An Agile Approach for Service-Oriented Architectures

Hamza Chehili[1], Mahmoud Boufaida[1] and Lionel Seinturier[2]
[1]*Laboratory LIRE, Mentouri University of Constantine, Constantine, Algeria*

[2]*LIFL-INRIA ADAM, University of Lille, 59655 Villeneuve d'Ascq, France*

Keywords: SOA, Agile, Design, SCA, BPMN.

Abstract: Today, service oriented architectures (SOA) has received much interest. the need of methodologies for the implementation of this architecture is very critical and crucial. In this position paper, we present an agile approach for the development of SOA with the respect of the principles of agile methods. This approach exploits the BPMN model to design incrementally business processes and the SCA model to describe the business functions of the system as an assembly of components. As implementation of this approach, we announce briefly a computer aided software engineering (CASE) framework.

1 INTRODUCTION

The change and the rapid evolution of needs have led companies to invest more on their information systems. The later ones are often criticized for their limitations in terms of reuse and evolution. Thus, the problems of incompatibility and redundancy at the application level have highlighted the need for a homogeneous enterprise-wide solution. Service-oriented architecture (SOA) has emerged as a solution that meets business expectations. It is considered as an approach to normalize the software architecture across the enterprise (Erl, 2005). The service is the key element of this paradigm. It is seen as an autonomous entity, independent of platforms, which can be described, published, discovered and loosely coupled (Papazoglou, 2008). Thus, SOA offers features that are necessary for the performance of the company's business processes.

Several methodologies have been proposed in the literature for the development of such architectures (Ervin, 2007). These methodologies have the objective of implementing a solution taking into account the business process level. However, they are costly to use at the expense of speed of development provided for the adoption of SOA. In the other hand, agile methods know today a remarkable success in software engineering. This success essentially amounts to their pragmatic approach. They are iterative and incremental allowing the rapid development of deliverable software. Moreover, these methods take into account the changing needs by involving the client throughout the development cycle (Manifesto, 2001). The main theme in agile methods is to promote and speed up responses to changing environments, requirements and meeting the deadlines (Rao, 2011).

In our work, we focus on the complementarity between the concepts of SOA and those coming from practices of agile methods in the field of software engineering. We propose an approach for developing an agile SOA by benefiting from the advantages of agile methods.

Our proposed approach exploits the BPMN model (Business Process Modelling Notation) to design incrementally the business processes of an enterprise (Owen, 2004). Next, the SCA model (Service Component Architecture) is used to describe the business functions of the system as an assembly of components (Beisiegel, 2007). They represent a development of sub-projects similar to those currently covered by agile methods.

This position paper is structured as follows. First, an overview of current approaches is highlighted. Next, the proposed approach is described. Finally, a short conclusion is given.

2 SOME RELATED WORK

The challenge of adopting an SOA has caused the enrichment of the literature from a variety of research that addresses various establishment plans

Chehili H., Boufaida M. and Seinturier L..
An Agile Approach for Service-Oriented Architectures.
DOI: 10.5220/0004128304680471
In *Proceedings of the 7th International Conference on Software Paradigm Trends* (ICSOFT-2012), pages 468-471
ISBN: 978-989-8565-19-8

of such architectures. Methodologically, some research work tried to meet the needs in this area by exploiting various models, techniques and processes. The following list is not exhaustive: (Kunal, 2006), (Papazoglou, 2006), (Zimmermann, 2004), (Butlet, 2009), (Seung, 2008), (Vamshi, 2009), (Emig, 2008), (Zou, 2006) and (Engels, 2008).

Let us notice that there is no methodology that has been developed for the implementation of an SOA satisfying the conditions of the agility. Moreover, the methodologies with a degree of agility are limited to an exploitation of agile methods at some of their phases winch makes the development heavy in detriment of, among others, the rapid response to change and speed of delivery.

Concerning the agile methods used in the software engineering domain, they don't cover the entire lifecycle of an SOA especially the business process analysis of a company. Moreover, these methods are more appropriate for small development projects. Nevertheless, an ideal approach will be the division of a large scale project in small projects more suited to the principles of agile methods.

In our approach, we intend to cover the life cycle of an SOA while adopting agile practices throughout the development process. Thus, at each phase, the problem is divided into sub-problems. We start by dividing the development of a SOA project on many sub-projects on having a goal for each one in order to develop their business process. Thus, each business process is described as an assembly of components. The potential development of each of these components is considered as a project in software engineering. The next section presents the different phases of our proposed approach.

3 DESCRIPTION OF AN AGILE APPROACH

The proposed approach is based on an iterative and incremental process that comprises six phases which are: analysis of enterprise business, elaboration of the SCA model for each business process, identification of the granularity of business components, development of component operations, assembling business components, and assembling business processes (see Figure 1).

This approach is described as agile because it respects the principles of agile methods specified in the manifesto of agile methods. In fact, it allows the development iterative and incremental of business processes, business components, and components operations. Thus, it takes into account the changing needs by involving the client throughout the different phases.

3.1 Analysis of Enterprise Business

The first phase of this approach is the incremental modeling of all business processes of the enterprise using the BPMN model. This allows us to have a high level abstraction of the interactions leading services to perform business processes.

3.2 Elaboration of the SCA Model for Each Business Process

This phase is initiated after a business process is modeled (we do not wait for the achievement of the analysis of enterprise business phase). For each business process, it defines a component architecture described using the SCA model. The mapping from BPMN to SCA is based on rules defined in (Dahman, 2010). These rules allow an automatic

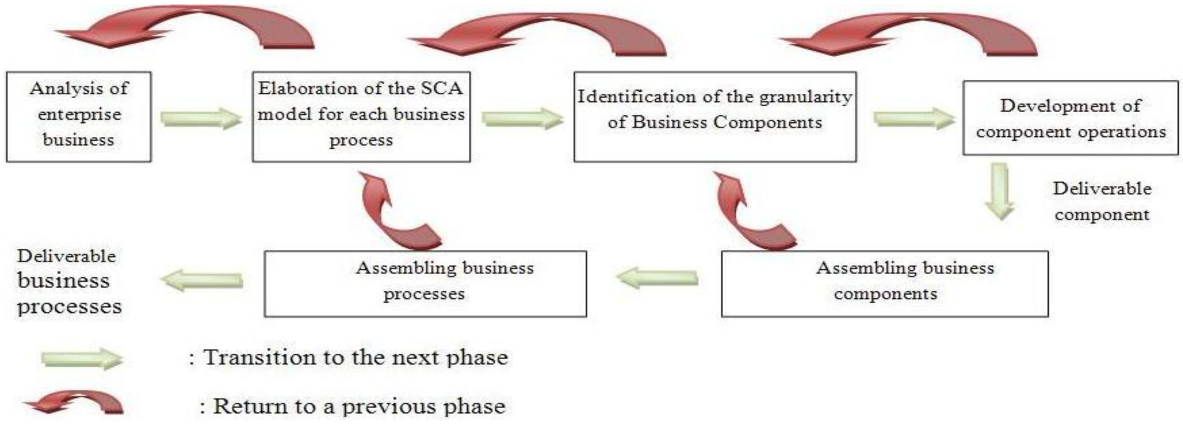
Analysis of enterprise business
Elaboration of the SCA model for each business process
Identification of the granularity of Business Components
Development of component operations
Deliverable component
Assembling business components
Assembling business processes
Deliverable business processes
: Transition to the next phase
: Return to a previous phase

Figure 1: Description of the approach.

mapping of business process activities described in the BPMN model to interactions description of a set of components in the SCA model. These components represent the high-level business functionalities of the enterprise. Thus, they can be assembled from components of a finer granularity having features or specific services that can be invoked by several business components.

3.3 Identification of the Granularity of Business Components

This phase takes as input the business components constituting the SCA architecture developed in the previous phase. The outputs of this phase are the descriptions of the components to be developed and the granularity of each business component. The process of this phase consists of the following steps:

Step 'Search': it takes as input the descriptions of the business components constituting the SCA architecture one by one and performs a search in descriptions of available components. It gives as output the list of components found realizing some of the requested operations.

Step 'Refinement': it takes as input descriptions of the components found in the search and description of the requested business component. It refines the latter based on fine-grained components found in the form of an assembly of components.

Research and refinement are repeated in the form of iterations until the level of refinement is optimal. Reconfigurations of the assembly of components are possible in each iteration according to the components found and the reached level of granularity.

At the end of this process, the component descriptions to be developed are passed to the development phase. The description of the assembly of the business component will be maintained in the platform. This description includes the assembly of the fine-grained components forming the business component in question (available components and those developed).

3.4 Development of Component Operations

This phase takes as input the descriptions of the components to be developed. The output of this phase are the components developed, tested and approved by the customer. The approach in this phase comes at a lower level of abstraction (process, components and operations now). Indeed, the process of this phase is inspired from the workshops of traditional software engineering, specifically, those adopted by agile methods.

The development process follows an iterative and incremental cycle with client involvement in the identification of tasks and tests. The process of this phase consists of the following steps:

Step 'Identification of tasks': it receives as input the description of a component to develop in the form of operations (identified in the previous phase). It allows the development team to identify the tasks to be developed for each operation. The outputs of this step are the tasks of each operation assigned to the developers with the tests to be performed.

Step 'development iteration': it takes as input the tasks to be developed. All developers receive the tasks allocated to them. In addition, any task completed must be passed directly to the test.

Step 'Unit testing': it receives the tasks and tests and allows one to make the planned tests. If there are anomalies, the task is returned to the development process.

Step 'Integration testing': it receives as input tasks that have passed the unit test successfully. It allows one to perform tests on the integration of operations and the entire component. If the integration fails, or the customer is not satisfied and request changes, the component is returned according to the change. This may be at one of the following levels:

- Step 'Development Iteration' if the change concerns a task.
- Step 'Identification of tasks' if the change concerns an operation.

In the case where the integration test passes, the component is deployed in the platform.

3.5 Assembling Business Components

This phase takes as input, for each business component, all components developed and deployed, and the components found in Phase 3. The output of this phase is the business component deployed. The process of this phase consists of the following steps: Step ‘assembling components’: it configures the business component from available components and those developed.

Step ' testing business component integration’: it takes as input the configuration of the assembly. It allows the integration testing of components involved in the assembly. If the test fails or the client is not satisfied and requests changes, the configuration will be referred to the assembly step or even to Phase ‘Identification of the granularity of Business Components' (Phase3) if the changes require the involvement of other components.

3.6 Assembling Business Processes

This phase allows the assembling and the configuring of a business process in which all the business components are configured tested and deployed. It takes as input these business components and return the business process deployed. The process of this phase consists of the following steps:

Step ‘assembling business components': It allows to configure the business process resulting from the business components from the previous phase. Step 'Testing business processes integration': It takes as input the configuration of the assembly. It lets do tests the integration of business components involved in the assembly. If the test fails or the client is not satisfied wanting change, the configuration is returned to the assembly step or even at the design phase of the SCA model for each business process (Phase 2) whether the changes give rise to the involvement of other business components.

4 CONCLUSIONS

In this position paper, we have presented an agile approach for the establishment of SOA. This approach covers the high level of abstraction of such architecture. It begins with the analysis of business on the scale of the enterprise. Thus, the successive division of the project, the customer involvement, and the acceptance test parts of the architecture can meet the terms of the manifesto of agile methods.

Currently, we are focusing on the definition of the description of components and their assembly in order to implement the approach. This can be achieved by identifying the operations of each component in relation to the processes described in the phases of the approach. Once the components are described, we will move to their development and deployment.

REFERENCES

Erl T, «Service-Oriented Architecture: Concepts Technology and Design». *Prentice Hall*. 2005.

M. P. Papazoglou et al., «Service-oriented computing: a research roadmap», *International Journal of Cooperative Information Systems*. Vol. 17, No. 2. P. 223–255. 2008.

R. Ervin et al., «A Survey of Service Oriented Development Methodologies». *The 2nd European Researchers Workshop on Service Oriented Computing*. 2007.

Agile Manifesto, Manifesto for Agile Software Development, 2001. Disponible en ligne: http://agile manifesto.org/.

K. N. Rao, K. Naidu, P. Chakka “A Study of the Agile software Development Methods, Aplicability and Implications in Industry”, *International Journal of Software Engineering and its Applications*, Vol. 6, No. 2, P. 35-45, 2011.

M. Owen, J. Raj “BPMN and Business Process Management: An Introduction to the New Business Process Modelling Standard”, *Business Process Trends*, March 2004.

Beisiegel, M. et al., “Service Component Architecture” 2007, www.osoa.org.

M. Kunal: «Service Oriented Unified Process (SOUP)». 2006.

G. Engels et al., «A method for engineering a true Service-oriented architecture». 2008.

M. P. Papazoglou et al., «Service-Oriented Design and Development Methodology». *International Journal of Web Engineering and Technology*. 2006.

W. S. Seung, K. K. Haeng, «A Framework for SOA-Based Application on Agile of Small and Medium Enterprise», *Computer and Information Science*, SCI 131, P. 107–120, 2008.

K. G. Vamshi, P. C. Andrew, «Implementing a SOA using Scrum: Achieving improved productivity using services trees», *Proceeding of the Southern Association for Information Systèms Conference*, Charleston, SC, USA, 2009.

O. Zimmermann et al., «Elements of Service-Oriented Analysis and Design», *IBM Corporation*. 2004.

J. Butlet, «An introduction to SoaML and its Place within the SAE Meta Model», *CDBI Journal*, Février 2009.

C. Emig and al., «Model-Driven Development of SOA Services». 2008.

K. Dahman, F. Charoy, C. Godart “Generation of Compenent Based Architecture from Business Processes: Model Driven Engineering for SOA», *Eight IEEE European Conference on Web Service*, P. 155-162, 2010.

Z. Zou, Z. Duan, «Building Business Process or Assembling Service Components: Reuse Services with BPEL4WS and SCA», *European Conference on Web Service*, 2006.