exist in SOA development, each varying in the amount of up-front analysis of the business domain and the dependencies on legacy systems.

SOA analysis & design coverage: Service-oriented analysis and design phases of SOA methodologies that will be analyzed and compared can be divided into five main activities that are further refined into steps. These steps are used for evaluation of SOA analysis & design coverage.

Main activities of SOA analysis & design phases:

- **Target Organization's Business analysis**. The aim of this step is to identify: organization's objectives, business goals and KPIs for their accomplishment. Also used technology, applications and people skills, common business terms vocabulary, business rules, business actors and main business use cases are defined. The step results in the creation of "as-is" and "to-be" business models.
- **SOA project planning.** The aim of this step is to formulate the vision and the scope of SOA project, select SOA delivery strategy (create services from scratch, create services from existing software components, buy services from third party providers), create project plan and accomplish financial analysis.
- **Service Identification.** The aim of this step is to identificate candidate services. All functional and non-functional requirements for SOA development are gathered. Created "to-be" business model is decomposed into business domains. After that, service candidates, their initial specifications, communication and initial dependencies are defined. Existing applications are analyzed in order to find which software components can be reused in SOA development.
- **Service Analysis and Specification.** The aim of this step is to select which candidate services will be developed and to create detailed service specifications for development. Services are grouped by their functionality into: business entity, application and business process services. Business process specifications that will group the services are created.
- **Service Realization Decisions.** The aim of this step is to document service realization decisions, to allocate service components to layers and to accomplish technical feasibility exploration.

Degree of prescription: SOA methodologies vary from the most prescriptive ones to the less descriptive ones. The degree of prescription is evaluated depending on the number of parameters provided in process description. Available parameters are: phases, activities, steps and inputs, outputs for each step.

Adoption of existing techniques and notation: Most of SOA methodologies are based on techniques such as OOAD, CBM, BPM, EA and notations such as UML and BPMN, while the others do not address specific techniques and notations and let the user to decide what techniques and notations are appropriate in a concrete situation, making the methodology harder to understand and to use for inexperienced users.

2. Analysis of Existing SOA Methodologies

IBM RUP/SOMA [6] is an integrated methodology developed by IBM in a will to bring unique aspects of SOMA to RUP. However, because SOMA is a proprietary methodology of IBM, its full specification is not available.

Methodology consists of four phases: *business transformation analysis, identification, specification, and realization of services*. Talking about SOA analysis & design all these phases are of great importance. However IBM RUP/SOMA does not cover the deployment and administration of services. The first phase *Business Transformation Analysis* can be mapped to *Inception* phase from classical RUP methodology. This phase is an optional one and can be omitted if organization's full business analysis and transformation is not performed. It aims to describe current "as-is" organization business process, to understand problem areas and improvement potentials as well as any information on external issues such as competitors or trends in the market. *Business Transformation Analysis* comprises such activities as: assessment of target organization and its objectives, identification of business goals and KPIs, definition of common business vocabulary and business rules, definition of business actors and main use cases, analysis of business architecture. The second phase *Service Identification* can be mapped to *Elaboration* phase from classical RUP and aims to identificate candidate services. Service Identification comprises such activities as: *Domain Decomposition, Goal-Service Modeling and Existing Asset Analysis*. The third phase *Service Specification* can be mapped to *Elaboration* phase from classical RUP and focuses on the selection of candidate services that will be developed. Service Specification phase comprises such activities as: *Service Specification, Subsystem Analysis and Component Specification*. The fourth phase *Service Realization* can be mapped to *Construction* phase from classical RUP and is focused on completion of component design for component implementation. Service Realization comprises such activities as: *Documentation of Service Realization Decisions, Allocation of Service Components to Layers*.

Service Oriented Architecture Framework (SOAF) [7] methodology consists of five main phases: *information elicitation, service identification, service definition, service realization, roadmap and planning*. The aim of SOAF is to ease the service identification, definition and realization activities by combining a top-down modeling of an existing business process with a bottom-up analysis of existing applications. The first phase *Information Elicitation* aims to define the scope and constraints of existing business process and used technology. Current business "as-is" model is created and "to-be" business model is defined. Candidate services are identified that will automate "to-be" business model. Non-functional requirements (NFRs) and Business Level Agreements (BLAs) should be also defined, categorized and prioritized. Process-to-Application Mapping (PAM) is performed that examines existing software assets in order to discover SOA candidate application functionality. *Service Identification* phase aims to define an optimal set of services. *Service realization* phase aims to define transformation strategies that will be used for transition from the legacy application architecture to the future application architecture by reusing, developing and buying third party services. The *roadmap and planning* phase purposes a detailed planning of transformation and identifies business and technical risks.

Methodology by Papazoglou [1]. In the paper, Papazoglou et al provide a SOA development methodology that covers a full SOA lifecycle. It is partly based on well-

established development methodologies as RUP, Component-based Development and BPM [5]. The methodology is based on iterative and incremental process and comprises one preparatory - *Planning* and eight main phases: *Service Analysis, Service Design, Service Construction, Service Test, Service Provisioning, Service Deployment, Service Execution and Service Monitoring.* Talking about SOA analysis & design only the *Planning, Service Analysis* and *Service Design* phases are important. The *Planning* phase is a preparatory one. Activities in this phase include analysis of business needs and review of current technology landscape, financial analysis of the project and a creation of SOA development plan. The aim of *Service-oriented analysis* phase is to elicit the requirements for SOA application. Business analysts create an "as-is" business process model that allows stakeholders to understand a portfolio of available services and business processes. The phase results in creation of the "to-be" business process model that will be implemented in SOA solution. Analysis phase consists of four main activities: *process identification, process scoping, business gap analysis and process realization*. *Service Design phase* aims to transform business processes and services descriptions to well-documented service interfaces and service compositions. Design phase consists of two activities: *Specification of Services and Specification of Business Processes*.

Methodology by Thomas Erl [2], [3]. This methodology is a step by step guide through the two main phases: service-oriented analysis and design. *Service-oriented analysis* comprises three main steps: *define business requirements, identify existing automation systems and model candidate services* and can be divided in two main parts: the first part in which business requirements are defined and the second part in which service candidates are modeled. The first part of the phase includes reviewing business goals and objectives, analyzing potential changes to existing applications in a will to find which processes and application components can be used in a future SOA application development. Business analysts prepare "as-is" process model which states the current situation and allows stakeholders to understand which business processes are already in place and which has to be introduced and automated, which application components can be reused. Service-oriented analysis results in the preparation of "to-be" process model that an SOA application will implement. The second part of service-oriented analysis is a *service modeling* sub-process by which service candidates are identified and modeled. Service modeling sub-process results in the creation of such artifacts as: *conceptual service candidates, service capability candidates and service composition candidates*.

Service-oriented Unified Process [8] or SOUP is a hybrid software engineering methodology that is targeted at SOA projects. As the name suggests this methodology is primarily based on the Rational Unified Process. Its lifecycle consists of six phases: *Incept, Define, Design, Construct, Deploy* and *Support*. SOUP methodology can be used in two slightly different variations: one adopting RUP for initial SOA projects and other adopting a mix of RUP and XP for the maintenance of existing SOA applications. When talking about SOA analysis only the first three phases *Incept, Define* and *Design* of this methodology are important. The first *Incept* phase aims to understand the business needs for SOA development and how SOA fits within the organization. The objective of this phase is to decide whether SOA project is profitable by evaluating project scope and risks or not. Incept phase comprises such activities as*: Formulation of the vision and of the scope of the system, Definition of SOA strategy, Return-on-Investment (ROI) analysis accomplishment* and *Creation of Communication Plan*. The

second *Define* phase is the most critical phase in SOA project. It aims to define the requirements and develop use cases. The objectives of this phase are: 1) to fully understand business processes affected, 2) to collect, define and analyze functional and non-functional requirements by using a formal requirements-gathering and management process like RUP, 3) to design support and governance model which explains how organization will support SOA, 4) to prepare a realistic project plan, 5) to define a technical infrastructure that is required to support entire SOA. The third *Design* phase aims to translate use case realizations and SOA architecture into detailed design documents. The objectives of this phase are: 1) to create detailed design document and data base model that explain the structure of the services, 2) to structure development process by defining the technology, coding standards and etc.

3. Comparison of SOA Methodologies

Analyzed and described in 2 section SOA methodologies were compared using characteristics described in 1 section by outlining main differences, benefits and drawbacks. Detailed comparisons are not included in the paper due to the space limits. The comparison resulted in a number of insights:

- The most prescriptive SOA methodology is IBM RUP/SOMA which is a proprietary one and widely used in industrial projects. It supports *meet-in-the-middle* SOA analysis & design strategy, covers all SOA analysis & design activities. It also has the best degree of prescription, because it provides activities, steps, inputs and outputs description for each phase. It adopts such existing techniques and notation as: BPM, UML, BPEL, WSDL, WS-BPEL.
- A methodology by Thomas Erl does not provide detailed descriptions how to start the SOA project, how to perform organization‘s business analysis, how to formulate the vision and the scope of the project, but, it provides detailed service-oriented analysis & design phases descriptions meaning that it cannot be used from the start of the project but it can be used in conjunction with other methodology that provides detailed recommendations how to initiate SOA project. It supports *top-down* SOA analysis & design strategy, has a good degree of prescription and also adopts such existing techniques as: BPM, WSDL, WS-BPEL, WS-* specifications.
- SOUP methodology is still only in its first steps and is not mature enough to assure successful SOA development because it lacks prescription: phases, activities, artifacts, process workers and their roles are not defined clearly. It supports *meet-in-the-middle* SOA analysis & design strategy, but it does not cover some of the SOA analysis & design activities. SOUP methodology lacks adoption of existing notations such as UML and BPMN that are used in service-oriented analysis and design.
- SOAF methodology supports *meet-in-the-middle* SOA analysis & design strategy, but it does not cover some of the SOA analysis & design activities, lacks prescription and adoption of existing techniques and notations to assure successful SOA development.
- Methodology by Papazoglou supports *meet-in-the-middle* SOA analysis & design strategy, adopts such techniques and notations as: CBD, BPM, BPMN, WSDL, BPEL, UML. It provides detailed recommendations for Service

Design and Specification, but as a methodology for SOA analysis & design it lacks prescription. It does not refine activities in concrete steps and does not provide inputs and outputs for them.

4. Conclusions

The aim of this paper was to compare the most widely known and popular SOA development methodologies by providing an in-depth analysis of Service-oriented analysis and design phases. In this paper we analyzed and compared the following SOA methodologies: IBM RUP/SOMA, SOAF, methodology by Thomas Erl, methodology by Papazoglou and SOUP. The research showed that: analyzed SOA methodologies vary in a degree of prescription from the most prescriptive ones, to the less prescriptive ones letting the user to tailor and to adapt the methodology to concrete project's scope. In addition to this, most of analyzed SOA methodologies are built upon and incorporate existing and proven techniques, notations such as OOAD, CBD, BPM, WSDL, BPEL, UML, meaning that earlier used approaches are still applicable and new ones for SOA development are offered, but new method for organizing the process of SOA development is lacking. Most of analyzed SOA methodologies propose meet-in-the-middle strategy for Service-oriented analysis, meaning that most of SOA projects do not start in an empty place and most of them are targeted to change legacy systems. Service-oriented analysis and design phases in each methodology result in similar list of key deliverables, although each methodology offers a slight different but at some activities overlapping approach to achieve them.

In the conclusion, we can say that much is already done in this area, but there is still a lack of mature, descriptive, validated in proof-of-concept case studies, non-proprietary SOA methodology.

References

[1] M. P. Papazoglou and W.-J. van den Heuvel, Service-oriented design and development methodology, *International Journal of Web Engineering and Technology* **2**(4) (2006), 412-442.
[2] T. Erl, *Service-Oriented Architecture: Concepts, Technology and Design*. Prentice Hall PTR, 2005.
[3] T. Erl, *SOA Principles of Service Design.* Prentice Hall PTR, 2008.
[4] H. M Shirazi, N. Fareghzadeh, and A. Seyyedi, A combinational approach to service identification in SOA, *Journal of Applied Sciences Research* **5**(10) (2009), 1390-1397.
[5] E. Ramollari, D. Dranidis, and A. J. H. Simons, A survey of service oriented development methodologies. In: *Proceedings of the 2nd European Young Researchers Workshop on Service Oriented Computing*, Leicester, UK, June 2007.
[6] *Introduction to RUP*, IBM Corp. [cited April 2012]. Available from: http://www.michael-richardson.com/rup_classic/#core.base_rup/guidances/supportingmaterials/introduction_to_rup_36B63436.html.
[7] A. Erradi, S. Anand, and N. Kulkarni, SOAF: an architectural framework for service definition and realization. In: *IEEE International Conference on Services Computing (SCC 2006),* 18-22 September 2006, Chicago, Illinois, USA. IEEE Computer Society, 2006, 151-158.
[8] K. Mittal, *Service Oriented Unified Process*. [cited April 2012]. Available from: http://www.kunalmittal.com/html/soup.html.