A Cloud Agency for SLA Negotiation and Management

Salvatore Venticinque[1], Rocco Aversa[1], Beniamino Di Martino[1], Massimilano Rak[1], and Dana Petcu[2]

[1] Second University of Naples, Italy,
[2] Institute e-Austria Timişoara, Romania

Abstract. Resources management facilities, based on service level agreements, are needed in the Cloud in order to negotiate a collection of inter-connected and virtualized computers between resource providers and consumers. In this paper we present the architectural design of a system named `Cloud Agency` which aims to respond to this need and to offer added value to the existing Cloud services. This system is in charge to broker the collection of Cloud resources from different providers that fulfills at the best the requirements of user's applications. The user is able to delegate to the Agency the necessary checks of the agreement fulfilment, the monitoring of resource utilization and eventually necessary re-negotiations.

1 Introduction

Cloud computing is an emerging paradigm that, due to an intensive use of the virtualization approach, offers to users resources on which they have full administrative control. Cloud computing is expected to be the paradigm that will deliver a basic level of computing service that is considered essential to meet the everyday needs of the general community [1]. Such a computing utility is targeted to a market of consumers who require specific QoS to be maintained by their providers in order to meet their objectives and sustain their operations. In this context the need of SLA-oriented resource management represents the solution to negotiate a collection of inter-connected and virtualized computers between resource providers and consumers (or between resource providers and a third-party broker) [2]. The selection of Cloud providers that fulfills the requirements of a particular application is a complex issue due to the different business models associated with such computing systems. Cloud providers usually employ a system-centric resource management architecture. According to [1] a market-oriented resource management is needed in order to regulate the supply and demand of Cloud resources, providing feedback in terms of economic incentives for both Cloud consumers and providers, and promoting QoS-based resource allocation mechanisms that differentiate service requests based on their utility. The current Cloud computing technologies offer a limited support for dynamic negotiation of SLAs between participants. There are no mechanisms for automatic allocation of resources to multiple competing requests. Furthermore,

M.R. Guarracino et al. (Eds.): Euro-Par 2010 Workshops, LNCS 6586, pp. 587–594, 2011.

current Cloud computing technologies are not able to support customer-driven service management based on customer profiles and requested service requirements. Also it is impossible according to [1] to derive appropriate market-based resource management strategies that encompass both customer-driven service management and computational risk management to sustain SLA-oriented resource allocation. New SLA-oriented resource management strategies must be designed for Clouds in order to provide personalized attention to customers. Service requirements of users can change over time, due to continuing changes in business operations and operating environment, and thus may require amendments of original service requests. We proposed recently in the frame of the EC-FP7-ICT project proposal, named mOSAIC, a solution, based on software agents and semantic data processing, for Cloud resource negotiations and service level agreements. The mOSAIC project (www.mosaic-cloud.eu) intends to improve the state-of-the-art in Cloud computing by creating, promoting and exploiting an open-source Cloud application programming interface and a platform targeted for developing multi-Cloud oriented applications. The main benefit of using the mOSAIC software package will be a transparent and simple access to heterogeneous Cloud computing resources and the avoidance of lock-in proprietary solutions. A special attention will be given to the applications that are data-intensive: the Earth Observation community is strongly involved in the platform testing. In this paper we present an important component of mOSAIC framework, we named `Cloud Agency`, which aims at offering value added Cloud services. It will be in charge to broker a collection of Cloud resources from different providers that fulfills at the best the requirements of user's applications. According to the available offers it will generate a SLA document that represents the result of resource negotiation and booking with supported Cloud providers. The user will be able to delegate to the Agency the necessary checks of SLA fulfilment, the monitoring of resource utilization and eventually necessary re-negotiations. The paper is organized as follows. The second section discusses the motivation of our research and the state of art of Cloud dealing with the resource brokering. The third section introduces the requirements for Cloud Agency design. In the fourth section the Cloud Agency design is described. Finally, in the last section, some conclusion are provided.

2 Requirements for SLA Negotiation and Management

The mOSAIC team plans to investigate Cloud-specific SLAs and QoS requirements in order to support resources management. The proposed approach is to start from Cloud usage patterns. Such patterns were recently identified in [3], reflecting a business view.

Definition of QoS parameters. One of the preliminary requirements, which are relevant to support negotiation activities into the Cloud, is the definition of QoS parameters for existing service. Of course this can be done after an exhaustive study of available Cloud platforms and services. There are critical QoS parameters to consider in a service request for Cloud computing, such as for instance:

time, cost, computer power, storage size, reliability, trust, security, or even location of resources due to business constraints. An attempt to define several QoS metrics is presented in [4]. Authors define response time, availability, reliability, cost and reputation. A reference of SLA model is provided in [5], where SLA objectives (SLOs) are used to compose a SLA. The existence of a number of service levels and performance metrics for each resource results in multiple SLOs for every service.

Role of users, brokers and providers. As actors of the Cloud market needs to be defined. Users submit service requests from anywhere in the world to the Cloud. Cloud providers offer resources, allocate the one acquired and bill their consumption. They need to control that there is no overloading of resources whereby many service requests cannot be successfully fulfilled. This leads the decision on whether to accept or reject the request. On the other hand users need to be aware about the resources they are really exploiting and the service level they are provided with at any time. The monitoring of QoS level is relevant to detect SLA violation which can be regulated by penalties that providers must pay. Applications and new services have to be designed in order to let user delegate to applications the automatic negotiation and management of SLA, resources and services on behalf of users and providers.

SLA negotiation and renegotiation. SLA negotiation with multiple Cloud providers is a first example of complex application that could be delegated to a third party, represented by a broker in a market based context. A broker intermediates between users and providers in order to negotiate the best SLA for both consumer and vendors. On user behalf it can:

- search for available Cloud services, compliant with user needs;
- check of trustness of providers;
- decide with whom to negotiate, according to user requirements and past experiences;
- negotiate the best price for the same offer by different providers;
- negotiate multiple SLAs, with different providers, to overcome the lackness of one compliant offer by a single provider.

Since consumers' requirements can potentially vary over time it needs to support dynamic re-negotiation of SLA. Some mechanisms to reconfigure virtual resources are already available, but it needs policies and protocols for changing the SLA parameters, to include new amendments and withdraw previous ones. Re-negotiation is another service that can be provided to solve some inconsistencies between the SLA and the real user's requirements which can change dynamically. Dynamic SLA re-negotiation has actually limited support. Issues to be investigated are:

- withdraw of a SLA and negotiation of a second one;
- deletion/addittion of a SLA objective;
- redefinition of a QOS parameter;
- negotiation of boundaries within which the SLA can be re-negotiated at the same price or with a pre-defined price adjustment.

Monitoring. The utilization of Cloud resources is another service that can be delegated. Providers monitor utilization of their resources for billing, to change bid prices in order to optimize profit, to not exceed in resource allocation beyond the capability of fulfill the agreements. On the other hand, the user, who has conflicting interests with providers, needs to trust a third party that can be delegated to monitor the satisfaction of the agreed service levels. Monitoring process should provide information about:

– under-utilization of cloud resources, in order to negotiate cheaper agreements;
– saturation of resources, to not let the users's applications work under the QoS level granted to users' clients;
– unbalanced utilization of Cloud resources, in order to check the correctness of negotiated parameters, or to tune the execution of applications in the Cloud;
– violation of SLA by providers.

3 A Cloud Agency in mOSAIC

Applications for SLA negotiation and management should act on behalf of their users and should be able to compose available Cloud services. They will be proactive applications that, beyond the stateless SOA model, are aware about the status of their user's resources and services and interact with brokers, providers and eventually with other Cloud actors in order to pursue user's objectives. Because of these considerations, we modeled the services as agents who implement a *Cloud Agency* [6] in a framework that aims at deliver and manage Cloud resources and services provided by different Cloud platforms.

3.1 Architecture and Agents' Role

The Cloud Agency architecture is showed in Figure 1. The main service provided by the Agency is the negotiation of Cloud resources. The core agents are enumerated in Table 1. A *Client Agent* acts as an access point for the user who is exploiting proactive services. It maintains the user profile and cooperates with the Negotiator in order to provide to the user the services with the requested quality levels. A *Mediator Agent* retrieves a list of available Provider Agents from the *Registry Agent*. It contacts each Provider Agent and requests a bid for the needed resources. Once it obtains responses from Provider Agents, it assesses the following: the QoS provided; the quality of the provider itself (requesting historical data from an Archiver Agent); after assessing the bid responses, it should put together a contract with the winning providers on behalf of the client; it replies to the Client Agent with the attached contract. The Negotiation Agent could try to optimize the contract by applying different trade-offs between performance or availability and costs, but within the bounds specified by the client. A *Provider Agent* accepts bid requests from the Mediator Agent, and tries to propose a contract for the resources it could provide. A *Registry Agent* will allow

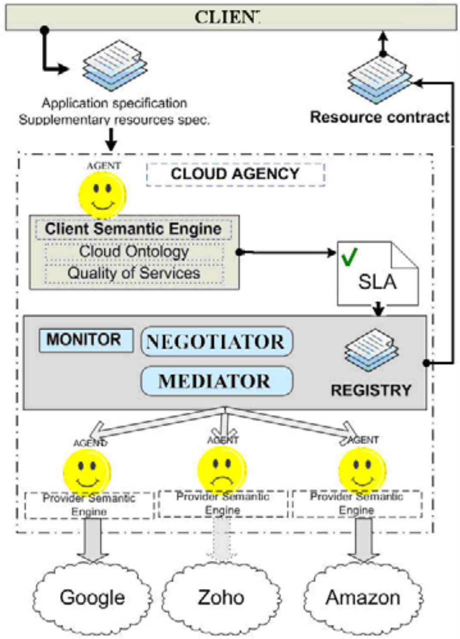

Fig. 1. The Cloud Agency Architecture

the publication of services which are available and accessible by the mOSAIC framework and their discovery. An important project activity includes an exhaustive study of the most important Cloud platforms, in order to regognize what services they can provide and how to use them. Agents exploit a *Semantic Engine* that will allow also semantic validation and translation of messages that use different ontologies and will refine the application specification document into a correct and complete SLA on behalf of the user. The monitoring of Cloud resources utilization is necessary to evaluate the satisfaction of SLA and the effective utilization of resources by the user's applications. A *Monitor Agent* will be in charge to collect all the available information from Cloud providers and from the user's application themselves to figure out the effective values of QoS parameters and the application performance. Even when the resource utilization is accessible for evaluation from outside the virtual environment, it could be relevant to measure the system performance inside the virtual resource in order to evaluate the perceived quality of service by the application without trusting the provider. Specific exceptions can be generated when particular events occur. For example a SLA violation, or saturation of resources by user's application can be notified to the user or directly to the *Client Agent* that will be responsible to ask for a re-negotiation of the SLA.

Table 1. Core agents of mOSAIC platform

Agent type	Function
Client agent	Responsible for collecting users' application requirements, for creating and updating the SLAs in order to grant always to best QoS
Negotiator	Manages SLAs and mediates between the user and the meta-broker, selects appropriate protocols for agreements, negotiates SLA creation, handles fulfilment and violation
Mediator	Select a vendor agent that is capable of deploying a service with the specified user requirements
Vendor agent	Interacts with virtual or physical resources at provider side, and in case the required service needs to be deployed it interacts directly with the automatic service deployer
Archiver	Stores historical data about quality of services and resources offered by providers
Automatic service deployer	Install the required service on the selected resource on demand
Benchmarker	Periodically build performance figures of used resources and notify the client agents about values of measured parameters

3.2 MAGDA as Agents Technology

The MAGDA toolset [7] will be the base for developments of the Cloud Agency in mOSAIC. Its architecture is showed in Figure 2. MAGDA provides a set of agents based services for distributed computing. Users can exploit existing

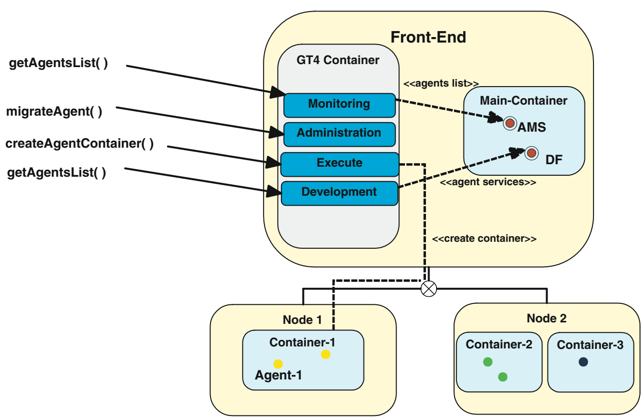

Fig. 2. MAGDA Architecture

services by different protocols, eventually being unaware about the agent technology, accessing a front-end and controlling the agent execution on available resources. On the other hand they can develop new agent based services. Advanced facilities such as strong authentication, agent migration across multiple domains, workload balancing, or dynamic resource allocations are provided [8]. MAGDA provides facilities to design, develop and deploy agents-based services. Services will be stored in the repository and will be able to execute on Cloud resources. Agents will communicate one with th other via standard ACL (Agent Communication Language) messages over http, or over other transport protocols if it will be necessary. Services provided by agents are exposing a Web Services interface that works as a Message Gateway: SOAP to ACL and ACL to SOAP.

3.3 Ontology and Semantic Engine

In order to support interoperability among users and different providers, an uniform *Cloud Ontology* will be defined. It represents a common vocabulary to be used for different purposes. First of all, requests and responses (SLA's, bid requests, and contracts) should be described according to that specific ontology that governs the Cloud domain. Because each user or proder could natively use a different ontology, agents could also implement a semantic mapping between the native ontology and the uniform one. The ontology will be used to implement a semantic discovery facility to find all relevant services and resources published in the Clouds. The ontology will describe Cloud resources and services, Cloud actors, Quality of Service Parameters, the negotiation protocol and the SLA. OWL is prposed as th onntology language. The benefit of using an ontology language is that it acts as a general method for the conceptual description or modeling of information that is implemented by actual resources. This approach will allow to easily take a Cloud resource model and adapt it within other ontology languages making it both platform and vendor agnostic. In this respect, mOSAIC aims to develop ontologies that would offers the main building block to describe the services at the three delivery models. In order to enable algorithms for matching different types of resources, functionalities and capabilities must be captured in these ontologies: a clear description and categorization of existing functionalities, capabilities and specificities of different resources (possibly existing in the Cloud) will ease rapid development of successful applications of the Cloud/over the Cloud. Matching algorithms will equally apply at the different layers of the Cloud architecture selecting the best available Cloud application from a large base of available applications, based on matching customers needs with application specificities. One of the most challenging goals of the semantic engine is to design and develop semantic-based Cloud services discovery. A prototypical tool will be built based on syntactic and structural schema matching. The input will be an ontology describing a service request and services descriptions. This can be achieved on the syntactic level through a service description language (like WSDL), or on the semantic level, through service ontologies (like in OWL-S and WSDL-S). Semantic matching is possible since service request and services descriptions are semantically annotated based on concepts from ontologies adopted for modelling the specific domain of application. The result of a semantic discovery is a new set of exploitable Cloud services and providers.

4 Conclusion

A step forward in the Cloud computing evolution is the development of tools that allows the negotiation and composition of services offered by different Cloud providers. The complexity of the business model related to a multi-Cloud environment imposes the automatization of the offer selections. In this context, mOSAIC proposes to use agent technologies incorporated in a Cloud Agency and a Virtual Cluster to enable the easy development and deployment of multi-Cloud based applications. The on-going development of mOSAIC's proof-of-the-concept prototypes and ready-to-use platform are based on the reasons, concepts, architectures and technologies that were exposed in this paper.

Acknowledgements. This research is partially supported by the grant FP7-ICT-2009-5-256910 (mOSAIC).

References

1. Buyya, R., Yeo, C.S., et al.: Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems 25(6), 599–616 (2009)
2. Sim, K.M.: Towards Complex Negotiation for Cloud Economy. In: Bellavista, P., Chang, R.-S., Chao, H.-C., Lin, S.-F., Sloot, P.M.A. (eds.) GPC 2010. LNCS, vol. 6104, pp. 395–406. Springer, Heidelberg (2010)
3. Cloud Computing Use Case Discussion Group, Cloud Computing Use Cases, White Paper v0.1 (August 2009)
4. Cao, B.-Q., Li, B., Xia, Q.-M.: A Service-Oriented Qos-Assured and Multi-Agent Cloud Computing Architecture. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) CloudCom 2009. LNCS, vol. 5931, pp. 644–649. Springer, Heidelberg (2009)
5. Kaminski, H., Perry, M.: SLA Negotiation System Design Based on Business Rules. In: Procs. 2008 IEEE International Conference on Services Computing, vol. 2, pp. 609–612 (2008)
6. Aversa, R., Di Martino, B., Rak, M., Venticinque, S.: Cloud Agency: A Mobile Agent Based Cloud System. In: Procs. 2010 International Conference on Complex, Intelligent and Software Intensive Systems, pp. 132–137 (2010)
7. Aversa, R., Di Martino, B., Mazzocca, N., Venticinque, S.: MAGDA: A Mobile Agent based Grid Architecture. Journal of Grid Computing. Journal of Grid Computing 4, 395–412 (2006)
8. Aversa, R., Di Martino, B., Venticinque, S.: Integration of Mobile Agents Technology and Globus for Assisted Design and Automated Development of Grid Services. In: Procs. 12th IEEE International Conference on Computational Science and Engineering, vol. 1, pp. 118–125. IEEE Computer Society, Los Alamitos (2009)