Service Computing System Engineering Life Cycle

Suhardi, Novianto Budi Kurniawan, Jaka Sembiring
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
suhardi@stei.itb.ac.id, noviantobudik@s.itb.ac.id, jaka@stei.itb.ac.id

***Abstract*—The development of service computing technology has triggered the need for organizations to build a service computing system. This system is able to deliver IT services that can meet the challenges and needs of business services. An understanding of the service computing systems engineering life cycle is necessary to model and build the systems. This paper conducts the study of the service computing systems concepts and proposes the life cycle of service computing system engineering. First, some publications relevant to the service computing systems are reviewed systematically in order to develop fundamental concepts and elements of the systems. Second, a metamodel of service computing systems is built based on those concepts. Third, the life cycle model of service computing systems engineering is proposed based on the metamodel. Finally, the verification of the life cycle is conducted by comparing the proposed life cycle to the life cycle of software engineering, service engineering, and service system engineering. The contribution of this paper is to provide a coincise understanding of the components that are needed to build such systems and show the interaction and collaboration between the components in the service computing systems.**

Keywords— Service Computing, Service computing systems, Services computing system Engineering, Service Oriented Architecture, IT Services.

I. BACKGROUND

IT has become a critical component in supporting innovation and the provision of business services organization [1]. Through IT, business process automation and acceleration of service innovation can be made to increase the productivity of services [2]. Over IT, organizations can use the infrastructure network (*network services*) to link between services and improve service performance [2][3]. Technology platforms, web services ecosystem, social media, and networking technologies are some of the technology services that can be used to support and improve the innovation of business services. [4][5]. Service-based architecture and technology, such as Service-Oriented Architecture (SOA), Microservices, Web services, Cloud Computing, and Grid Computing have evolved and provided opportunities for the development of organization's business services innovation. IT supports in the form of electronic services (*e-services*) makes users easy to use and take benefit of the business services. IT supports in the form of the IT application services (*software services*) are also a crucial element. Organizations need IT service system to fulfill the business needs and able to meet users' requirements.

Although IT services are already widely used to support business services, there are some phenomena that occur in the organization. This condition can be seen from the two approaches: business needs approach and IT trends approach. Of the approach to business needs, the organizations must always be able to meet the goals of its business through the business services provided, because of either the existence of the demands or the needs of the service users or triggered from the competition impact of business services. It causes the organization to be able to implement a business model that is able to quickly adapt to the business environment (this refers to enterprise agility). As a consequence, it will trigger business services to be fast and dynamic. However, these conditions are not the same as the IT infrastructures and applications services that support the business services. The IT infrastructures and application services tend to be less agile because it involves various hardwares and softwares so that they are more complex than the business. This will result in a condition that IT services are not able to meet the demands of business services. Moreover, this wil have an impact on business service innovation.

From the IT trend approach, the new technology can not be easily applied or implemented in the organization. It should be done strictly in accordance with the business needs and business objectives of the organization rather than just following the current global trends. The implementation of new technology is intended to create IT services that support the business services and provide value to the organization. In addition, the tendency of the organization's business units that do not understand the trend of IT is also becoming a challenge in this new technology implementation. In addition, the implementation of the IT trend is considered as the IT investments that increase the financial budget of the organization. These things can, of course, be barriers in the process of accelerating the business services innovation and triggered disparities between the IT and the business. Furthermore, these conditions cause the gap between IT services and business services.

The gap between IT services and business services became a major concern in the service computing. Service computing aims to overcome the gap by implementing service-based technology to provide business services more efficiently and effectively. The application of services computing technology is needed to create IT services that support the innovation of organization's business services. In services computing, each service is viewed as a service system that represents a collaboration of service model, service technology, IT architecture, and service optimization methods/techniques [6]. This represents a service computing system. Services computing systems do not only focus on the IT domain but also cover the entire life cycle of business processes that focus on the application of technology-based services such as web services, service-oriented architecture to create a service that

978-1-5386-0549-3/17/$31.00 ©2017 IEEE

meets the needs of the organization. This indicates that the service computing systems are not merely the application of technology-based services, but should pay attention to the business needs of the organization to elaborate on the requirements of the service users with the capability of business process and technology owned by the organization. This emphasizes service computing systems as the approach to service-based IT solutions that are able to align the IT services and business services and be able to bridge the gap between both services.

II. Literature Review

A. *Services Computing*

Definition of services computing has been proposed by some experts from various perspectives. From the perspective of software system design and development, Papazoglou and Georgakopoulos [7] defines the service computing as a way to build the service-oriented applications systems. From the perspective of application of service technology, Singh and Huhns [8] defined service computing as the set of technologies that combine the concepts of services, service system architecture, service technology, and infrastructure together to provide support and guidance how to use the services. From the perspective of distributed computing, Orlowska and Weerawarana [9] defined service computing as a distributed computing paradigm that evolved from object-oriented computing.

From the discipline perspective, Zhang et al. [6] define service computing as interdisciplinary science that includes computers, information technology, information management, and consulting services with the aim to eliminate the gap between business services with IT services through the application of service computing technology. Wu et al. [10] define service computing as a new computing discipline with web services and SOA as the primary technology service support, service composition as the main approach in developing software and based on the principles of service-oriented software design and analysis. From the definitions above, it can be stated that the service computing focuses on the application of service computing technology and infrastructures (such as software, hardware, infrastructure, service technology, service architecture) to the modeling, design, and implementation of IT services to increase the productivity and effectiveness of business services.

B. *Services Computing Technology*

Some of the service computing technologies have emerged and evolved to support services computing. SOA, Web services, Microservices, Cloud Computing, and Grid Computing are some examples of service computing technologies. SOA and web services are the underlying technology in service computing [10]. SOA is a fundamental architectural model that supports the whole paradigm of service computing from an architectural point of view. SOA is a service-based architecture that allows the design of application services (software) that are loosely-coupled to meet the needs of business processes and users. Through SOA, each application is built to be treated as a service to support business process services. SOA creates flexibility of IT services, such as the use of the service (reusability) and interoperability, which can support business process automation. SOA implementation mainly involves the construction of service-based applications and makes the application as a service available to others.

Web service is the main technology used in service computing to implement SOA and realize IT services. Web service exists as a software system that is aimed to support the services interoperability that can interact over the network (via internet protocol). Web service provides functions and operations that can be accessed over a network and does not rely on specific technology platforms and programming language so that it is able to support interoperability between services. This technology model is able to facilitate collaboration between business services openly because every organization can expose its business application services through the Internet. Through SOA and web services, this type of technologies can be used to build a service system that is able to meet the business needs of the organization.

C. *Service computing systems*

A service computing system is an IT service encapsulation system built on SOA by applying service computing technology. The system is based on the implementation, utilization, and development of service computing technology. Huang et al. [11] [12] defines the service computing system as a complex system, which involves the interaction, composition, and collaboration among IT services, as well as the interaction and dependencies between users and service providers. Chen and Lin [13] describe service computing system as a group of interacting IT services that not only focus on the technology of service computing and IT infrastructures but also pay attention to the performance of the IT services (such as QoS, service discovery). Moreover, Ye et al [14] describe service computing system as a complex system that consists of various services that can be consumed by users by integrating individual services into composite services to meet the complex requirements of users. Based on the definitions above, it can be summarized that service computing system is a service system enabled by service computing technology, meaning this system focuses on the enablement of the concept of service systems through the model, technology, architecture and optimization & analysis. In short, this is a system that uses computing technology to build service systems.

Table I shows the synthesis of concepts and components of a service computing system which are summarized from previous related studies. As service computing is the development of service-oriented computing paradigm, then the concept of service computing systems (SCS) itself also evolves from service-oriented computing systems (SoCS), which not only focuses on IT services (ITS) but also includes business services (BS), interaction between the two services, and the evaluation of IT services in order to support the productivity and effectiveness of business services. From the table, it appears that the concept of service-oriented computing systems only focuses on IT services, IT systems, IT infrastructure, and of course SOA, while the service computing systems is an extension of the service-oriented computing system paradigm by considering another aspects of business services, service users, service providers, service systems, and the interaction between the services.

TABLE I. SERVICE COMPUTING SYSTEM CONCEPTS

Ref.	Services computing system(s)		
	Concepts	***Term***	***Component(s)***
[15]	Complex and large-scale distributed computing systems with SOA as a reference model for achieving the systems.	SoCS (ITS)	Complex system, SOA
[16]	An enabling technology architecture that supports the future Web, which coordinates human interactions, business processes, computing resources, and business applications in a uniform fashion.	SoCS (BS-ITS)	Technology architecture, Interactions, Business processes, Resources
[17]	Computer-based service systems built from Service-oriented computing paradigm based on SOA	SoCS (ITS)	SOA
[18]	A system packaging information, computing resources belongs to different virtual organizations into services with uniform interfaces and call method, and provide them to users.	SoCS (BS)	Information, Computing resources, Users
[19]	A system of dynamic service selection and composition in order to fulfill a user's request and their related architecture (SOA).	SoCS (ITS)	Service selection, composition, SOA
[20]	A large and complex service-oriented application systems such as Internet-based grid systems, e-market places, computing systems.	SoCS (ITS)	SOA, Service computing technology
[21]	The system's point of view of IT systems and services with its characteristics of massive-scale service sharing, wide-area network, heterogeneous components and interactions among them.	SCS (BS-ITS)	IT systems, Service sharing, Network, Components, Interactions
[22]	A collection of interacting services that constitute a set of functionality offered by a service provider or server to its clients.	SoCS (ITS)	Service interactions, Service provider Service client
[23]	A system that provides a way for users to access services, such as web applications.	SCS (BS-ITS)	Services discovery, Web application
[11]	The large-scale IT systems (in a well-defined architecture) that have been deployed across the Internet.	SCS (ITS)	IT systems, IT architecture, Internet
[24]	A type of complex systems consisting of a number of loosely coupled autonomous and adaptive components.	SoCS (ITS)	SOA, Components
[12]	Large-scale complicated services computing systems (IT service systems) such as SOA or Cloud.	SCS (ITS)	IT systems, IT services, SOA
[13]	A complex system that consists of various services by integrating individual services into composite services to meet the requirements	SCS (ITS)	Complex systems, Service composition, Requirements
[14]	Encapsulation of IT service systems and services (in a well-defined architecture) with large-scale computing infrastructures	SCS (ITS)	IT services systems, Architecture, Infrastructure

Fig. 1 shows how the components of a service computing system can be enabled through the interaction and collaboration of four elements. Zhang et al [6] represent the elements (components) in the IT service system that should be supported by the service model, services computing technology, service architecture, and service optimization and analysis. Service model describes how IT services operate, how IT services are delivered to the users, and how the capacity of IT services is provided. In building the system, it requires the relevant and essential service computing technology to improve the performance of IT services. The IT architecture (includes data/information, application, and technology/infrastructure) is an important component in building the services computing systems. This architecture provides guidance in designing and building the system. Finally, the methods for optimizing the systems are fully needed so that the system can be built, developed, managed, and improved continuously.

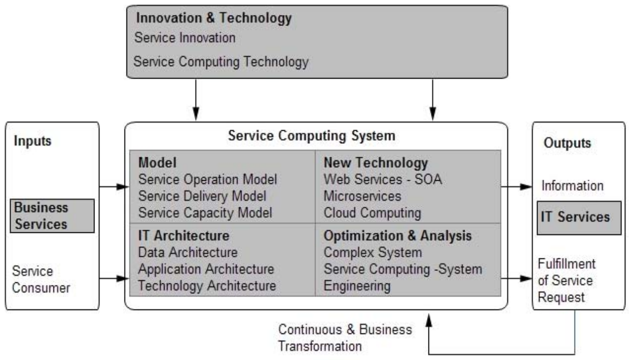

Fig. 1. Enablement of services computing systems (Modified from [6])

III. METHODOLOGY

This study applies the literature review to build the concept of service computing systems and proposes the life cycle of the system. The methodology is encompassed into four steps. First, some publications relevant to the service computing system are reviewed systematically in order to develop fundamental concepts and components of the systems (as seen in Table I). Second, a metamodel of service computing systems is built based on those concepts. Third, the life cycle of service computing system engineering is proposed based on the metamodel. For this purpose, TOGAF content metamodel that visualizes the development of Enterprise Architecture life cycle is used as the same approach to the development of the service computing system metamodel [25][26]. Finally, the verification is conducted by comparing the proposed life cycle to the life cycle of software engineering, service engineering, and service system engineering.

IV. RESULTS

A. Metamodel of Services computing systems

Fig. 2 shows a metamodel of services computing systems that is constructed based on the study of services computing systems concepts and components. The metamodel is organized to illustrate the components that are needed to build a service computing system which are grouped into five layers.

1) Service Strategy, Objectives and Requirements: This layer captures the strategies, goals, and requirements of services computing systems that will be built. The components of the layer are (1) Service strategy and objectives (cover business strategy, technology strategy, service objectives and drivers, and stakeholders), (2) Service requirements (cover business services, service request, and the user's requirements), and (3) Service actor (covers service users, service customers, service providers, and service brokers)

2) Service Model and Architecture: This layer describes the model and the technology of service computing systems that will be constructed as a reference to design the architecture of service computing systems and to provide the mechanisms for system implementation. The components are (1) Service model (covers business process, service description, and service candidate), (2) IT resources (include information, IT Infrastructures, application system), (3) Service technologies (include SOA, web application, web services, microservices, and cloud systems), (4) Service architecture (covers business service architecture, data.information architecture, application architecture, infrastructure architecture), (5) Service catalog, and (6) service transformation (includes service orchestration, service interaction, service sharing, human interaction, service collaboration, and service network.

3) Service Development and Realization: This layer represents the implementation and development of services computing systems based on the model and architecture design from the previous layer. The components are (1) Service implementation (service coding and testing), (2) Software system development, (3) Service composition (includes service discovery and service selection), (4) Service integration, (5) Service invocation, (6) Service validation, and (7) Service migration planning.

4) Service Production and Operation: This layer explains how the service computing systems is ready for rollout and delivered to the users as well as provides an overview of the service computing system operational procedures. The components are (1) Service rollout, (2) Service testing, (3) Service repository, (4) Service registry, (5) Service publishing, (6) Service operation, and (7) Service monitoring.

5) Service Evaluation: This layer plays a major role in evaluating, measuring and optimizing the performance of services computing systems. The components of this layer are (1) Service performance (covers service quality, service dependability, service experience, and service response), and (2) Service Analysis and Optimization (covers the analysis of the new system performance and the optimization for improving the system)

B. Life Cycle of Service Computing System Engineering

Based on the metamodel above, a life cycle of service computing system engineering is built by transforming every layer in the metamodel into every stage of the life cycle. Every cycle that represents stages shall be interconnected and sustainable so that all of the stages form a continuous life cycle. Fig. 3 shows the life cycle of service computing systems engineering that comprises five stages: (1) objectives and requirements, (2) modeling, (3) development, (4) deployment, and (5) evaluation. The whole stages are interacting as an integrated framework. The five stages are systematically linked together and provide a guidance in the process of service computing systems engineering.

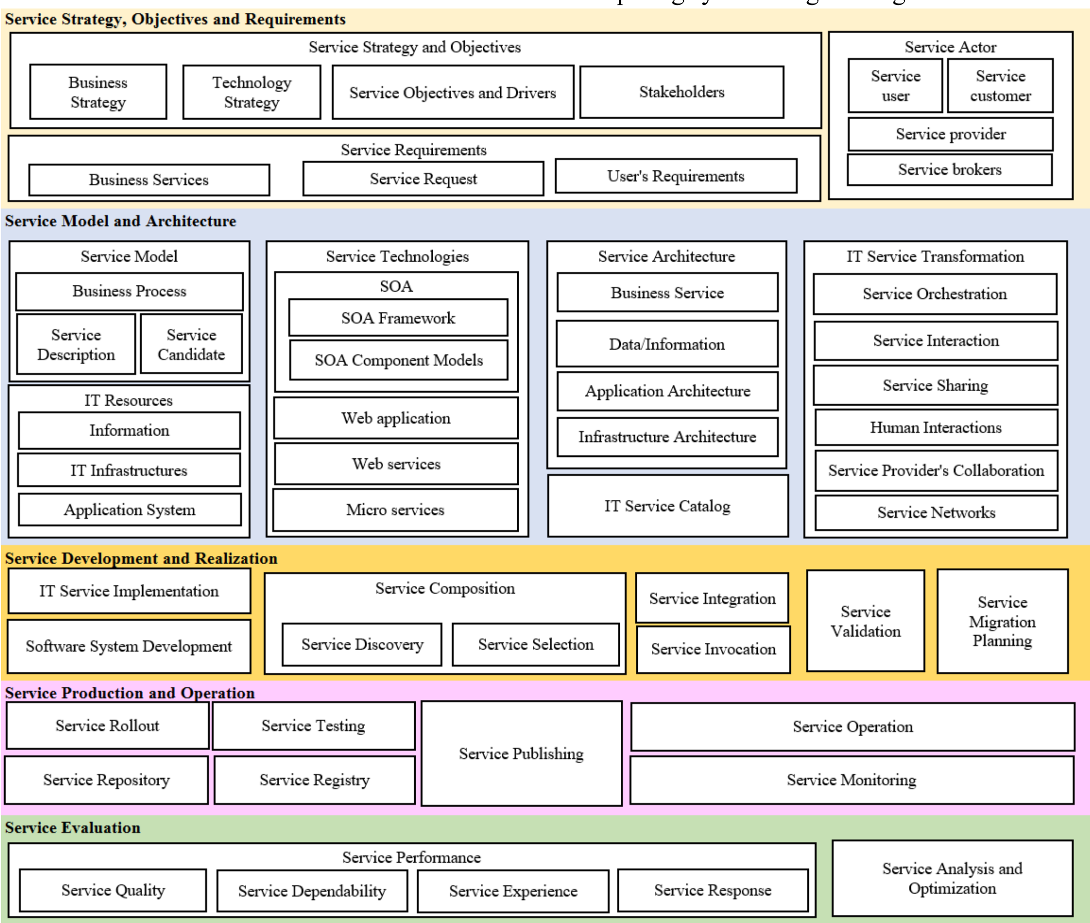

Fig. 2. Metamodel of services computing systems

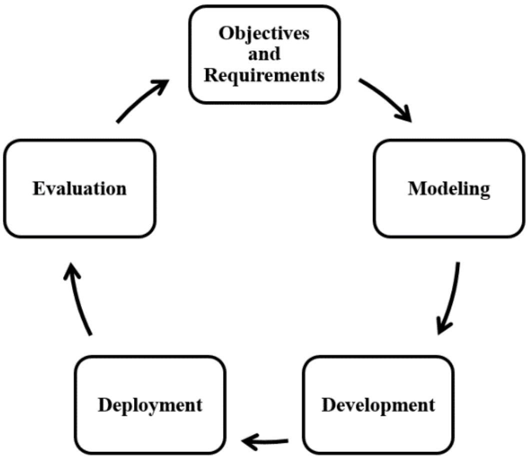

Fig. 3. Services computing systems engineering life cycle

1) Objectives and Requirements: This is the initial and preparation stage in the process of service computing systems engineering. At this stage, the identification and analysis of system requirements are conducted based on the approach to the business needs and the service improvement. The understanding of business strategy and IT strategy is absolutely necessary to compile the service objectives and drivers to suit the needs and objectives of the organization. Collaboration with stakeholders is also required in order to capture the needs and the intention from the executive's point of view. Service needs analysis is conducted based on the evaluation of the existing performance, the service request from the user, and the user's requirements. Identification of service actors also has to be comprehensive.

2) Modeling: At this stage, the design of service computing systems is conducted by using the technique of modeling, through business process analysis to get the business service candidate that need to be improved. This stage plays the very important role in the whole life cycle, given that the stage includes three areas of the enablement of service computing systems, i.e. the service model, technology, and architecture. The analysis of IT resources and the selection of service computing technology become the important activities in this stage. SOA and web services are the main technology used in the development of service computing systems, including the strategy of how to select the engineering approaches and SOA framework. The service architecture should be built, not only covers the IT architecture (data/information, application, and technology architecture) but also includes the business architecture. The interactions between models, technology, and architecture are then represented in the SOA reference architecture. Furthermore, the processes of transformation and interaction between IT services (cover service orchestration, service interaction, collaboration, etc) are conducted.

3) Development: This stage outlines service computing system development activities in order to build IT service systems. The implementation of IT services include service coding and service testing that become the main activity in this stage. The selection of software development methodology is also a major concern. The output (IT services systems) are implemented using web services technology. In addition, service migration planning should also be considered to accommodate the preparation to the production of environment.

4) Deployment: This stage enables service computing systems in order to be operationalized and be used by the user. A series of activities to ensure the system that is completely ready to run had to be traversed and the system monitoring shall also be fully conducted.

5) Evaluation: In principle, this stage delivers the evaluation process of the operationalized service computing systems by performing some performance measurements. The results of this evaluation should be analyzed in depth and used as the reference materials to optimize systems performance. This will lead to the service computing system improvement.

V. VERIFICATION

The verification of the proposed life cycle is conducted by comparing the service computing systems engineering life cycle to the lifecycle of software engineering (SWE), service engineering (SE), and service system engineering (SSE). Tabel II shows the result of mapping and comparison between the proposed life cycle (services computing systems engineering) with the engineering methodologies (SWE, SE, and SSE).

TABLE II. LIFE CYCLE VERIFICATION RESULT

Proposed Life cycle components	SWE	SE		SSE	
	[27]	[28]	SOA [29]	[30]	[31]
I. Service Strategy, Objectives & Requirements	**System/ Software Require-ments**	**Identify**	**Service Analysis**	**Identifi-cation**	**Service Need/ Strategy**
Service Strategy and Objectives	-	-	V	V	V
Service Requirements	V	V	V	V	V
Service Actor	-	-		-	-
II. Service Model and Architecture	**Design**	**Design**	**Service Modeling & Design**	**Design**	**Service Design**
Service Model	V	V	V	V	V
IT Resources	-	-	-	-	-
Service Technology	V	V	-	-	-
Service Architecture	V	V	V	V	-
Service Transformation	-	-	V	-	-
III. Service Development & Realization	**Develop**	**Develop**	**Service Develop.**	**Develop**	**Service Develop.**
Service Implementation	V	V	V	V	V
Software System Development	V	V	V	V	V
Service Composition	V	V	V	V	V
Service Integration	-	V	V	V	V
Service Invocation	-	-	-	-	-
Service Validation	V	V	V	V	V
Service Migration Plan.	V	-	-	-	-
IV. Service Production and Operation	**Operation**	**Deploy**	**Service Deploy-ment**	**Deploy**	**Service Deployment**
Service Rollout	V	V	V	V	V
Service Testing	V	V	V	-	V
Service Repository & Registry	-	-	-	-	-
Service Catalogue	-	V	-	-	-
Service Publishing	-	V	V	-	-
Service Operation	V	V	V	V	V
Service Monitoring	V	V	V	V	V
V. Service Evaluation	-	-	-	-	**Continuous Service Improvement**
Service Evaluation	V	-	V	V	V
Service Performance	V	-	-	V	V
Service Analysis & Optimization	-	-	-	V	V

From the three engineering methodologies above, SSE is the closest approach that meets the completeness of the proposed life cycle component. This is because the subject of SSE is a service system (or complex service systems) which is or are actually similar to the service computing systems. The deficiencies of SSE are the identification of the service actor, the analysis of IT resources, the implementation strategy of service computing technology, the development of service architecture, and the transformation of services. In addition, the weakness of this type of engineering is also noticed as the absence of service migration planning, service repository & registry, and the mechanism of service publishing. On the other hand, SE still keeps the lack of business analysis, because this methodology is actually focused on the development of IT services. An assessment of the IT service performance has also become another weakness of this methodology. SWE as a methodology that focuses on the development of application system (software) also has some shortages which are the lack of analysis of business service strategy and objectives, the identification of the service actors, and collaboration between applications. In addition to that, the optimization of service applications is also not systematically covered in this life cycle.

VI. CONCLUSION

In this paper, the research on service computing systems gives the understanding of service computing, service computing technology, and service computing systems from the engineering perspective. Some of the fundamental technologies of computing system services have also been described. The needs for building the services computing systems in an organization have also been presented in this paper. Through this study, the metamodel and life cycle of service computing systems are proposed. The objective of this paper is to propose a service computing system engineering life cycle based on the concepts of service computing, computing technology, and services computing systems. The contribution of this paper is to provide a concise understanding of the components that are needed to build such systems and shows the interaction and collaboration between the components in the systems. Moreover, this paper becomes a good foundation for identifying stages and steps that will be needed in building service computing systems. Therefore, future research will focus on the development of service computing system engineering framework based on the proposed life cycle.

REFERENSI

[1] P. Yustianto, Suhardi, and R. Doss, "Consolidating service enginering perspectives," *2015 Int. Conf. Inf. Technol. Syst. Innov. ICITSI 2015 - Proc.*, pp. 1–7, 2015.

[2] P. P. Maglio, S. Srinivasan, J. T. Kreulen, and J. Spohrer, "Service systems, service scientists, SSME, and innovation," *Commun. ACM*, vol. 49, no. 7, pp. 81–85, 2006.

[3] C. Hsu and J. C. Spohrer, "Improving service quality and productivity : exploring the digital connections scaling model," *Int. J. Serv. Technol. Manag. Vol.*, pp. 272–292, 2009.

[4] A. P. Barros and M. Dumas, "The rise of web service ecosystems," *IT Prof.*, vol. 8, no. 5, pp. 31–37, 2006.

[5] S. Barile, R. Lusch, J. Reynoso, M. Saviano, and J. Spohrer, "Systems, Networks, and Ecosystems in Service Research," *J. Serv. Manag.*, vol. 27, no. 4, 2016.

[6] L.J. Zhang, J. Zhang, and H. Cai, *Services Computing*. New York: Springer US, 2007.

[7] M. P. Papazoglou and D. Georgakopoulos, "Introduction: Service-oriented Computing," *Commun. ACM*, vol. 46, no. 10, pp. 24–28, 2003.

[8] M.N. Huhns, and M. Singh. "Service-oriented computing: Key concepts and principles," IEEE Internet Computing, (February), pp. 75–81, 2005.

[9] M.E. Orlowska and S. Weerawarana, et al. (Eds.), Proceeding of First International Conference on Service-Oriented Computing, Springer, 2003

[10] Z. Wu, S. Deng, and J. Wu, *Service Computing Concepts, Methods and Technology*, vol. 1. Waltham, MA: Elsevier Inc., 2015.

[11] J. Huang, C. Lin, X. Kong, B. Wei, and X. Shen, "Modeling and analysis of dependability attributes for services computing systems," *IEEE Trans. Serv. Comput.*, vol. 7, no. 4, pp. 599–613, 2014.

[12] J. Huang, C. Lin, and J. Wan, "Modeling , Analysis and Optimization of Dependability-Aware Energy Efficiency in Services Computing Systems," in *IEEE 10th International Conference on Services Computing*, 2013.

[13] D. Ye, Q. He, Y. Wang, and Y. Yang, "An Agent-based Integrated Self-evolving Service Composition Approach in Networked Environments," *IEEE Trans. Serv. Comput.*, vol. 1374, no. c, 2016.

[14] Y. Chen and C. Lin, "Energy Efficient Scheduling and Management for Large-Scale Services Computing Systems," *IEEE Trans. Serv. Comput.*, vol. 10, no. 2, pp. 217-330, 2017.

[15] G. Cai, J. Gao, and B. Hu, "Computing System Development," 2006.

[16] L. Zhang, "Service HyperChain Architecture of Web X . o and A Case Study," no. Icws, 2007.

[17] H. Sarjoughian, S. Kim, M. Ramaswamy, and S. Yau, "A Simulation Framework for Service-Oriented Computing Systems," no. Acims 2001, pp. 845–853, 2008.

[18] Y. Xian-Zhen, J. Jing, P. Zhen-Kuan, and X. Bai-Qiang, "2008 International Seminar on Business and Information Management," pp. 216–220, 2008.

[19] A. F. N. Lumala, "A Distributed Problem Solving Approach for Service-oriented Computing Systems," 2009.

[20] X. Su, M. Zhang, Y. Mu, and K. M. Sim, "PBTrust : A Priority-Based Trust Model for Service Selection in General Service-Oriented Environments," 2010.

[21] J. Huang, C. Lin, X. Kong, and Y. Zhu, "Modeling and analysis of dependability attributes of services computing systems," *Proc. - 2011 IEEE Int. Conf. Serv. Comput. SCC 2011*, pp. 184–191, 2011.

[22] M. Stal, D. C. Schmidt, and W. R. Otte, "Efficiently and transparently automating scalable on-demand activation and deactivation of services with the activator pattern," 2011.

[23] C. Mu and W. Zeng, "Security Measurement in Service-based Computing Systems," 2012.

[24] J. Jiang and Q. Bai, "Correlated Contribution Analysis for Service Composition in Dynamic Environments," 2013.

[25] The Open Group, *TOGAF ® Version 9.1 Enterprise Edition*. 2011.

[26] A. Zimmermann, H. Buckow, O. F. Nandico, G. Piller, and K. Prott. "Capability Diagnostics of Enterprise Service Architectures using a dedicated Software Architecture Reference Model*", IEEE International Conference on Services Computing*, 592–599, 2011

[27] A. M. Davis, E. H. Bersoff, and E. R. Comer, "A Strategy for Comparing Alternative Software Development Life Cycle Models," vol. 14, no. October, 1988.

[28] Suhardi, R. Doss and P. Yustianto, "Service Engineering Based on SOA Methodology," pp. 1–12, 2015.

[29] T. Erl, *Service-Oriented Architecture: Analysis and Design for Services and Microservices*, Second Edi. USA: Prentice Hall, 2017.

[30] Suhardi, N. B. Kurniawan, J. Sembiring, and P. Yustianto, "Service Systems Engineering Framework Based on Combining Service Engineering and Systems Engineering Methodologies," *International Conference on Research and Innovation in Information Systems*, 2017.

[31] A. J. Lopes and R. Pineda, "Service Systems Engineering Applications," *Procedia Comput. Sci.*, vol. 16, pp. 678–687, 2013.