Paper 3

by Paper Aradea

REQUIREMENTS ENGINEERING FOR CLOUD COMPUTING ADAPTIVE MODEL

Kridanto Surendro, Aradea & Iping Supriana
School of Electrical Engineering and Informatics
Bandung Institute of Technology, Indonesia

endro@informatika.org; aradea.informatika@gmail.com; iping@ informatika.org

ABSTRACT

The easy access and flexibility provided by cloud computing have made various types of organizations use this technology as an alternative solution. However, the decision to use the cloud technology sometimes ignores the basic aspects related to the understanding of its compliance with the characteristics of the organization. Thus, many advantages of the adoption of the technology do not align with the needs. This paper discusses the requirements engineering of the alignment of the organizational characteristics and the cloud computing technology in order to create the adaptive ability to guide strategic-business driven organizations. The model proposed in this article is formulated based on three views: architectural, alignment, and adaptive, by employing the Goal-Oriented Requirements Engineering. As a case study, the model is applied to cloud computing for university. The architectural view can translate the environmental characteristics of organizations through goal decomposition into detailed needs. Meanwhile, the alignment view can meet the implementation, so the dynamic adaptive ability can be realized through the adaptivity view.The mechanism of our proposed model can cover the adaptation needs before the migration (pre-migration), during the migration, and after the migration (post-migration).

Keywords: requirements engineering, adaptive model, cloud computing, goal-oriented, capability maturity model.

INTRODUCTION

The increasing demand for cloud computing has become the trend of organizations with the hope that they can deliver quick, easy, and flexible services. However, it also raises some concerns, especially when the organizations are too enthusiatic and quick in making decisions without identifying and considering what the organization needs. One of the concerns is related to understanding the differences between traditional environment and cloud computing environment (Gill, 2015), which deal with the characteristics of some computational resources such as software, platform, and other infrastructures, including other characteristics influenced by the type of the organizations.

Each organization will require different computational resources. This is influenced by the business view of the organization which has particular characteristics different from the characteristics of other organizations, even when they belong to the same type of organization. Thus, the role of understanding the organization's environment should be well defined; a holistic understanding which covers the real/actual needs of organizations. If this aspect is not considered, computational resources will not support the organizations; rather, they have a short life time and become a burden for the organizations.

Key strategies which are proposed to overcome the problem include: (a) architectural view, that is an understanding of the arhitectural dimension which reveals the needs of computational resources based on business view, (b) alignment view, that is an integration and standardization model able to provide detailed descriptions and mapping of the needs, and (c) adaptive view, that is a formulation of the parameter needs and functions to adaptively realize the needed services. Using the Goa-Oriented Requirements Engineering (GORE) (Lapouchnian, 2005) we developed a requirement model for the organization to adopt cloud computing. The proposed model was generic, which could be applied to all types of organizations. For the purpose of this study, the cloud computing model for a university will be discussed as an example. Part II of this article describes the requirement approach, while part III discusses the proposed model and a case study. Part IV presents related discussions and tasks, and part V outlines the conclusions.

REQUIREMENTS FOR CLOUD COMPUTING MODEL

The complexity of cloud environment is difficult to identify, clarify, and manage its needs. This is because cloud environment is stochastic and dynamic

(Iankoulova, 2012 ; Zardari, 2011). Thus, factors related to changes of services and requirements of cloud computing become unpredictable. Zalazar (2015) defined 3 causes of the changes of cloud requirements: (a) organizational policy changes on business priorities, (b) environment or marketplace changes due to the emergence of new competitors and business targets, and (c) legal changes due to requests of new forms, features, and alogarithm of cloud applications. Based on such aspects, we argue that changes related to priorities and business targets require the formulation of a goal model. Thus, the architectural dimension can be employed as an initial view for defining the key elements through the identification of business facts and the environment. The view of the architectural dimension can be specified to generate a relevant definition of technology and related to factors of legislation changes. For realizing the view, we use the GORE approach formulated as a new model. This is further discussed in Part 3.

The GORE approach is employed because it has some advantages. For example, (Zalazar, 2015) the goal can be abstracted to some levels of abstraction, which are high-level goals, related to business strategies and the organizations, and low-level goals, related to the needs of technical objectives and design options (Zalazar, 2015). Therefore, goals can be formulated in different levels of abstraction in order to define functional and non-functional needs. Furthermore, GORE is able to capture, elaborate, structure, specify, analyze, negotiate, document, and trace requirements (Zardari, 2011; Wind, 2011).

Therefore, requirements and the model can be obtained systematically by defining a goal, through vertical decision reasoning and tracking from high-level strategies to technical value (Lamsweerde, 2009). In this article, we propose the expansion and a new view of the GORE approach to realize the ability of the cloud computing adaptive model.

Currently, there are various cloud computing adaptive models proposed by researchers. For example, Zalazar (2015) proposed the dynamic handling of the requirements in cloud computing by focusing on the workflow used as a guideline of the requirement activities. However, this model does not discuss in detail how the view of the need is defined from the available environment characteristics, while our approach defines it. Furthermore, Zardari, (2011) developed a mechanism of requirements by decomposing the goal model into the need of cloud adoption, including the adaptation need through the risk management approach. However, the developed model does not reflect the capture of the contexts representing the system environment coverage (context-

aware), as in our proposed model. Besides that, Katsaros, (2011) specified the requirements of the application adaptation and cloud the infrastructure through the QoS measurement using the multi-layered monitoring approach. However, how the application and the infrastructure become the elements which align with the environment characteristics, such as the alignment view in our model, is undefined. Meanwhile, Kertesz, (2014) focuses on the requirements for the adaptation ability through the model of the SLA-based service visualization. In this research, how the SLA is defined by considering its appropriateness with the needs of the organization is not discussed as in our proposed model.

CLOUD COMPUTING ADAPTIVE MODEL

The main service of cloud computing consists of three models (Amanatullah, 2013), which include (a) cloud infrastructure as a service (IaaS), the lowest level – having a role in processing, storing, networking, and other computational resources as a standardized service in all networks, (b) cloud platform as a service (PaaS), the middle level – providing abstraction on a higher level and services to develop, test, run, and maintain applications in an integrated environment of development, and (c) cloud software as a service (SaaS), the highest level – providing full access to the offered applications in the form of a service. Besides that, multitenancy is a concept in software architecture used in cloud-based applications. This architecture provides instantaneous application software which can be used by some users simultaneously (Yaish, 2013), so that the instantaneous application is logically isolated but physically integrated.

Based on the illustration of the basic concept of the cloud service model, we propose an approach that can be used to formulate the need of cloud computing through 3 views by using the ability of GORE. In this article, we define the need of the model, apply it in the case of a university, and evaluate it.

A.   Model Conception

Figure 1 and Table 1 illustrate the proposed model, consisting of 3 views – architectural, alignment, and adaptive. Based on the main elements in the GORE approach (Yu, 2009; Hui, 2003), the alignment of the organizational characteristics can be defined through the GORE basic elements, namely the goal, softgoal, plan, and resource. Goal is defined as the set of behavior or circumstance that has to be fulfilled or accepted by the system in the predefined condition. The goal definition must be clear enough such that it could be verified if the system is able to fulfill that goal, so that this element can be

seen as a requirement for business strategy and stakeholders. Each of these goals will have sub-goals that represent each of its business process. Softgoal is the condition that has a subjective level of satisfaction, or softgoal is used to document the system's alternative behavior, such that its satisfaction could be verified, so that it would relate to the considerations of achieving each goal requirement. Plan defines the certain course to do some action or that has a function to process the activity operation; that is an active component; could represent human, hardware, software, that has a specific role in achieving the goal. It is this element that will monitor any property of the goal requirements, to ensure the changes are related to the achievement of the goal. Resource is an entity (physical or informative), the main concern here is the availability of the resource and identifies the provider if the external party provides it. It is this element that will monitor any property of the goal requirements, to ensure the changes are related to the achievement of the goal.

Figure 1. Model requirements for adaptive cloud computing.
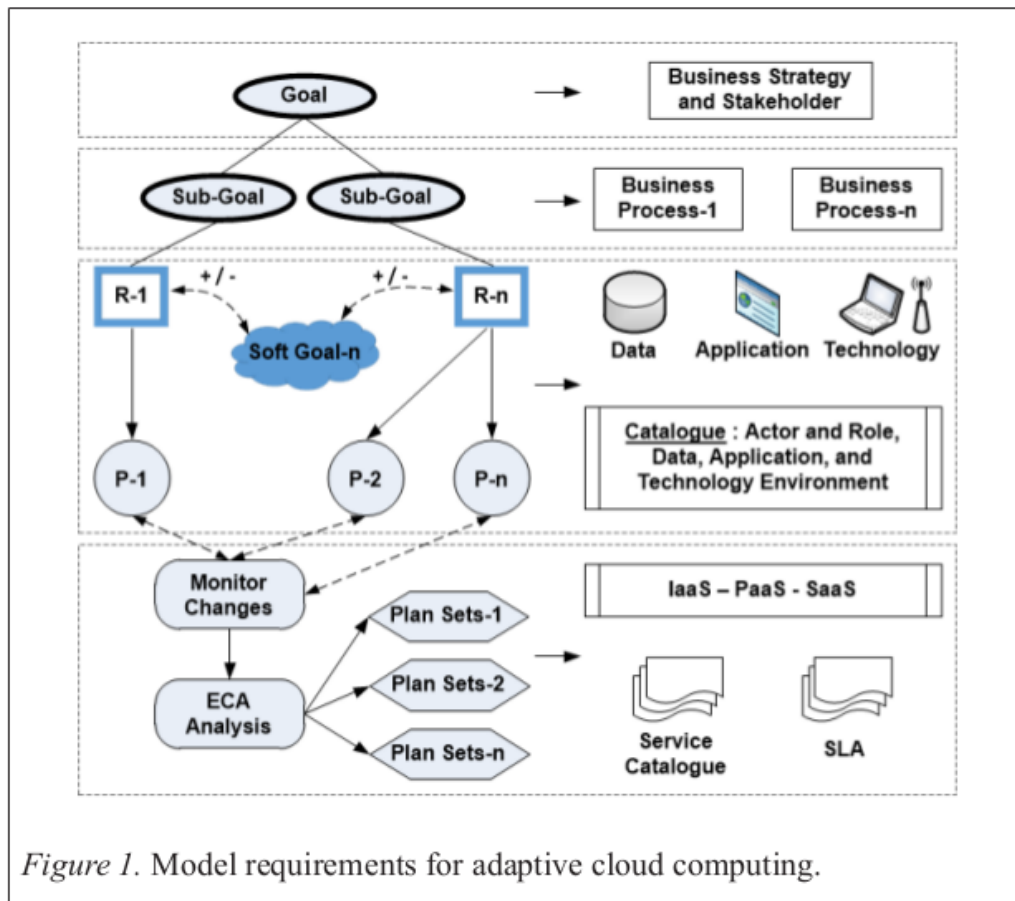
For the GORE basic elements, we map the requirements into three views, namely architectural, alignment, and adaptive. The first view used is the architectural dimension. It is employed to understand the environment which

should be accurately defined. As defined in ISO/IEC 42010, architecture is a fundamental concept or property of a system in the realization of the organization, involving elements, relationships, and principles of its design and evolution. In practice, to realize the view, the available enterprise architectural framework can be adopted. However, the selection of the framework should consider its characteristics and the characteristics of the organization domain.

The next step is the artifact, generated based on the view of the architectural dimension, should align with the characteristics of cloud technology. The integration and standardization ability of the elements involved in the multitenancy concept should be considered. Thus, the mapping of the architectural dimension which has been defined can fulfill the 3 criteria of the cloud computing service model, which are the IaaS, PaaS, and SaaS models.

Based on the theory of the adaptive cloud enterprise architecture (Gill, 2015), adaptive cloud service is an interaction cycle through scan and sense patterns toward its changes and adjustment between contexts (enterprise level) and rationalization (service level). The enterprise level will be related to business architecture, information architecture, and social architecture. On the other hand, the service level is the realization of the need of cloud IT (IaaS, PaaS, dan SaaS). If the theory is mapped toward the views of architecture and alignment, a key view needs to be stated, that is the adaptive view. This view is a medium which can monitor, analyze, plan, and execute adaptation actions of architectural and alignment views. The adaptive is related to the service strategy and management of requirements, so that it can provide plans, execution, coordination, and cloud adaptation management.

In the adaptive view, all decomposition of the functional goal on the views of architechture and alignment is categorized as service catalogue, which consists of business and technical services. Meanwhile, the non-functional goal used as a quality reference for the delivery of cloud service is represented in the form of the service level agreement (SLA). However, according to Kertesz, (2014), the SLA for the environment of cloud computing will involve services in a complex business process, which can hinder the achievement of the SLA. This can be caused by system failure, component changes, work load and external conditions, and hardware and software failure so that the interaction of users and the system during the SLA negotiation and service execution can fail.

Table I

Description Stages Of Requirements

Descriptions of requirements		
View	**Model**	**Description**
View-architecture	Business Architecture: strategy and process	Root goal can be seen as the representation of the goal of business strategy and stakeholder.
	Data Architecture: data entities	Root goal is decomposed into some subgoals which represent the organization functions, so that the goals of data operation, application, and technology can be defined.
	aplication Architecture: Application candidate	
	Technology Architecture: technology infrastructures	
View-alignments	Role and Actor Catalogue: business service, human actor, computer actor	Every functional goal is related to service requirement (R), so that the need of the role and actor catalogue, data catalogue, application catalogue, and the integration of infrastructures and cloud technology can be identified by considering the positive or negative contribution toward the soft goal.
	Data Component Catalogue: separated database, separated schema, shared schema	
	Application Portfolio Catalogue: primary modules, extension modules	Soft goal is a non-functional goal, which is a constraint of service quality, related to reliability, availability, security, etc. This becomes a benchmark for the alignment view in determining the need of cloud service and view adaptability as its adjustment with the need of adaptation.
	Technology Environment Catalogue: platform decomposition, security system	
View-adaptive	Service Catalogue: business service, technical service	Monitoring is conducted on all functional goals (service catalogue) aligned with the IaaS, PaaS, SaaS models.
	Service Level: service level agreements (SLA)	Adaptation mechanism refers to every functional goal realized in the form of the SLA.

In the model that we propose, this problem is anticipated with the adaptation mechanism formulated from the requirements (R-1, R-2, R-n) of the decomposition result of each functional goal, which has a contribution to a non-functional goal. Each variable and parameter (P-1, P-2, P-n) of each identified goal will be monitored; next, the scenario adaptation of the plan is determined (Plan Sets-1, Plan Sets-2, Plan Sets-n).

The method adopted to realize the ability is the event-condition-action (ECA) (Daniele, 2006), which represents the patterns of monitors, analyzers, planners, executors, and knowledge (MAPE-K) (Kephart, 2003). Further explanation of this concept can be found in our previous article (Aradea, 2016). The developed adaptation mechanism is based on the disatisfaction or failure in achieving the goal. This is related tp changes in the SLA, such as service negotiation, failure in achieving the SLA, etc.

B.   Case Study

For the implementation of the developed model, we selected higher education institutions as the domain of the problem. The first requirement formulated was to understand the environment characteristics through the architectural view. In this case, we choose the TOGAF framework (Harrison, 2011) as a model reference. This framework was selected because it belongs to hybrid enterprise architecture and has some characteristics oriented to the development process. Thus, we could expand it using the principle of the GORE approach. Besides that, the framework has a good compatibility level for the development of enterprise architecture and has high completeness/ comprehensiveness to be comprehensively used.

The principle developed from the architectural view is developed into business goal, characteristics, constraint, business process, and system and technology owned by organizations in the academia. The architectural baseline is based on the basics of education in Indonesia, which has a role in the determination of the vision and mission of the higher education institutions, such as the objective of the Ministry of Research and Technology, the vision and mission of the Directorate General of Higher Education, the higher education institutions, and the three principles of higher education institutions.

This can be considered as root goals which can be decomposed into sub-goals for achieving the main goal. Based on the modelling stages, the following system artifacts were obtained, as presented in Table II.

Table 2

Architecture View

Architecture requirements	
Architecture	***Artifact***
Business architecture	5 business functions, 20 business services, 7 human actors, 2 computer actors
Data architecture	20 logical data entity
Application architecture	9 application moduls, 35 logical application components
Technology architecture	Multitenancy, multifactor authentication, bring your own device (BYOD)

The next view used was the alignment view. Every artifact produced based on the alignment view must have the ability of integration and standardization. In this case, we refered to the view of enterprise resource planning integration (Nah, 2001; Nieuwenhuyse, 2011), that was the process integration and planning support as well as decision-making. This system integrated the business functions in an integrated and standardized system, replacing the previous application which was separated and run individually. The results of the evaluation are presented in Table 3.

For realizing the ability of the adaptable view, services which have been defined based on the previous two views can be considered as a strategy of service catalogue management and SLA. The service catalogue consists of the business service catalogue containing all the services related to the primary and extension modules, and the technical catalogue related to all technical specifications of the technology environment. Meanwhile, SLA is a service level agreement between the service provider and the users, so that the sources of changes and failure can be identified based on the SLA which is not fulfilled or violated. Based on a review of the case study context, requirements of the higher education institution services are in the form of service catalogue based on the SLA specifications, as shown in Table 4 (only some of the service specifications are presented). The service catalogue describes the details of the use of the services by each user, and the descriptions of each service specification. The service time explains the length of time used during the service. Options in the catalogue are a category or service class which can be selected based on the level of interest or the effect on the goal (1 = primary, 2 = secondary, 3 = support). The response is part of the activities carried out against the emergence of changes or incidents; the response is classified by the level of seriousness of the incident or problem. The next step was detemining the

cloud service provider which could fulfill every defined service specification. This was done through the implementation of the adaptive view.

Table 3

Alignment view

| Alignment requirements	
No	*Alignment*
1	**Role and actor catalogue**
	New bussines services under the function of the organization resource management are needed, because after the higher education institutions employ the designed system architecture, they need the services which manage the use and administration of the application systems in the institutions and support the parties involved in the business process, particularly in the adaptation of the use of the cloud system .
2	**Data Component Catalogue**
	The multitenant data architecture model is needed. The one used in this study included the approach of the shared database and separate schema. This was based on the flexibility of the database scheme change to follow the database scheme of the client. Besides that, it required less cost with a maintained degree of customization freedom.
3	**Application Portfolio Catalogue**
	Module clarification is needed, that is the primary module (PM) which can only be used optimally when it is run as an integrated module, and extension module (EM) which is supportive/additional. Thus, PM is obtained: academic administration module, system administration module, master administration module, lecturer module, and student module. EM: alumnae module, admision module, finance administration module, utility module.
4	**Technology Environment Application**
	The multitenancy model is needed, it is an application which can be accessed from the Internet by clients who have instantaneous application according to the expected specification. Clients have certain degrees of freedom for customizing the application, but it does not affect the instantaneous application of other clients. Meanwhile, in terms of security, technical planning is required for authentication at tenant level (higher education institutions) to the host or users.

Table 4

Service Catalogue for Higher Education Institution

IT service	Service catalogue								
	Description			User (U)					
	Time	Option	Responds	U_1	U_2	U_3	U_4	U_5	U_n
E-learning	24x7	1	1	√	-	√	√	√	√
Virtual laboratory	24x7	2	2	√	-	√	√	-	-
Office e-mail address	24x7	2	2	√	√	√	√	√	√
Local area network	24x7	1	1	√	√	√	√	√	√
Internet service	24x7	1	1	√	√	√	√	√	√
Registration filestore	24x7	1	1	-	√	-	√	-	-
Faculty filestore	24x7	1	1	√	√	√	√	-	-
Department filestore	24x7	1	1	√	-	-	√	√	√
Student filestore	24x7	1	1	-	-	-	√	-	√
Lecturer filestore	24x7	1	1	-	√	√	√	-	-
Course filestrore	24x7	1	1	-	√	√	√	√	√
Printing service	08-16	2	2	√	-	√	-	√	-
Telephone and fax	08-16	3	3	√	√	-	-	√	-
Course schedule system	24x7	1	1	√	√	√	√	-	√
Course schedule filestore	24x7	1	1	-	√	-	√	-	√

Figure 2 illustrates the adaptation mechanism between the higher education contexts and the contexts of the cloud service providers. It starts with the agent is performance in capturing events from the higher education contexts, consisting of the requirements (req) of business and technical catalogues. The agent also captures the contexts of some service providers offering their services.

Event is information of contexts associated with service specifications; those needed by the higher education institutions and those offered by could-be service providers. The condition for the evaluation is based on certain events, including capturing the service characteristics and changes taking place in every context. Therefore, negotiation of SLA occurs to select the cloud service provider according to the requirements. Finally, based on the most appropriate condition, actions to make decisions are done, including determing the apropriate provider and cancelling contracts, for example, based on the adaptation or the evaluation done to achieve SLA.

11

Figure 2. Adaptation mechanism.
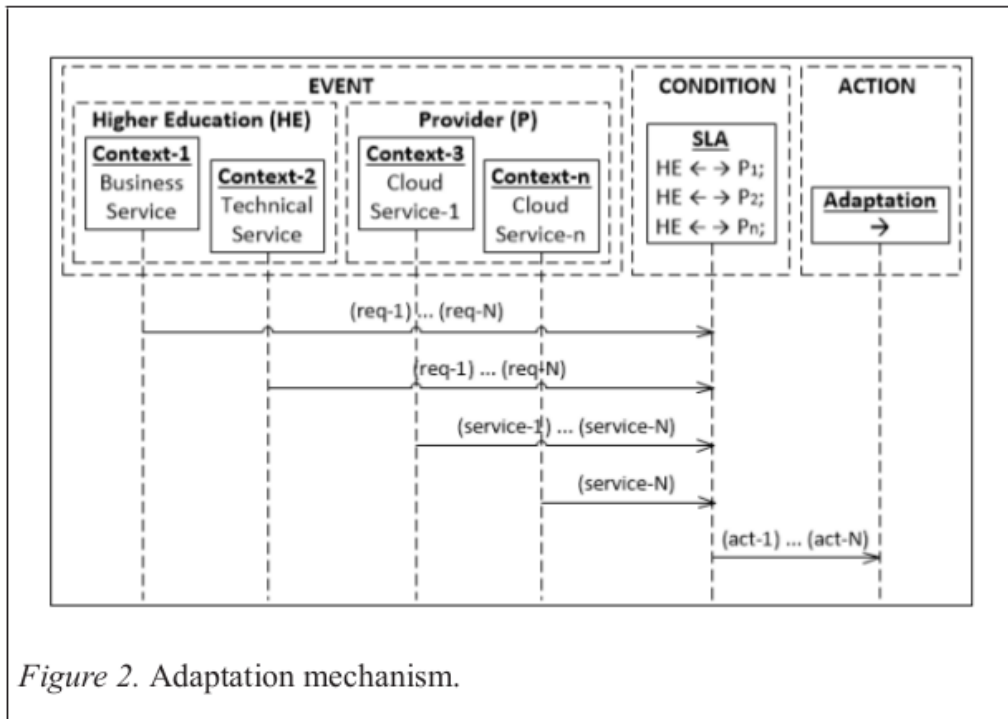
C. Evaluation

Evaluation aims to consider and validate the model which has been designed. The evaluation is done to obtain a conclusion that the proposed model is suitable with particular criteria and controls. The measurement is done by using the standard of the adaptive capability maturity model (Gill, 2015). This standard defines the adaptation maturity model into 5 levels: 0: Infancy, 1: Initial, 2: Transition, 3: Defined, 4: Managed, 5: Adapting. If the proposed model is only applied to one adaptation cycle, for example, the cycle prior to or during migration into the cloud service, the maturity state of the organisation will reach level 4. However, if all cycles are applied, the level 5 target can be achieved. The illustration and description of the scale of adaptation maturity level are shown in Figure 3 and Table 5.
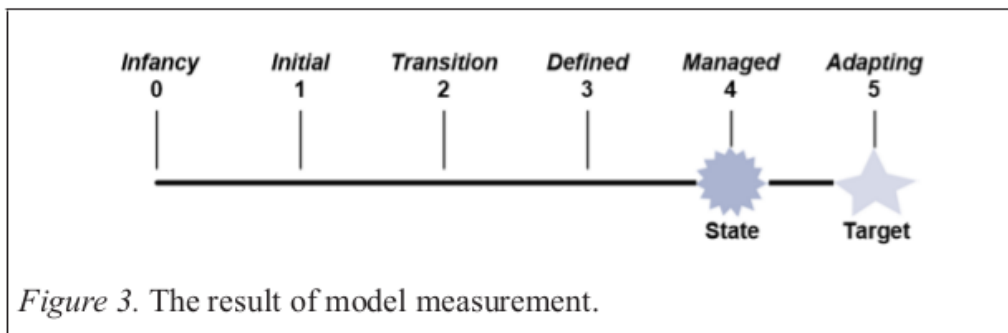

Figure 3. The result of model measurement.

Table 5

Description of Maturity Model

Maturity Model	
Level	**Description**
4 Managed	Changes are effectively managed in an adaptive enterprise architectural ability (cloud) (including architectural artifacts). There are integrated involement and management for the changes. There is a good support for managing changes in an adaptive enterprise architectural ability (cloud).
5 Adapting	Organizations monitor, evaluate, and respond to changes through continuous adaptation of the adaptive enterprise architecture (cloud). There are integrated involvement and management for the adaptation. There is good support for the adaptive enterprise architectural ability (cloud).

The evaluation process is done by mapping the characteristics of the proposed model of the criteria at the level of capability maturity models. Table 6 shows that based on the artifacts obtained in Table 2, the alignment recommendations in Table 3, the service catalog in Table 4, and the mechanisms of adaptation in Figure 2, the characteristics of this model meets the criteria that are between levels 4 and 5.

Table 6

Description of Model Evaluation

Model evaluation	
Maturity criteria	**Proposed model**
Level 1 : Initial • Enterprise is operating ad-hoc local. • formal to support the individual (cloud) projects. • Missing link and alignment between architecture and other capabilities. • Limited support for establishing an adaptive (cloud).	-
Level 2 : Transition • Enterprise has started defining the formal adaptive (cloud). • Limited integrated engagement and governance. • Growing support for establishing an adaptive (cloud).	-

(continued)

Model evaluation

Maturity criteria	Proposed model
Level 3 : Defined	
• Enterprise is effectively operating a formally defined adaptive (cloud).	
• Integrated engagement and governance.	-
• Clear linking and alignment between architecture and other capabilities.	
• Good support for operating a formal adaptive (cloud).	
Level 4 : Managed	
• Changes are effectively managed in the adaptive (cloud).	
• Integrated engagement and governance for handling changes.	√
• Good support for handling changes in the adaptive (cloud).	
Level 5 : Adaptive	
• Enterprise is monitoring, assessing, and responding to changes for continuous adaptations in the adaptive (cloud).	√
• Integrated engagement and governance for adaptations.	
• Good support for the adaptive (cloud).	

DISCUSSION

The cloud adaptive model proposed through the adaptive view based on the description requirements of the architectural view can be applied to three adaptation categories:

1. Pre-migration adaptation into the cloud service – the adaptation process is related to service management done by internal service providers and also related to the authorization of service portfolio, for example, when the service should be renewed, replaced, kept, directfactor, stopped, and rationalized. This is based on the percentage of performance indicators of the SLA achievement. Based on this process, it is decided whether the new or the existing service transition strategies are migrated to the cloud service. If the adaptation process is not met, then the organization will not know which services are to be refurbished and it will be difficult to determine which service providers must be selected.

2. Adaptation when migration is done – the selection of cloud service providers is done based on the need of service portfolio authorization. The consideration done is based on the fulfillment of SLA, that is conducting SLA negotiation, and making or cancelling contracts with vendors providing cloud service. If the adaptation process is ignored,

then the determination of cloud service providers should be done by studying all the agreements offered by the cloud service provider, and we cannot guarantee that the SLA negotiations can, according to the required services;

3. Post-migration adaptation – this is used for the evaluation of performance and maintenance during the implementation or the use of cloud service. Usually, there are some service providers applying a probational period. During this period, the evaluation is done by referring to the SLA achievement so that the decision for continuing or cancelling the contract can be made. If the adaptation process is not done, then we must accept whatever consequences that may occur in the future.

CONCLUSION

The success of an organization in adopting cloud technology depends on the understanding of the organization of its internal environment characteristics and its appropriateness with the characteristics of the service environment offered by service providers. Besides that, the adaptation of the two characteristics should consider the adaptation cycle in the pre-migration, during migration, and in post-migration cycles. If they are are neglected, the application of advanced capabilities offered by a cloud service becomes temporary and costly for the purchase and management of computational resources.

This article descibes the model based on the need. The proposed model provides a holistic view of the activities of the requirements. The architectural view can translate the environment characteristics of organizations through goal decomposition into detailed needs. Meanwhile, the alignment view can meet the implementation, so that the dynamic adaptive ability can be realized through the adaptive view. The mechanism of our proposed model can cover the adaptation needs before the migration (pre-migration), during the migration, and after the migration (post-migration).

ACKNOWLEDGEMENT

REFERENCES

Gill. A.Q. (2015). Adaptive cloud enterprise architecture. *Intelligent Information Systems* Vol. 4. ISBN 978-981-4632-12-6. World Scientific Publishing Co. Pte. Ltd., Singapore.

Lapouchnian, A. (2005). *Goal-oriented requirements engineering - An overview of the current research*. Department of Computer Science, University Of Toronto.

Iankoulova, I., & Daneva, M. (2012). Cloud computing security requirements: A systematic review. *Research Challenges in Information Science* (RCIS), 2012 Sixth International Conference, pages 1-7, IEEE.

Zardari, S., & Bahsoon, R. (2011). Cloud adoption: A goal-oriented requirements engineering approach. *Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing*, pp 29-35, ACM.

Zalazar, A, S., Rodriguez. S., & Ballejos. L. (2015). Handling dynamic requirements in cloud computing. Conference: 44JAIIO – *16th Argentine Symposium on Software Engineering* (ASSE 2015), Rosario, Argentina.

Wind, S., & Schrodl. H. (2011). Requirements engineering for cloud computing: A comparison framework. *Web Information Systems Engineering - WISE 2010 Workshops*, pp 404-415, Springer.

Lamsweerde, et al. (2009). Requirements engineering: From system goals to UML models to software specifications. *Requirements Evaluation*. John Wiley and Sons.

Amanatullah., Y., Lim., C., Ipung, H., & Juliandri, A. (2013). Toward cloud computing reference architecture: Cloud service management perspective. *Proceedings of the International Conference On ICT for Smart Society* (ICISS), Jakarta: IEEE.

Yaish, H., & Madhu, G. (2013). A multi-tenant database architecture design for software applications. IEEE 16th *International Conference on Computational Science and Engineering*, IEEE.

Yu, E. (2009). Social modeling and i*. *Conceptual modeling: Foundations and applications*. Springer.

Hui, B., Liaskos, S., Mylopoulos, J. (2003). Requirements analysis for customizable software: A goals-skills-preferences framework. *Proceedings of International Conference on Requirements Engineering (RE'03)*, Monterey, USA.

Kertesz, A., Kecskemeti, G., Brandic,I. (2014). An interoperable and self-adaptive approach for sla-based service virtualization in heterogeneous cloud environments. *Future Generation Computer Systems*, 32, 54 – 68.

Daniele, L, M. (2006). *Towards a rule-based approach for context-aware applications*. Masters Thesis, University of Cagliari, Italy.

Kephart, J.O., Chess, D.M. (2003). The vision of autonomic computing. IEEE *Computer*, 36(1), 41–50.

Aradea, Supriana I., Surendro, K., & Darmawan, I. (2016). Integration of self-adaptation approach on requirements modeling. *International Conference on Soft Computing and Data Mining (SCDM)*, Springer, August 18-20, Telkom University, Bandung.

Harrison, R. (2011). *TOGAF foundation*. The Open Group.

Nah, F., Fui-Hoon, J., Lee-Shang Lau, & J. Kuang. (2001). Critical factors for successful implementation of enterprise systems. *Business Process Management Journal* 7(3), 285-296.

Nieuwenhuyse, Boeck, Lambrecht, & Vandaele. (2011). Advanced resource planning as decision support module to ERP. *Computers in Industry* 62(1), 1 – 8, DOI:http://dx.doi.org/10.1016/j.compind.2010.05.017

Katsaros, G., Kousiouris, Kyriazis, Menychtas, & Varvarigou. (2011). A self-adaptive hierarchical monitoring mechanism for clouds. *Journal of Systems and Software* 85(5), 1029 – 1041.

Paper 3