TELKOMNIKA, Vol. 13, No. 4, December 2015, pp. 1466~1477
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v13i4.2388

Service Engineering Based on Service Oriented Architecture Methodology

Suhardi[1], Robin Doss[2], Purnomo Yustianto[3]
[1]School of Electrical Engineering and Informatics, Institute of Technology Bandung, Indonesia, 40132
[2]School Information Technology, Deakin University, Melbourne, Australia, 3125
*Corresponding author, e-mail: suhardi@stei.itb.ac.id[1], pyustian@deakin.edu.au[3]

Abstract

Service Engineering (SE) and Service Oriented Architecture (SOA) originally reside on different plateaus of discipline. SE is defined as a set of activity introducing a new business service, while SOA is a technical approach to redefine an enterprise business processes as a set of IT enabled services. This paper proposes a SOA embedded SE framework as a comprehensive approach in re-defining business service and its IT implementation. After an introduction, a review of existing SE frameworks and SOA methodologies is presented in the paper. Afterward, a complete SE framework is proposed with several results on early case studies. A survey results are then presented to prove the usability and benefit of the proposed framework. The framework is designed and proposed to help practitioners and researchers to conduct service engineering by employing principles and methodology offered by SOA approach.

Keywords: *Service Engineering, Framework, SOA, Business Model Canvas, Service Blueprinting*

1. Introduction

The global shift towards a service-based economy has brought importance to the concept of Service Engineering (SE). Enormous changes in globalized business environment in recent years have made service innovation a critical priority for all business managers. Business entities compete to provide the best value for customers by taking advantage of the competitive business environment. SE methods have the potential to help the process of service innovation to deliver intended and designed value of a service to the customer.

Information Technology (IT) has been an important component in the recent growth of the service industry. It also takes a central role in innovating new services. Therefore, as a practical and an academic field, SE requires complementary analysis from IT point-of-view. As a result of this trend, it is imperative that the field should be advanced and enlarged with participants, not only from business and management field, but should also involve practitioners and researchers with IT expertise.

The term '*service engineering*' (SE) was first mentioned during the mid-nineties as a technical discipline in systematic development and design of services using appropriate models, methods, and tools [1] SE encompasses broad range of activities starting from business strategic layer to process design and its detailed implementation in technical layer. The introduction Service Science, Management and Engineering (SSME) as a discipline reiterates the emphasize on an interdisciplinary approach, combining management and engineering theories, including from IT discipline, in the objective of improving the competitiveness in the organization based on growing services needs in the business environment [2]. To handle such complex undertaking, a frameworks and methods to guide the process is required [3], which can cover through the technical aspect in the IT layer.

An emerging approach from IT discipline, Service Oriented Architecture (SOA), was introduced with the capability of bridging the business process (BP) requirements with the IT development processes implemented as services component [4]. Complementing SE with SOA approach, therefore, should be an ideal approach for business service development to answer the competitiveness challenge and the demand for agility in the service industry. While a methodical framework to guide a service engineering process is required, a formalized and practical framework which can guide an application of the SE with SOA approach is still largely untouched. Some early propositions have already been made [5] [6], but these works did not

Received August 3, 2015; Revised September 8, 2015; Accepted October 2, 2015

delved into the details of SOA methodology, and did not provided case studies as a proof of concept.

This paper proposes a formalized SE framework, specifically embedded with SOA approach. Three characteristics are considered in formulating the proposed framework: simplicity, conformity and uniformity. The framework is designed to simplify the process in terms of easiness of adoption and implementation, and at the same time it also ensures the completeness and appropriateness of the solution provided for given situation in an organization. The existence of the framework is also designed to create the uniformity of service engineering process across multiple implementations. The proposition of the framework is presented in this paper with case studies and survey results on its implementation.

2. Service Engineering Frameworks

The definition of service engineering with regard to e-service scope is specified as "*an approach that provides a discipline for using models and techniques to guide the understanding, structure, design, implementation, deployment, documentation, operation, maintenance and modification of e-services*" [7]. Within this scope, the objective of SE framework is to provide the services in IT layer, specifically in the form of web services which will allow a high degree of automated interaction among services. Therefore, it is highly desirable to redefine the architecture of the enterprise using approaches from Service Oriented Architecture (SOA).

From SE point of view, two frameworks for designing (new) services were selected to be a comparative base to propose the new framework. The first framework was taken from classic NSD concept, proposed by Lin and Hsieh [8], and appropriately named as New Service Development (NSD) framework. Based on ITIL, the second framework was introduced as Service System Development Process (SSDP) [9].

Both frameworks propose the combination use of various management analysis tools such as: Feasibility Study, Socio Techno-Economical Analysis, Environmental Scanning, Trend Analysis, BCG matrix, and Quality Function Deployment (QFD). A comparison can be made between the two focusing on the approach in designing a service, specifically in modeling the service.

The first framework, the NSD framework, defines the Service Design as two distinct stages: (1) service modeling and (2) service implementation. The service modeling itself is composed from four components: (1) product model, (2) process model, (3) resource model, and (4) marketing concept. Among others, the UML notation is employed as a tool for product model. Service blueprinting technique is suggested as a process model tool. The main proposed tools of service design in NSD framework is the QFD matrix.QFD is suggested as a tool to visualize the customer needs during the service development process. A modified QFD is also suggested as a service-planning matrix to display customer requirements, technical measures, target values, and competitive analyses in the form of House of Quality (HOQ).

The second framework (SSDP) defines the service design and development as analyzing service requirements into the identification of (1) service entities functions, (2) service interfaces, (3) service interoperability, and (4) service level agreements. While not explicitly mentioning SOA approach, this definition basically embodies the SOA design approach. In modeling the service, SSDP utilizes a service meta-model [10] to help stakeholders conceptualize the service value chain.

The two frameworks are inherently an iterating process of continuous improvement, and thus the result of each stage is a feedback for the next iteration. Also, tests could be conducted in the end of each step, ensuring conformity of the step results with the input specifications stated in the beginning of the step, before deciding to continue to the next step.

A combination and generalization between the two has also been proposed, defining the General Service Engineering Framework [11]. It is composed from four groups of activities: (1) Identification, (2) Design, (3) Development, and (4) Operation. This paper is a follow-up on this concept, by providing elaboration details into the proposed framework and by focusing the highlight on the role of SOA methodology in SE Framework.

3. SOA Methodologies

SOA is defined as software architecture based on loosely coupled software services which integrated into a distributed computing system, by means of service-oriented programming [12]. In SOA perspective, services are the building blocks of an enterprise. An enterprise is defined by its pool of services, both available internally and publicly, and the interaction pattern between the services. The service interaction is not bound internally in an enterprise, but the business landscape or the business environment is also comprised of interacting services among various organizations. Thus, the well-being of an enterprise and the business environment is determined by the implementation quality of these services.

Since its inception in the mid 2000's, the SOA approach has experienced growth in maturity and rate of adoption. While several main concepts were indeed converged, SOA contributors are still unable to provide on a singular cohesive view on the SOA adoption. Various methodologies are offered in the form of step-by-step guidance in designing the implementation of SOA. From compiling works on SOA methodologies [13] [14] [15] [16] [17], we can collect at least 15 different methodologies. Two most popular SOA methodologies are (1) IBM's SOMA [18] and (2) Thomas Erl's SOA [19], often labeled as Mainstream SOA methodology (MSOAM) [15].

IBM's SOMA is a highly recommended SOA method due to its comprehensiveness and its vast industrial adoption [20] [14] [17]. It covers a complete cycle of service engineering, from business side to the technical implementation. The method consists of six main stages: (1) business transformation analysis, (2) identification, (3) specification, (4) realization, (5) implementation, and (6) deployment-monitoring. Each stage in-turn consists of three to four sub stages.

Table 1. Stages of SOMA Methodology [1]

1. Business Modeling Transformation	4.3. Component Specification
2. Solution Management	4.4.Refactor & Rationalize Services
3. Identification	**5. Realization**
3.1. Goal-Service Modeling	5.1. Realization Decisions
3.2. Domain Decomposition	5.2. Technical Feasibility Exploration
3.2.1 Functional Area Analysis	5.3. Componant Layering
3.2.2 Process Decomposition	**6. Implementation**
3.2.3 Variation-Oriented Analysis	6.1. Service construction
3.3. Existing Asset Analysis	6.2. Unit Test
4. Specification	6.3. System & Integrat Test
4.1. Service Specification	**7. Deployment**
4.1.1 Service Flow Specification	7.1. Deploy Services
4.1.2 Message & Event Specification	7.2. User Acceptance Test
4.2. Subsystem Analysis	7.3.Monitor & Manage

A variation of SOMA proposes the use of Component Business Model (CBM), as an analytical tool in the early stage of the methodology, during stage 1 to 3. CBM is basically a matrix of enterprise's business competencies against accountability level, i.e. directing, controlling and executing [21]. The resulting elements of the matrix are defined as business components. Each business component is defined to have attributes of: (1) purpose, (2) activity, (3) resource, (4) governance, and (5) services offered. The matrix can be used to determine the importance of each business component, and ultimately to define new service offering.

A complete SOMA methodology, therefore, can be considered as SE framework in itself. Yet, adopting SOMA independently has been made difficult due to the limited availability of detailed SOMA reference material. Combining SOMA with SE framework therefore will necessitate the removal of several part of the methodology due to the process redundancy. With the help from a detailed reference, SOMA methodology can still be useful in building SE framework, especially in the specification stage and onward.

Another well-known SOA methodology is the one devised by Thomas Erl. In his book [19-6] he elaborated a definitive reference for SOA implementation, MSOAM. The methodology consists of seven stages of activities: (1) Ontology Definition, (2) Business Model Alignment, (3) Service Oriented Analysis, (4) Service Oriented Design, (5) Service Development, (6) Service Testing, and (7) Service Deployment. The emphasis of MSOAM elaboration is during analysis and design activities. This is evident by the number of sub-stages defined for the two stages.

While providing detailed prescription on analysis and design activities, MSOAM does not provide elaborate guidance for activities during both preliminary business analysis and post-design. These lacks make it somewhat appropriate to be combined with SE framework. MSOAM methodology strength in analysis and design stages is quite useful to be adopted, but care should be taken to ensure that no activity redundancy is performed. Additional activities should also be defined in the SE framework to fill the gaps for pre-design and post-design activities in original MSOAM.

Table 2. Stages of MSOAM Methodology [6]

1. Define Relevant Ontology	4.2. Design Entity-Centric Business Serv.
2. Align Relevant Business Models	4.3 Design Application Services
3. Perform Service Oriented Analysis	4.3.1. Review Existing Services.
3.1. Define Business Requirement	4.3.2. Confirm Context
3.2. Identify Automation System	4.3.3. Derive Initial Interface
3.3.Model Candidate Services	4.3.4. Apply Service-Orientation.
3.3.1. Decompose Business Process	4.3.5. Standardize Service Interface
3.3.2. Identify Operation Candidate	4.3.6. Add Speculative Features
3.3.3.Abstract Orchestration Logic	4.5 Design Serv. Oriented Business Process
3.3.4. Create Service Candidates	4.4. Design Task-Centric Business Serv.
3.3.5. Refine & Apply Service Orientation	4.4.1. Define workflow logic.
3.3.6.Identify Service Compositions	4.4.2. Derive initial interface.
3.3.7.Revise Operation Grouping	4.4.3. Apply Service Orientation.
3.3.8. Analyze Processing Requirements	4.4.4. Standardize Service Interface.
3.3.9.Identify Application Service Operations	4.4.5. Identify Required Processing.
3.3.10. Create Appl. Service Candidates	4.5 Design Serv. Oriented Business Process
3.3.11.Revise Service Compositions	4.5.1. Map Out Interaction Scenarios
3.3.12. Revise Operation Grouping	4.5.2. Design Process Service Interface
4. Perform Service Oriented Design	4.5.3. Formalize Partner Serv. Conversation
4.1. Compose SOA	4.5.4.Define Process Logic.
4.1.1. Chose Service Layer	4.4.5. Align Interaction Scenarios &Refine
4.1.2. Position Core Standards	5. Develop Services
4.1.3. Choose SOA Extensions	6. Develop Test Service Operations
	7. Deploy Services

The heart of SOA design activity is the specification activity, employing modeling notation. A review article [22] reveals six available notations for SOA modeling: (1) SOA-RM, Reference Model for SOA, (2) SOA-RFA SOA Reference Architecture Foundation, (3) SOA Ontology, (4) SOMF, Service-Oriented Modeling Framework, (5) PIM4SOA, Platform-Independent Model for SOA, and (6) SOAML, SOA Modeling Language. For a simple SOA implementation, a basic UML or a Thomas Erl's notation is sufficient, but for a complex SOA implementation, SOAML, as an extension of UML, is suggested due to its referential availability and vast adoption in several SOA methodologies.

4. The Proposed Framework

The proposed framework is drawn from generalized SE approach, described in second part of this article, with specific use of business analysis tools, and combined with SOA methodology, in order to achieve a more practical approach in service engineering. The proposed framework consists of four stages: (1) identification stage, (2) design stage, (3) development stage and (4) deployment stage.

The first stage, the identification stage, is the business side analysis in which new (electronic) services potential are identified and proposed with analysis of existing condition of the whole organization entity. The design stage, as the second stage, consists of both business design and technical design process. The services are designed or redesigned in this stage with regard to the existing condition of the organization. Implementing the framework for the first time in an organization is also an opportunity to adopt SOA approach in the organization of IT system. The decision of employing SOA in top-down or bottom-up manners is made during this stage. The third stage, the development stage, is a software engineering process conforming to SOA approach. The last stage, the deployment stage, consists of migrating to production environment, along with applying monitoring measures and setting up SOA Governance to ensure the conformity of systems for future improvements.

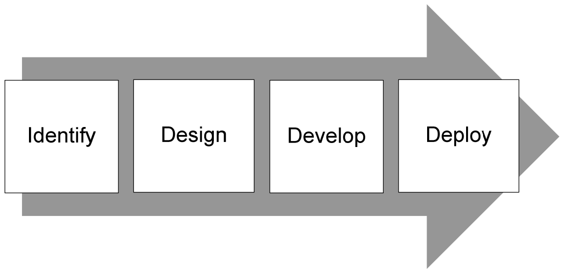

Figure 1. Proposed Service Engineering Framework

4.1. Identification Stage

In the identification stage, new service(s) potential for a customer is identified and defined. The underlying motive for innovating new service is to improve customer value. In that sense, the service identification stage must consist of steps to identify services with high customer value. Business Model Canvas (BMC) [23] is used in this stage as an analysis and modeling tool of the firm to identify components with strong potential to improve the value proposition to the customers. Nine blocks of BMC are the analytical bases for improving the firm value. The potential of improvement might arise from customer segment, revenue stream, cost structure, value proposition, or in key activities.

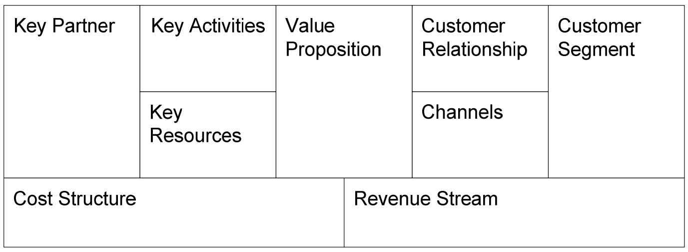

Figure 2. Business Model Canvas Template

The identification stage should start with "as-is BMC", but the real result of the BMC analysis is the newly proposed BMC: the "to-be BMC". This new BMC represents the proposition of service innovation. As an example, the decision to improve the Key Activities block might require a creation of service innovation in supporting activities. Therefore, a new support service should be designed and implemented to increase the effectiveness of the key activity process. This type of service can be categorized as "service-as-support". In another case, the decision of improvement might reside in the Value Proposition block. In this case a new service should be proposed for the customers, in the form of direct customers facing services. This type of service can be categorized as “service-as-product”.

Another common tool to identify the service innovation is the questionnaire method. The method is regularly used during a market research activity. The use of questionnaire is part of true requirement investigation, as the requirement is defined based on actual interaction with prospective customers [5]. An analysis from the questionnaire data might instigate a new service innovation. For example, the questionnaire result might suggest an improvement in the firm key activities. Therefore a service innovation can be introduced from “service-as-support” category to enhance the actual key activities.

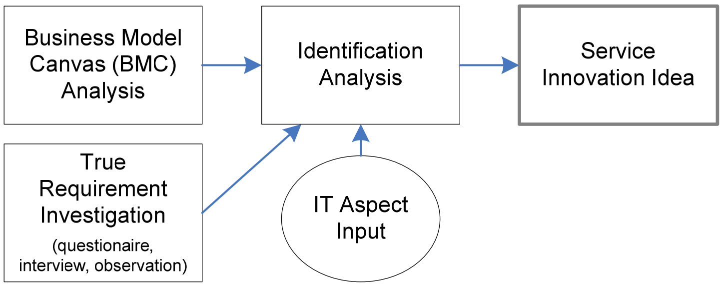

Figure 3. Stages of Identification Stage

Another common form of true requirement investigation is observation. Observation is a method to capture the information based on the real-world situation. Observation process can also include questionnaire, or interview activities. During the observation the researcher must directly visits or experiences the site of actual business process and customer transaction performed.

A service innovation in these days frequently leads to the implementation of new Information Technology (IT) component [24]. An input from IT assessment is therefore will be valuable in the identification stage. The latest trend in IT, such as cloud computing, could serve as an important element in the service innovation.

The final result of an identification stage is presented in the form of "to-be BMC" complemented with summary of the service innovation idea. Both components should be formalized as a service identification document.

4.2. Design Stage

The design stage of the proposed framework is divided into two sub-stages: (1) service process design and (2) SOA design. The goal of the first sub-stage, the service process design, is to create the design of the service defined from analysis in the identification stage. The second sub-stage, SOA design, is the sub-stage to elaborate the design using SOA approach and methodology.

During service process design, the service blueprinting technique [25] is proposed to represent how the service should be delivered to the customer. Service blueprinting technique is normally used to describe existing service or to represent the service innovation. The technique defines five layers for service interaction: (1) physical evidence, (2) customer action, (3) on-stage contact employee, (3) back-stage contact employee and (5) support process. In addition to service blueprinting technique, Business process diagram (BPD), with its Business Process Modeling Notation (BPMN) as the tool, can be used to elaborate the process of service innovation.

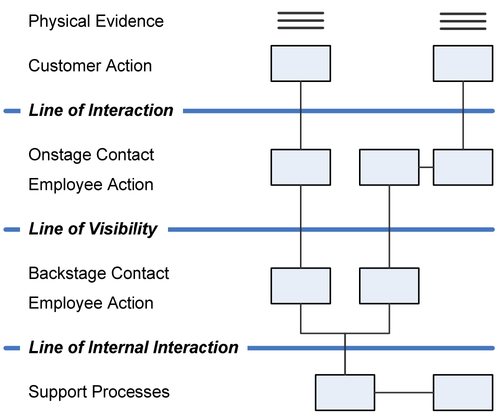

Figure 3. Service Blueprint Template

The intention of the service design sub-stage is to have an overview of the service. The result from this process will be used as a reference to determine the service operation in the SOA implementation sub-stage.

The second sub-stage in design stage is the SOA implementation sub-stage. In this sub-stage, an SOA methodology will guide the creation of service design in IT terms. An SOA methodology commonly involves an identification activity followed by design activity. The purpose of identification is to identify the candidate services, while the purpose of the design stage is to define the services specification, such as service contract and choreography. In the proposed framework, some part of SOA service identification has already been performed during the previous step, in the form of to-be BMC, service blueprint and BPMN. These artifacts are the starting point for Service identification in the SOA methodology. For example, the case of adopting Thomas Erl's MSOAM methodology, the first two steps of it, Ontology Definition and Business Model Alignment, can be simplified by employing the results of BMC analysis. The same idea is also applicable to steps 3.1 (Business Requirement Definition) and 3.3.1 (Decompose Business Process) in which the previous result from Business Process Diagram and Service Blueprinting can be used.

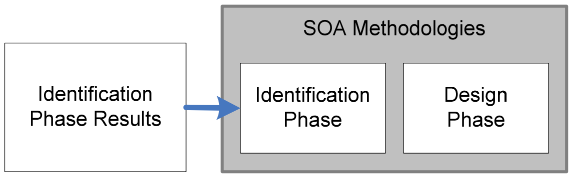

Figure 4. SOA Methodologies in Design Stage

4.3. Development Stage

For a full-fledged SOA implementation, several commercial platforms are available to support service-oriented computing approach of SOA, such as IBM WebSphere, SAP NetWeaver, BEA Weblogic, and Oracle SOA Suite. An open source alternative also exists as JBoss Enterprise SOA Platform.

For our goal with the proposed framework in a simple deployment, Microsoft's Windows Communication Foundation (WCF) technology is recommended due to its simple development requirement compared to other platform. Selecting WCF will help defining activities in stage 4.1.2 (Position Core Standards) and 4.1.3 (Choose SOA Extensions) of MSOAM. Using WCF as the core standard and SOA technology also simplify stage 4.2 (Design entity-centric business services) and 4.3 (Design application service) due to the automation of WDSL (Web Service Definition Language) generation in WCF.

During and in the end of development stage, several tests for the system must be conducted. Due to the nature of SOA implementation as distributed-messaging system, performing unit test and system test for large scale or enterprise-wide SOA system can be a challenging undertake. Proposition for testing SOA (web) services involves generating test requirements by using Service Message Flow Diagram (SMFD) calculating every connections variation, and the use of test tools for web service, such as SoapUI to WebInject [26].

Before moving to deployment stage, a User Acceptance Test (UAT) must be performed to ensure conformity between the result of development and the result from analysis and design stage, also to verify that the service system meet the user requirement.

4.4. Deployment Stage

A successful testing result can be deployed and further monitored and enhanced for the system. Otherwise, the process must be repeated by analyzing and determining steps to be reiterated, in analysis, design or prototyping stage, and repeating the stage to fix the deviation. Entering the deployment stage, any measures for management change of the organization is planned and executed to ensure a smooth transition from the existing system. Monitoring activities should also be conducted to observe the real-world behavior and performance of the newly installed service system and its infrastructure as a means for continuously improving the service. For enlarging the implementation of SOA approach, SOA governance should also be designed and put into place.

The following table summarizes the concept and tools employed along with artifacts produced during each stage of the framework.

Table 3. Proposed SE Frameworks Tools and Artifacts

Stage	Tools	Artifact
1. Identify	True requirement	Questionnaire result Observation anaysis
	BMC	As-Is BMC Service Innovation To-be BMC Business Service Catalog
2. Design	Service Blueprinting BPMN SOA Method (e.q. MSOAM/SOMA)	Service Blueprint BPD UML / SOAML IT Service Catalog
3. Develop	SOA Platform (e.q. WCF)	IT Service Implementation Unit and system test results UAT result
4. Deploy	SOA Governance	SOA Policy &Standards Monitoring measures

5. Case Studies

As an attempt to formalize the proposed Service Engineering Framework, the framework was incrementally built by using components of the proposed framework and undertaking the following case studies. The details are presented below.

5.1. First Case Study

The first case study was performed before a definitive formal framework was devised. The locus of the study was a State Palm Plantation Firm. The goal was to provide an integrated Enterprise Resource Planning (ERP) based on SOA approach. As a service engineering work, the study employed SOMA as its SOA methodology. From the three identification techniques in SOMA: domain decomposition, goal service modeling and existing system analyst, only the first two are performed, due to the minimal IT resources available in the firm. In modeling the services the work enhances the use of UML with SOAML in the diagrammatical forms of Service Specification, Service Interface and Service Realization. Finally, a prototype of the service design was produced by using WCF technology. This case study served as a basis to assess the context of a SOA Methodology within a Service Engineering process.

5.2. Second Case Study

The second case study was a service engineering study on the process of land usage permit in the National Land Authority [27]. The study mainly employed SOMA as the basic methodology, but only performed Domain Decomposition and Goal Service Modeling during the identification stage. Service blueprinting was used to describe the business process of the service. Several modeling notations were used in the study: UML's Sequence Diagram, Service Component Specification, Subsystem Dependency Diagram, and an instantiation of the SOMA reference architecture. WSDL was also used for describing and formalizing web services, and finally a prototype was developed by using WCF Framework and PHP SOAP. In this case study, several components of the framework, i.e. the modeling tools, are introduced and tested.

5.3. Third Case Study

The third case study was done after the first draft of the framework was formalized. The case study employed the full proposed framework in applying SOA approach for managing data silos in the Government Statistical Agency. The case study was started with strategic analysis of the agency with BMC and followed by a series of BPMN to describe and to analyze business processes. Due to the focus on the internal agency process, service blueprinting tool was only used lightly to specify the data dissemination service based on request from customers. MSOAM was selected as the SOA method and was employed during the design stage to produce three layers of service: task-centric, entity-centric, and application utility. Services and service composition is described by using standard notation. Finally, the service interface is described by using WSDL.

5.4. Fourth Case Study

The fourth case study also selects the Government Statistical Agency as the study locus, but focusing on the design of proposed external data service for the agency. The work was also done by following the completed and formalized framework. The work produced two BMC diagrams: as-is BMC to describe existing situation and to-be BMC to propose the new service. The SWOT analysis was used between the two BMCs to understand the existing business model and to validate the need for the proposed service in the agency. Service blue printing and BPMN diagram were then used to design business specification of the service. For SOA design, MSOAM was employed with Thomas Erl's standard notation complemented with basic UML diagrams: Use Case, Class Diagram and Sequence Diagram.

5.5. Usability Test

Despite several required field adjustments and implementation variations, the adoption attempt on these case studies proves the usability Service Engineering Framework. To further test and evaluate the concepts, the proposed framework was taught in an IT master degree class for a semester[1]. Each student then assigned to perform a case study on an organization by designing and developing a prototype of technology-enabled services by adopting the methodology from the framework.

Table 4. Questionnaire Structure

Hypothesis	Questions
1. Framework is easy to comprehend	Q1, Q5, Q6
2. Framework is easy to adopt and implement	Q3, Q8, Q9, Q10, Q11
3. Framework helps communicate project	Q2, Q4
4. Framework is straightforward	Q7

At the end of the semester, a survey was conducted to 30 students of the class to assess the values and benefits of the framework. The focus of the questionnaire is to test the simplicity of the framework, operationalized under four hypotheses described in table 4. The responds are given under 5-scale *likert* respond, from "strongly-agree" to "strongly disagree". The questions and their results are presented in the following charts.

[1] Master of Informatics, IT Concentration, School of Electrical Engineering and Informatics, Bandung Institute of Technology, 2014.

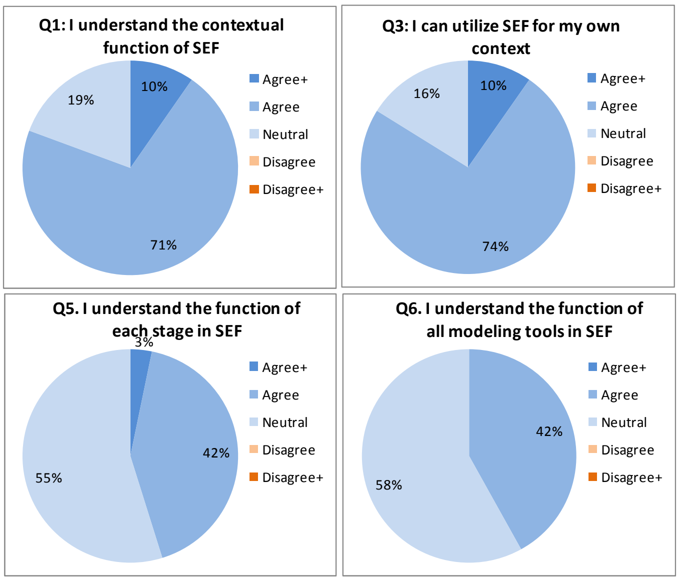

Figure 5. Survey Results on Comprehension to the Framework

The results from figure 5 show that more than 80% of the respondents agree that they understand and able to adopt the framework. On the other hand, while no respondents provide negative feedback regarding his understanding, only about 40% of respondents feel truly understand the role of each stage and tools employed in the framework. The later results provide a motivation to elaborate the stages description further and enrich the rational for each tool it employed.

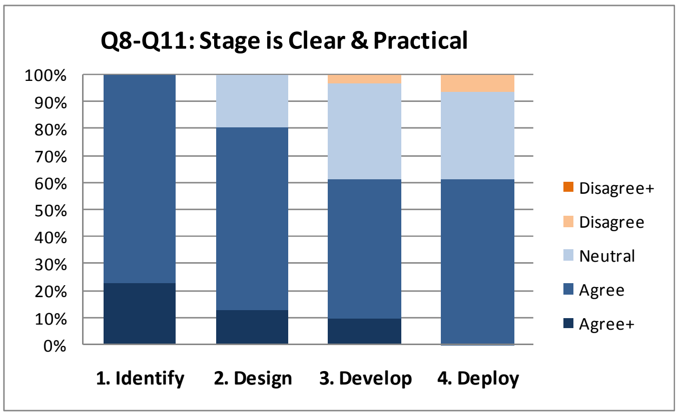

Figure 6. Survey Results on Stages Clarity and Practicality

Delving further into each successive stage in figure 6, we can see the improvement potentials of stages and tools elaboration. All of respondent is agree on the clarity and practicality of the identification stage. Also, 80% of the respondent has similar opinion for the design stage. The rate of agreement among respondents drops to 60% for the third (Develop) and fourth (Deploy) stage. The results for the third and fourth stage are rather expected due to the complexity of development required in the third stage, and the limited formalization defined in the framework for the fourth stage. Nevertheless, these results also support the motivation on further works for stage description, especially for the last two stages.

In terms of benefits of the proposed framework, figure 7 shows that more than 90% of the respondents agree that the framework can actually help an organization to introduce a new service. Also, more than 60% also agree that the framework help to visualize and communicate the project. While probably having no previous reference on other service engineering framework, 70% respondents still agree that the proposed framework able to streamline the process of service engineering.

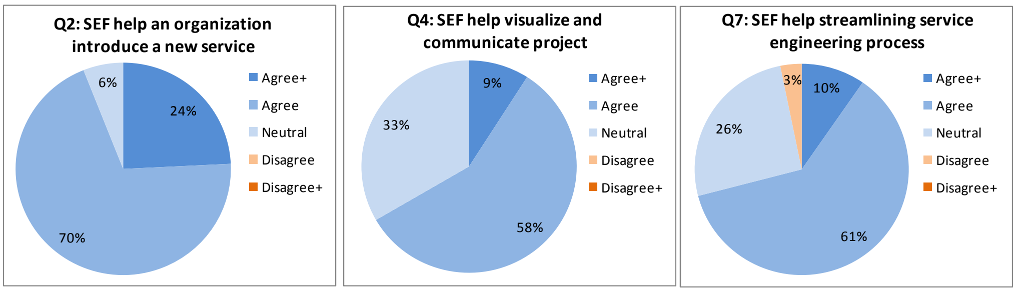

Figure 7. Survey Results on Benefit Values the Framework

These results from the survey already support the usability and the simple characteristic of the proposed framework.

6. Conclusion and Future Works

An alternative framework for adopting SOA methodology within SE framework is proposed here along with several trial case studies. The proposed framework has intentionally simplified the service engineering approach by explicitly select particular tools for each activity. The main analysis tool proposed in the service identification stage is BMC to innovate for new services with strong regard to the existing business model. Service blueprinting and Business Process Diagram tools are then used during service design to ensure that the proposed service is implementable. With the emphasis on technicality, the framework suggests much of the effort in implementing the SOA method during the service design and development. Case studies and survey results already indicated the both the usability and benefit of the proposed framework.

Further works to elaborate the details flow and artifacts of the second half of the framework is still required to achieve the goals of standardized approach. The proposed framework also only explicitly specifies testing and validation activity during the development stage in the form of unit tests, integration test, system/service test, and user acceptance test. An alternative of this approach is to introduce a testing activity in each appropriate stage. This mechanism will reduce probability failure or deviation of iteration from the earlier stage, in an effort to control the risk. Compared to a single validation test in the final stage, the drawback of these verification mechanisms is that more resource, effort and cycle time should be allocated. Therefore, the simplified approach of the proposed framework will be more suitable for smaller scope of service introduction.

A tool suggested for verification during identification stage is Quality Function Deployment (QFD). QFD matrix can be used to check the conformity between the organizational needs and the proposed service. QFD can also be used to verify whether the resulting service design produced during the second stage meets the service specification mandated by the identification stage. The SOA design as a by-product of the design stage will take the form as a design model. Therefore, verification might be done with the help expert review judgment. Before moving on to the other characteristic goals of the framework, several improvements to the framework are still required in addressing the simplicity feedbacks provided by the survey result. Further elaborations are required in improving the definition of each stage, the flow of process and the rational of the tools employed.

The framework also needs to be tested on its value on conforming the business real situation and its ability to provide uniform stage pattern, artifacts and results. To test the conformity of the framework, a case study must be performed to acquire feedback from real business process owner in a real business situation. The third characteristic can only be validated after examining results from multiple case studies.

References

[1] Bullinger HJ, Fähnrich KP, Meiren, and Thomas T. "Service engineering – methodical development \ of new service products". *International Journal of Production Economics*. 2003; 85(3): 275-287.
[2] Stauss B, Engelmann K, Kremer A, and Luhn A. *Services science: Fundamentals, challenges and future developments*. Springer Science & Business Media. 2007.
[3] Luczak H, Gudergan G. "The Evolution of Service Engineering—Toward the Implementation of Designing Integrative Solutions". *Introduction to Service Engineering*. 2009; 545-575.
[4] Hermawan, Sarno R. "Developing distributed system with service resource oriented architecture". *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2012; 10(2); 389-399.
[5] Wu LC, Wu LH. "Service Engineering: An Interdisciplinary Framework". *Journal of Computer Information Systems*. 51(2) 2010, 14.
[6] Kohlborn T, Korthaus A, Chan T and Rosemann M. "Identification and analysis of business and software services-a consolidated approach". *IEEE Transactions on Services Computing*. 2009; 2(1): 50-64.
[7] Cardoso J, Voigt K and Winkler M. "Service engineering for the internet of services". *Enterprise Information Systems*. Springer, Berlin, Heidelberg. 2009: 15-27.
[8] Lin FR and Hsieh PS. "A SAT view on new service development". *Service Science*. 2011; 3(2): 141-157.
[9] Pyster A, Olwell DH, Hutchison N, Enck S, Anthony JF Jr, and Henry D. "*Guide to the Systems Engineering Body of Knowledge (SEBoK)*. 2012; 1.0.1.
[10] Lopes AJ and Pineda R. "Service Systems Engineering Applications". *Procedia Computer Science*. 2013; (16).
[11] Suhardi, Budhiputra PM, Yustianto P. "*Service Engineering Framework: A Simple Approach*". Proceedings of International Conference on Information Technology Systems and Innovation (ICITSI) 2014, Bandung-Bali. 2014.
[12] Xu W, Li N. "An SOA-based Noise Mapping Platform for Urban Traffics". *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2013; 11(5): 2780-2790.
[13] Gholami MF, Habibi J, Shams F and Khoshnevis S. "Criteria-Based evaluation framework for service-oriented methodologies". In *Computer Modelling and Simulation (UKSim)*, 2010 12th International Conference on, IEEE, 2010: 122-130
[14] Ramollari E, Dranidis D and Simons AJH. "A survey of service oriented development methodologies". In The *2nd European Young Researchers Workshop on Service Oriented Computing*. 2007; 75.
[15] Emadi M, Jazi MD, Moghadam RA and Bahredar F. "An improved methodology for service oriented architecture". *In Computer Science and Automation Engineering (CSAE)*, 2012 IEEE International Conference on, IEEE, 2012; 2: 350-354.
[16] Gu Q and Lago P. "Guiding the selection of service-oriented software engineering methodologies". *Service Oriented Computing and Applications*. 2011; 5(4): 203-223.
[17] Svanidzaitė S. "A Comparison of SOA Methodologies Analysis & Design Phases". *Databases and Information Systems Baltic DB&IS* '2012. 2012: 202.
[18] Arsanjani A, Ghosh S, Allam A, Abdollah T, Ganapathy S and Holley K. "SOMA: A method for developing service-oriented solutions". *IBM systems Journal*. 2008; 47(3): 377-396.
[19] Erl T. *Service-oriented architecture: concepts, technology, and design*. Pearson Education India. 2006.
[20] Gu Q and Lago P. "Guiding the selection of service-oriented software engineering methodologies". *Service Oriented Computing and Applications*. 2011; 5(4): 203-223.
[21] Ernest M and Nisavic JM. "Adding value to the IT organization with the component business model". *IBM Systems Journal*. 2007; 46(3): 387-403.
[22] Mohammadi M and Mukhtar M. "A Review of SOA Modeling Approaches for Enterprise Information Systems". *Procedia Technology*. 2013; (11): 794-800.
[23] Osterwalder A and Pigneur Y. *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*. Wiley. 2010.
[24] Kandampully J. "Innovation as the core competency of a service organisation: the role of technology, knowledge and networks". *European Journal of Innovation Management*. 2002; 5(1): 18-26.
[25] Bitner MJ, Ostrom AL and Morgan FN. "Service blueprinting: A practical technique for service innovation". *California Management Review*. 2008; 50(3): 66.
[26] Yoon H. "Test Requirements of Service Connections with their Applications to a Testing Tool". *International Journal of Computer Science and Network Security*. 2010; 10(3): 129.
[27] Wijayanto AW, Suhardi. "Service Oriented Architecture Design using SOMA for Optimizing Public Satisfaction in Government Agency Case Study: BPN – National Land Authority of Indonesia". *Proceeding on ICT for Smart Society (ICISS)*, Bandung. 2014.