Economic Model Based Cloud Service Composition

Zhen Ye[1,2]
Supervised by: Athman Bouguettaya[3] and Xiaofang Zhou[1]

[1] The University of Queensland, Australia
[2] CSIRO ICT Centre, Australia
[3] Royal Melbourne Institute of Technology, Australia

Abstract. Cloud service composition is usually long-term based and driven by meeting the Cloud economic model. We use CP-Net to represent the long-term based economic model. We consider service composition as a Nash game to capture the behaviors of Cloud application providers and IaaS providers. Finally, we propose an economic model based service composition approach in Cloud computing.

1 Introduction

Cloud computing is becoming increasingly popular as the next-generation platform for conducting business. The main advantage of Cloud computing is its ability to scale to users' demands for storage, CPU and network resources [1]. The enabling technology for cloud computing is *Service Oriented Computing* (SOC)[2]. Cloud computing has *Cloud Providers* and *Cloud Consumers* like other traditional service-oriented platforms, such as Web service technology, Grid computing. Cloud service consumers are mostly *Cloud Application Providers*[1]. These consumers supply *Cloud Composite Applications* to end users. Cloud applications are mostly long-lived scalable applications subject to time-varying workload.

Compared to the traditional platforms, Cloud computing makes services more efficient and cost-effective for both providers and customers. Cloud service consumers, on one hand, save their cost by increasing and reducing resources at will. Cloud service providers, on the other hand, maximize the utilization of their resources and profits from the benefits of economics of scale. Indeed, Cloud computing introduces a new *Economic Model* for both consumers and providers.

We assume that the service infrastructure in Cloud computing is managed by a *Service Management System* [3]. The management system includes service description, service query and service composition etc. We focus on *Cloud Service Composition.* Service composition is the process of grouping/mashing up a set of services to provide a value added service. The result of this process is a composite service. This meets both the functional and non-functional (i.e., Quality of Service or QoS) requirements of consumers.

[1] In the remainder of the paper, we use *Cloud Consumer* and *Cloud Application Provider* interchangeably.

G. Pallis et al. (Eds.): ICSOC 2011, LNCS 7221, pp. 183–188, 2012.

Service composition in traditional platforms (e.g. SOC, Grid Computing) mostly assume it is short-term based and driven by meeting QoS at time of service query. In contrast, Cloud service composition is long-term based and driven by meeting the requirements of the economic model. In the Cloud economic model, computing resources, platforms and applications are virtualized as trading services in the market. Cloud services in the market are charged in a pay-as-you-go manner. Each Cloud application provider wants to maximise its profit while complying with QoS requirements, specified in Service Level Agreements (SLAs) with end users. Cloud service providers, in their turn, want to maximise the revenues obtained from providing the resources. We propose to use CP-Nets to represent the long-term based economic model for both Cloud application providers and Cloud service providers. Each Cloud application provider competes with others and bids for the use of Cloud services. The Cloud service provider competes with other Cloud service providers for obtaining more valuable consumers. In these competitions, the best choice of one depends on the choices of others. Therefore, we model both Cloud application providers and Cloud service providers as independent economic agents in a *Nash Game*. All players in the game attempt to selfishly optimise their utility. We then use the competition results to develop an efficient algorithm for Cloud service composition.

2 Composing Cloud Services

Cloud services refer to both the applications delivered as services over the Internet and the hardware and system software in the data centers that provide those services [2]. Cloud services are classified into three types: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS).

For the sake of simplicity, we make several assumptions about Cloud services: 1) We only consider three types of IaaS during service composition. They are CPU services, network services and storage services. 2) Each SaaS provider is supported by a single PaaS provider. 3) Each PaaS provider is supported by a single IaaS provider. 4) Each IaaS provider supplies IaaS to multiple PaaS providers. 5) Each PaaS provider supplies PaaS to multiple SaaS providers. Cloud service composition, therefore, consists of compositions in three layers: SaaS composition, PaaS composition and IaaS composition. PaaS providers and IaaS providers are determined whenever a certain SaaS is selected for composing a composite service. We focus on SaaS composition and IaaS composition in this research.

2.1 Motivating Scenario

We use the scenario depicted in Fig. 1 to motivate and explain our approach. We consider an evaluation application. Universities, acting as Cloud application providers, supply evaluation applications to end users (e.g. tenure committee members etc.). Based on the request workload on each university, the composition system aims to compose component SaaSs to form the evaluation composite applications. Each component SaaS further expects CPU, network and storage resources from IaaS providers.

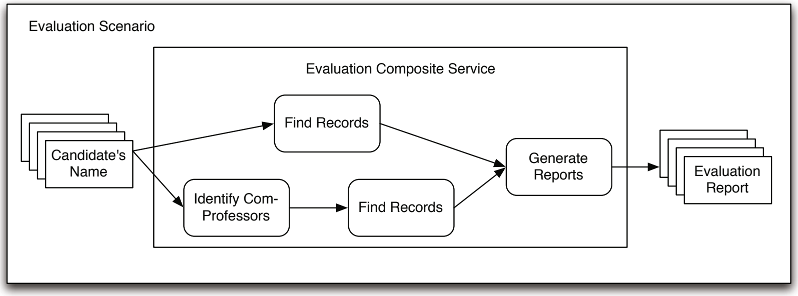

Fig. 1. Evaluation Cloud Service

2.2 Economic Model

The main part of the economic model is a representation of the long term based non-functional economic requirements. We use Conditional Preference Network (CP-Net) [4] to describe Cloud consumers' and Cloud providers' long term preferences over the non-functional properties.

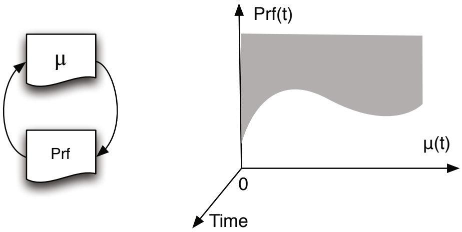

Fig. 2. Economic Model for a University

The economic model of a Cloud application provider depends on the SLAs established between the Cloud application provider and its end users. Each Cloud application provider usually serves different types of end users. The requests from different user types normally demand different composite applications both in functional and non-functional properties. Fig. 2 shows the economic model for a university in the example at any time t. The economic model includes two QoS attributes: $\mu_i(t)$ and $Prf_i(t)$. $\mu_i(t)$ represents the mean service rate of requests in university U_i at time t. $Prf_i(t)$ represents the total profit for university U_i from time t to $t + dt$. The universities always prefer larger $\mu(t)$ and $Prf(t)$. However, a larger $\mu(t)$ does not always lead to a larger $Prf(t)$. Hence, the university needs to clarify the satisfactory combinations of μ and $Prof$. The gray area in Fig. 2 represents the tolerant values for combination $[\mu(t), prf(t)]$ at time t.

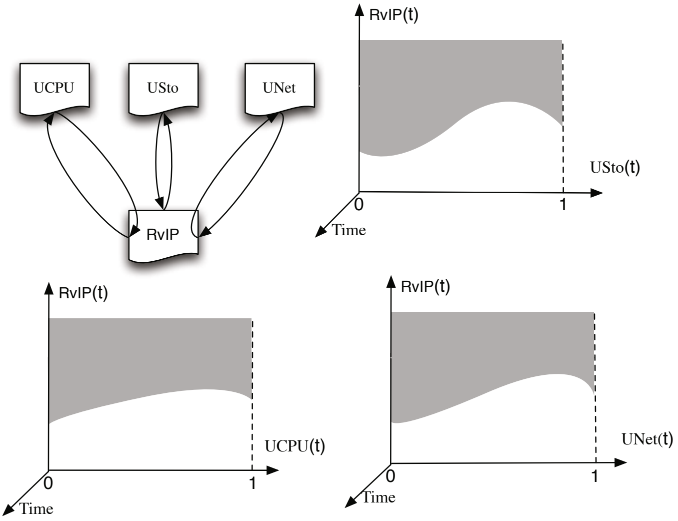

Fig. 3. Economic Model for an IaaS Provider

The economic model of an IaaS provider includes the non-functional requirements on the three types of Cloud services. SaaS obtain the three types resources, i.e., CPU, storage and network resources from the Cloud service provider in three ways [5]. Each of them defines a particular service market, as follows: 1) On-demand Market. This market allows resources to be supplied on demand at any time. The user is charged a usage fee, which is according to the time each resource is used. 2) Reservation Market. This market allows Cloud resources to be reserved for long periods of usage. It requires the payment of an upfront one time reservation fee. 3) Spot Market. Resources in this markets are offered in a best-effort way. Users access resources in this market by indicating the maximum usage fee that they are willing to pay. Fig. 3 shows the economic model for an IaaS provider in the example at any time t. The economic model includes four QoS attributes: utilization of CPU resources, storage resources and network resources, and the revenue. $UCPU_j(t)$, $USto_j(t)$ and $UNet_j(t)$ represents the utilization of CPU resources, storage resources and network resources in IaaS provider IP_j at time t. $RvIP_j(t)$ is the total revenue of IaaS provider IP_j from time t to $t + dt$. The IaaS provider aims to have higher utilization of its Cloud services, and higher revenue. Whereas, a higher utilization does not always lead to a higher revenue. The gray area in Fig. 3 represents the tolerant value combinations:$[UCPU(t), USto(t), UNet(t), RvIP(t)]$ at time t.

In the motivating scenario, each university would change its selection of SaaS and the unit number of IaaS according to its request workload. We assume that IaaS providers charge SaaS in terms of hourly basis [5]. Hence, each university faces the problem to satisfy its economic model by determining every hour the optimal component SaaS and the unit number of IaaS from each market. In addition, each IaaS provider's revenue and utilization of resources would vary by changing the prices of its Cloud services. Each IaaS provider, therefore, faces the problem to satisfy its economic model by determining every hour the time unit price for reservation, on-demand resources and spot resources. The ultimate goal of the composition system is to satisfy the economic model for the Cloud system.

2.3 Cloud Service Composition Approach

In a Cloud system, the goal of a Cloud application provider is to determine every hour the number of reservation, on-demand and spot resources to execute its composite services to maximise its profit. Moreover, the goal of an IaaS provider is to determine the time unit cost for a unit number of on-demand, reservation and spot IaaS resources to maximise its total revenue. In this framework, Cloud application providers and IaaS providers are making decisions at the same time. The decisions of a Cloud application provider depend on those of other Cloud application providers and IaaS providers. Vice versa, the IaaS objective function depends on application providers' decisions and other IaaS providers.

We model Cloud service composition as a *Nash Game* [6] where Cloud application providers and Cloud service providers are players in the game. In the Nash game, a set of strategies for the players constitute a *Nash Equilibrium* if no player can benefit by changing its strategy while other players keep their strategies unchanged. Following this Nash equilibrium concept [7], Cloud application providers and IaaS providers adopt a strategy such that none of them can improve its revenue by changing its strategy unilaterally. At the end of this competition (game), there might exist more than one equilibrium. The equilibrium which leads to a Cloud system that satisfies the economic model are the final solution for the Cloud service composition problem.

3 Related Work and Summary

Most composition approaches in traditional platforms (e.g. SOC) use linear programming methods. Two composition approaches are proposed in [8]. One of them focuses on local optimization. The other approach focuses on global optimization. They use integer programming to solve the optimization problem. Ardagna et al. [9] further propose an improved approach. This approach uses Mixed Linear Programming (MILP). They also introduce several concepts such as loop peeling and negotiation mechanisms to address situation where no feasible solution can be found. Alrifai and Risse [10] propose an approach to decompose global QoS constraints into local constraints with conservative upper and lower bounds. These local constraints are resolved by using an efficient distributed local selection strategy. Recently, several approaches have been proposed to solve service-related problems in Cloud Computing. Wu et al. [11] propose a dynamic workflow-based resource allocation approach to enhance the overall performance of composite services in Cloud computing. Ardagna et al. [6] model the service provisioning problem as a generalized Nash game, and proposes an efficient algorithm for the run time management and allocation of IaaS resources to competing SaaSs. However, the problem of service composition has not been considered in Cloud computing platform.

We adapt an economic model based approach to solve Cloud service composition problems. Compared to traditional service composition approaches, our work leverages two difficulties when considering service composition in Cloud computing. Cloud service composition is driven by meeting the Cloud economic model. In addition, Cloud service composition is usually long-term based. Hence, we adopt CP-Net to represent the long-term based economic model. Service composition needs to be conducted every hour to meet the changes. Furthermore, We consider service composition as a Nash game to capture the behaviors of Cloud application providers and IaaS providers. We finally use the results to develop an efficient algorithm for Cloud service composition.

References

1. Motahari-Nezhad, H., Stephenson, B., Singhal, S.: Outsourcing Business to Cloud Computing Services: Opportunities and Challenges. LABs of HP (2009)
2. Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., et al.: Above the clouds: A berkeley view of cloud computing. EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28 (2009)
3. Yu, Q., Liu, X., Bouguettaya, A., Medjahed, B.: Deploying and managing web services: issues, solutions, and directions. The VLDB Journal The International Journal on Very Large Data Bases 17(3), 537–572 (2008)
4. Boutilier, C., Brafman, R., Domshlak, C., Hoos, H., Poole, D.: Cp-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. Journal of Artificial Intelligence Research 21(1), 135–191 (2004)
5. Amazon elastic compute cloud pricing (2011), `http://aws.amazon.com/ec2/pricing/`
6. Ardagna, D., Panicucci, B., Passacantando, M.: A game theoretic formulation of the service provisioning problem in cloud systems. In: Proceedings of the 20th International Conference on World Wide Web, pp. 177–186. ACM (2011)
7. Facchinei, F., Kanzow, C.: Generalized nash equilibrium problems. 4OR: A Quarterly Journal of Operations Research 5(3), 173–210 (2007)
8. Zeng, L., Benatallah, B., Ngu, A., Dumas, M., Kalagnanam, J., Chang, H.: QoS-aware middleware for web services composition. IEEE Transactions on Software Engineering 30(5), 311–327 (2004)
9. Ardagna, D., Pernici, B.: Adaptive service composition in flexible processes. IEEE Transactions on Software Engineering, 369–384 (2007)
10. Alrifai, M., Risse, T.: Combining global optimization with local selection for efficient qos-aware service composition. In: Proceedings of the 18th International Conference on World Wide Web, pp. 881–890. ACM (2009)
11. Wu, B., Chi, C., Chen, Z., Gu, M., Sun, J.: Workflow-based resource allocation to optimize overall performance of composite services. Future Generation Computer Systems 25(3), 199–212 (2009)