Position Paper: Multi-tenants Context-aware Service Composition in Cloud Computing

Wael Sellami, Hatem Hadj Kacem and Ahmed Hadj Kacem
ReDCAD Lab, University of Sfax, Sfax, Tunisia

Keywords: Cloud Service Composition, Context-awareness, Multi-tenancy, Adaptation.

Abstract: Cloud computing has gained a momentum by providing multi-tenancy utility and applications as a service. The power of services occurs in the ability to combine different services in order to obtain a composed service. Context-aware cloud service is anticipated to bring an innovation in mobile computing. Therefore the composed service can be a multi-tenant context-aware, scalable and adaptable. In this paper, we define the problem of coupling context-aware cloud service composition with multi-tenancy in the cloud and we determine crucial challenges that can be solved. For that, we propose an approach to enable multiple providers to run their composed services within the same instance, with different contexts and without modifications of the workflows.

1 INTRODUCTION

Cloud computing emerged as one of the most powerful and flexible software environment. Briefly, NIST (Mell and Grance, 2011) defines cloud computing as a promising and challenging model which delegates the material's management and software applications which users pay as they go. Services in cloud computing are the combination of utility computing services and service oriented architecture in order to provide resources as a service classified into three layers: SaaS (Software as a Service), PaaS (Palteform as a Service) or IaaS (Infrastructure as a Service). In fact, the PaaS layer is a model consisting of elements capable of managing business processes through tools and technologies such as multi-tenancy, scalability/elasticity, data persistence and son on. The multi-tenancy is a key feature in PaaS layer. It means that only one instance of each service is shared by all tenants through virtualization technologies and isolation. With the advent of mobile cloud computing and the proliferation of the customized services, the principle of multi-tenancy can be influenced by context-awareness applications. According to (Dey, 2001), the context includes information that can characterize the situation of a user's local environment and the computing devices themselves. These applications should take into account the situations of tenants context change or applications which may influence their behavior by defining new views on their data and services. Cloud services composition is defined as the process of linking existing software and hardware services. However, it is different from SOA and utility computing in terms of web or grid service form. Therefore, it is hard to use traditional service matching methods by the characteristics of dynamically configured, massively scalable, distributed resources and multi-tenancy architecture of cloud services. Indeed, combining the concepts of context-aware cloud service composition with multi-tenancy is complex and causes new problems. Indeed, applying each technique independently and serially provides a unique view of the composed cloud service for all tenants regardless of their preferences and contexts. Furthermore, adaptation of the composition during sharing between all tenants.

In this position paper, we aim to develop a new middleware for establishing the context-aware cloud service matching and adaptation. Indeed, this middleware must be able to carry out a process of adaptation in order to deliver the most suitable composed service to tenants not only according to their queries but also in relation to their contexts for ensuring a high performance and respecting SLA clauses.

This paper is organized as follows. Section 2 introduces our motivation while Section 3 gives a background of multi-tenant architecture on the cloud. In Section 4, we present the related works. Section 5 discusses our initial approach and Section 6 deals with conclusions and ongoing works.

Sellami W., Hadj Kacem H. and Hadj Kacem A..
Position Paper: Multi-tenants Context-aware Service Composition in Cloud Computing.
DOI: 10.5220/0004406303860389
In *Proceedings of the 3rd International Conference on Cloud Computing and Services Science* (CLOSER-2013), pages 386-389
ISBN: 978-989-8565-52-5

2 MOTIVATION

Many service providers are interested in using the multi-tenancy, scalability and other benefits of the cloud (Paraiso et al., 2012). With the rapid development of mobile cloud computing, context-aware information (such as personal, environmental and operational contexts) can satisfy tenants with their really needed composed services without any explicit requests. However, conflicts can be generated by assembling the multi-tenancy with the service composition context-aware. Indeed, there are antagonistic concepts since the aim of multi-tenancy is the ability to use the same application to multiple tenants. While the purpose of the context-aware composition is to suit a single application for multiple tenants with different contexts and preferences. To understand the problem well, we can take two scenarios of multi-tenancy and context-aware service composition:

A. Multi-tenancy

- **Scenario.** Executing a composed cloud service for products research into on-line market.
- **Result.** Sharing the composed services instance to provide all images and descriptions of all the products sorted out by category for all tenants.

B. Context-aware Service Composition

- **Scenario.** Executing the same composed cloud services for products research into on-line market from a mobile phone.
- **Result.** Providing product description only with women's preferences.

Towards the two scenarios, the instance of composed cloud services has the same behavior and all the used data are shared between all tenants regardless of specific contexts and requirements for each tenant. So, this challenge leads us to ask a question:

How can we realize a multi-behavior instance of a composed service for all tenants ?

Combining the concept of multi-tenancy with context-aware cloud service composition to establish new compromise middlware is complex and provides new and interesting settings for both issues.

3 A CONCEPT FOR MULTI TENANCY ARCHITECTURE

In cloud computing, multi-tenant architecture is the ability of hosting multiple users under a single server and sharing the same infrastructure and applications (Domingo et al., 2010). A multi-tenant application must support multiple tenants which can have multiple users and user group preferences. In order to meet the different tenant requirements and the variation context, services must be highly customizable and configurable. So, service providers have to offer certain level of variability in their multi-tenants application, and they should be aware of the level of the variability, in terms of its complexity impact on service flexibilities. From our perspective, we think that the balance between composed cloud services and the variability of context has to be well maintained. This can be done by integrating service facilities on a per tenant (as it is shown in Figure 1) and modifying the system behavior to suit the tenant responses.

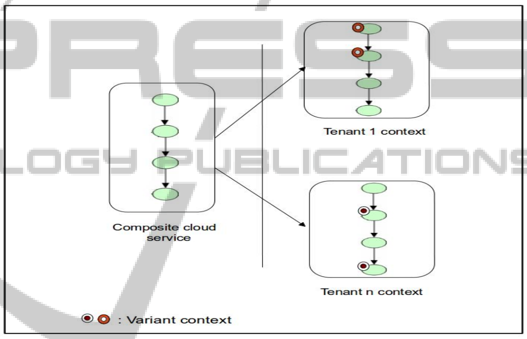

Figure 1: Variation points of multi-tenant composite service.

4 RELATED WORKS

The problem of cloud service matching and adaptation has received a lot of attention in academia and industry in the recent years. In this section, we present in the first part service composition approaches and in the second part we give an overview of services adaptation.

4.1 Cloud Service Composition

A new approach for services composition is presented in (Tsai et al., 2011). This work is based on the mechanism of service selection using service interfaces and service injection techniques. Thus, it proposes a series of tests to check services and ensure the reliability of composition. In (Ye et al., 2011), authors propose a multi-cloud service composition approach. Users requirement is translated into data flow graphs. Then, by using a genetic algorithm, a service composition solution is provided. Furthermore, in (Kofler et al., 2010), authors determine a heuristic based approach

to compose services in cloud computing. They focus on user's functional and non-functional requirements. Thus, an algorithm is applied to automate the selection of optimal services and to realize service composition Another approach to build a decentralized framework based on self organized agents is determined in (Garcia and Sim, 2010). Authors used contract net protocol for dynamic communications technical knowledge of acquintance networks for incomplete information about existing services. In (Zeng et al., 2009), authors solve the problem of cloud service composition using a service matching algorithm (SMA). It is based on QoS to filter services and perform the matching algorithm of services based on the semantic similarity between input and output parameters to achieve this composition.
We notice that these works do not provide appropriate support for the concept of context and progress scenarios during cloud services searching and matching. Moreover, they do not retain the concept of the cloud which is the multi-tenancy nature during composition where a single instance is shared between tenants.

4.2 Cloud Service Adaptation

The approach presented in (Truyen et al., 2012) describes a framework for tenant-specific customization and manages changes of software-tenant SaaS applications. To achieve this, this framework is based on the context oriented programming to provide flexible SaaS services for multi-tenants suitable to the context information. In (Papakos et al., 2010), authors develop a middleware called VOLARE which monitors resources and context of mobile devices, and dynamically suits the cloud services. This approach allows a reliable services discovery and cost reduction at runtime to fit in the current context of the consumers. In (La and Kim, 2010), authors propose a framework for context-awareness services for mobile computing. In fact, this framework allows the capture of the context, determines the specific adaptation and executes the right services.
However, these efforts focus only on the customization of a single service but no solution has been proposed to resolve the adaptation of composed cloud service. Also, they deal with the personalization of responses at a high level but not the personalization of services at a lower level before the composition to provide more flexibility of applications.

To overcome these issues, we propose in this paper a primal tenant cloud service composition middleware to address all aspects of the context awareness starting from the dynamic services matching until to adaptation.

5 INITIAL APPROACH

In this section we present, in detail, our multi-tenants context-aware cloud service composition middleware. Indeed (as it is shown in Figure 2), we aim to focus on two steps of the life cycle: (1) services matching and (2) adaptation of a composed service. Each step contains components, and their related communications.

5.1 Tenant Users

Tenant users layer is composed of multiple tenant applications deployed on portable multimedia player and mobile device such as Android or SmartPhone. Thus, we consider that a tenant can be a single user or a whole company consisting of a collection of distinct users which can access to his flexible services whenever he wants and wherever he is.

5.2 Service Matching

A composition engine is a multi-tenant service executing on the cloud platform. It parses the tenant query and uses the contexts information. Then, it can be able to lookup its cloud services from repositories component using a given policy (such as the nearest data center to tenant location, the fastest response if the tenant is running, and so on) and choose the most appropriate composition that suits the tenant's context. In fact, once the services are selected, the composition engine is responsible for defining a sequence of services to be executed and supervising the composed service. The composition engine can decide which concrete services are included in the dynamic composition, and how many instances of each concrete service are needed in order to satisfy the tenant's context. Therefore, this component can dynamically update the composition and substitute any concrete service at any time with another service based on the feedback of the tenant's runtime context data.

5.3 Service Adaptation

In context-aware composite service provisioning, adaptation can be achieved by splitting it into separate services and tailoring them for the given context. Tenant specific customization can be specified as independent software variations according to some categories of context such as tenant profile, environment information and so on. For that, we propose the runtime behavior adaptation component which is called tenant context adaptor in order to suit the composition, by retrieving context information from the Context Base registry, and updating it to new situations.

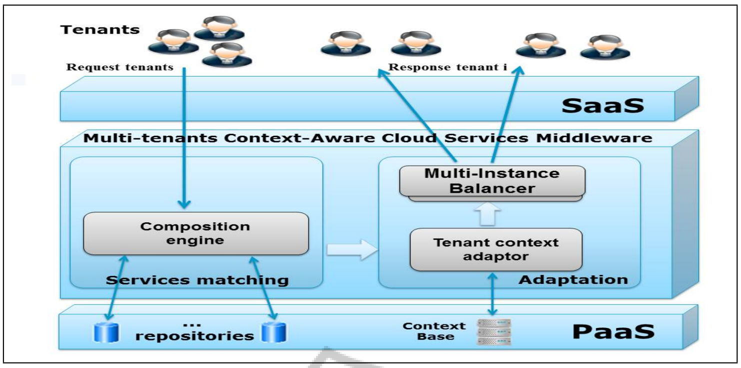

Figure 2: Multi-tenants context-aware cloud service composition middleware.

This adaptation is used to integrate new behaviors and concerns as persistence or security into the original cloud service without affecting other applications that share the same service with other tenants. After that, the multi-instance balancer component distributes the instance for each tenant according to his global context and to satisfy his requirements.

6 CONCLUSIONS AND ONGOING WORK

Due to the open and flexible nature of cloud services, there is a gap between service composition efficiency and the tenants flexible response needs. In this paper, a context-aware cloud service composition middleware is investigated for improving the services matching efficiency in mobile cloud environment. Concretely, this middleware permits dynamic context-aware composition and adaptation of flexible cloud based services, and how these characteristics influence the multi-tenancy adaptation. In the ongoing work, we are developing a middleware and are defining the experimental techniques to simulate cloud service composition according to the context-aware. Thus, we are aiming to exploit new composition scenarios in a real environment to analyze the middleware efficiency. For future work, we will detail each component of our middlware and propose efficient techniques for implementing them. Moreover, we will enhance the adaptation of the tenant composite service with other contextual situations.

REFERENCES

Dey, A. (2001). Understanding and using context. *Journal of Pervasive and Ubiquitous Computing*.

Domingo, E., Nino, J., Lemos, A., Lemos, M., Palacios, R., and Berbs, J. (2010). Cloudio: A cloud computing-oriented multi-tenant architecture for business information systems. *IEEE 3rd International Conference on Cloud Computing*.

Garcia, J. and Sim, K. (2010). Self-organizing agents for service composition in cloud computing. *International Conference on Cloud Computing Technology and Science*.

Kofler, K., Haq, I., and Schikuta, E. (2010). User-centric, heuristic optimization of service composition in clouds. *International Euro-Par conference on Parallel processing*.

La, H. and Kim, S. (2010). A conceptual framework for provisioning context-aware mobile cloud services. *International Conference on Cloud Computing*.

Mell, P. and Grance, T. (2011). The nist definition of cloud computing.

Papakos, P., Capra, L., and Rosenblum, D. S. (2010). Volare: Context-aware adaptive cloud service discovery for mobile systems. *9th Workshop on Adaptive and reflective Middleware*.

Paraiso, F., Haderer, N., Merle, P., Rouvoy, R., and Seinturier, L. (2012). A federatedmulti-cloud paas infrastructure. *IEEE 5th International Conference on Cloud Computing (CLOUD)*.

Truyen, E., Cardozo, N., Walraven, S., Vallejos, J., Bainomugisha, E., Gunther, S., D'Hondt, T., and Joosen, W. (2012). Context-oriented programming for customizable saas applications. *27th Annual ACM Symposium on Applied Computing*.

Tsai, W., Zhong, P., Balasooriya, J., Chen, Y., Bai, X., and Elston, J. (2011). An approach for service composition and testing for cloud computing. *International Symposium on Autonomous Decentralized Systems*.

Ye, Z., Zhou, X., and Bouguettaya, A. (2011). Genetic algorithm based qos-aware service compositions in cloud computing. *International conference on Database systems for advanced applications*.

Zeng, C., Guo, X., Ou, W., and Han, D. (2009). Cloud computing service composition and search based on semantic. *International Conference on Cloud Computing*.