Moving Integrated Product Development to Service Clouds in the Global Economy
J. Cha et al. (Eds.)

doi:10.3233/978-1-61499-440-4-271

A Computing Resource Selection Approach Based on Genetic Algorithm for Inter-Cloud Workload Migration

Tahereh NODEHI[a], Sudeep GHIMIRE[b] and Ricardo JARDIM-GONCALVES[c]
[a] *Departamento de Engenharia Electrotecnica, Faculdade de Ciências e Tecnologia da, Universidade Nova de Lisboa (UNL), Campus de Caparica, Portugal*
[b] *CTS,UNINOVA, Departamento de Engenharia Electrotecnica, FCT,UNL, Portugal*
[c] *CTS,UNINOVA, Departamento de Engenharia Electrotecnica, FCT,UNL, Portugal*

Abstract. Cloud computing has been one of the most important topics in IT which aims to assure scalable and reliable on-demand services over the Internet. The expansion of the application scope of cloud services would require cooperation between clouds from different providers that have heterogeneous functionalities. However, current cloud systems do not fully support inter-cloud interoperability and require more research work to provide sufficient functions to enable that seamless collaboration between cloud services. This paper proposes an efficient model for selecting appropriate computing resource from multi-cloud providers that is required to achieve inter-cloud interoperability in a heterogeneous Infrastructure as a Service (IaaS) cloud environment. The goal of the model is dispatching the workload on the most effective clouds available at runtime offering the best performance at the least cost. We consider that each job can have six requirements: CPU, memory, network bandwidth, serving time, maximum possible waiting time, and the priority based on the agreed Service Level Agreement (SLA) contract and service price. Additionally, we assume the SLA contract with suitable criteria between cloud-subscriber and multiple IaaS cloud-providers is signed beforehand. This computing resource selection model is based on Genetic Algorithm (GA). The resource selection model is evaluated using agent based model simulation.

Keywords. Cloud Computing, Inter-Cloud Interoperability, Workload Migration, Infrastructure as a Service (IaaS), Model Driven Architecture (MDA) and Service Oriented Architecture (SOA)

Introduction

Today, a global cloud system includes heterogeneous clouds with finite physical resources. The expansion of the application scope of cloud services would require cooperation between clouds from different providers that have heterogeneous functionalities [1][2]. Cooperation between the heterogeneous cloud service vendors can provide better QoS (eg. scalability and reliability, service availability and performance), avoidance of vendor lock-in, and reduced service production costs. It also can support inter-cloud resource sharing and can provide cloud users the ability of using combined services from different service providers. The required seamless interworking mechanism between clouds is called “Inter-cloud Interoperability”. Most

of the current cloud environments do not support inter-cloud interoperability and more research work is required to provide sufficient functions to enable global seamless collaboration between cloud services [3].

Considering our research work on an Inter-cloud Interoperability Framework (IIF), this paper presents job scheduling model for dispatching the workload from IaaS Cloud Subscriber on other available IaaS Cloud Providers. The purpose of the job scheduling model is to reduce the queuing time and improving the QoS at the lowest cost. The model uses Genetic Algorithm (GA) [4] for job scheduling and considers some suitable criteria for Quality of Service (QoS) and Service Level Agreement (SLA) for cloud systems. Model Driven Architecture (MDA) and Service Oriented Architecture (SOA) are identified as two appropriate approaches for implementing the model in the framework.

This paper includes four sections. The first section discusses on the state of the art in inter-cloud interoperability, section two presents Inter-cloud Computing Resource Selection Approach, section three presents a short introduction on evaluation method, and final section concludes the paper.

1. Inter-cloud Interoperability

The inter-cloud concept is based on the fact that each single cloud has limited computing resources in a restricted geographic area [5][6][7]. Inter-cloud requires interoperability between various cloud computing instantiations allowing cloud costumers to migrate in and out of the cloud and switch between providers based on their needs, without a lock-in which restricts customers from selecting an alternative provider. The inter-cloud network scenario is still in an early stage. Celesti in 2010 [8] proposed a three-phase (discovery, match-making, and authentication) cross-cloud federation model. It has been claimed in [9] Point to Point protocols are not appropriate for inter-cloud protocols and accordingly many-to-many mechanisms have been proposed such as Messaging and Presence Protocol (XMPP) for transport, and Semantic Web techniques such as Resource Description Framework (RDF) as a method to specify resources. In order to show the distinctive ways of interaction between cloud users and providers, NIST [10] defined following use cases for Cloud Computing Interoperability:

- Copy Data Objects between Cloud-Providers.
- Dynamic Operation Dispatch to IaaS Clouds.
- Cloud Burst from Data Center to Cloud.
- Migrate a Queuing-Based Application.
- Migrate Virtual Machines (VMs) from One Cloud Provider to Another.

Nagireddi and Mishra [11] proposed an ontology based framework for searching services provided by different Cloud Service Providers. Abouzamazem and Ezhilchelvan [12] studied tolerating outages by inter-cloud replication and proposed an approach to replicate a service on N outage-independent clouds. Pop and colleagues [13] presented a genetic scheduling algorithm for independent tasks in inter-cloud environments where the selection phase is based on reputation evaluation. Finally, Demchenko and colleagues [14] presented their on-going research on developing the IIF to support on-demand provisioning by heterogeneous cloud service providers.

Nevertheless, from the analyzed state of the art, there is not yet a comprehensive proposal that support the inter-cloud interoperability concerns. We are working on an Inter-cloud Interoperability Framework (IIF) that can support inter-cloud interoperability for dynamic operation dispatch to IaaS Cloud Providers (CP). A fundamental module for the IIF framework is the Computing Resources (CR) selection module from available IaaS CPs. This paper discusses on the CR selection module that uses GA to select the most appropriate CRs from multi CPs.

2. Inter-cloud Computing Resource Selection Approach

As discussed in previous section, we are working on an IaaS IIF framework that will support inter-cloud interoperability for IaaS clouds. The IIF framework's focus is invoking the operations dynamically on the most adequate CPs available based on a the application requirements that are evaluated at runtime. The conceptual model for the IIF framework for IaaS CPs [15] is shown in Figure 1.

The IIF framework is for operations that are independent of unique resources of the IaaS Cloud Subscriber (CS). The IaaS CS attempts to run a job on the CP that is able to provide the best performance, at the least cost. The CS opens an account with each discovered IaaS CP based on CP's SLA and CS has the list of charges and QoS promises of each CP. Then the CS considers a test workload, with specified CPU power, and memory or network performance requirements. The CS operates the test workload a few times on each CP, to arrange the CPs by availability, and performance and price aspects. Moreover, the CS evaluates the CPs for the price and QoS metrics such as availability, and forwards the workloads accordingly. All data and model transformation and mapping tasks between CS and CPs are happening through IIF. In addition, the IIF framework requires one module to select the most effective CRs from IaaS CPs. This section discusses on the CR selection approach.

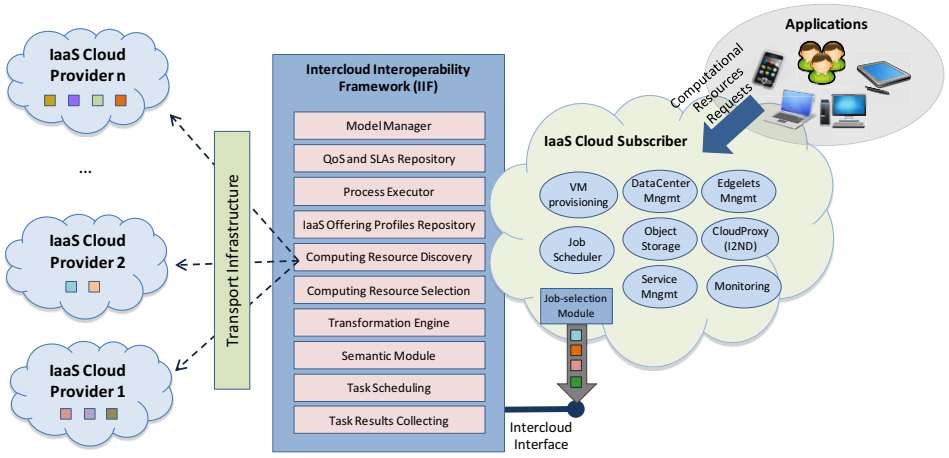

Figure 1. Inter-cloud Interoperability Framework (IIF).

2.1. Job Model

Considering Figure 1, the input of IIF framework from CS is a finite set $J = \{j_i | i = 1, \dots, n\}$ of jobs j_i. The job production is dynamic and each job j_i is based on the specified requirements of applications. Each j_i has a set of requirements $R_i = \{t_i, cp_i, b_i, m_i, d_i, p_i\}$ which t_i is serving time, cp_i is computing power requirement, b_i is bandwidth requirement, m_i is memory requirement, d_i is maximum possible waiting time, and finally p_i is priority based on the agreed SLA contract and service price. In the evaluation section, the possible choice for cp_i, b_i and m_i for the case in this paper are specified. The job arrivals are Markovian (modulated by a Poisson process). The paper considered the serving time t_i has the general distribution, and there are n CPs that may offer the appropriate requirements of the job j_i. *Job-Selection Module* selects the jobs from waiting queue in CS, considering the deadline d_i of job j_i is longer than network delay to get service from other CPs. In short, the paper considers m/g/n queue for the modeling the inter-cloud environment. Additionally, the paper considers the IIF framework supports appropriate functions for IaaS inter-cloud interoperability.

2.2. Genetic Algorithm based Resource Selection (GARS)

In this paper, it is assumed IIF framework receives the workload from CS, and provides the required object model, operation model, and data model of each job j_i. Additionally, it is considered the IIF identifies the QoS parameters $(t_i, cp_i, b_i, m_i, d, p_i)$ for the requirements of each job j_i. Moreover, the SLA criteria between the CS and other IaaS CPs and user profiles are identified in IIF. The Genetic Algorithm based Resource Selection (GARS) approach has following steps:

- The first step in GARS approach is *identifying the available CPs which meet the current work-flow requirements*. To provide this functionality, GARS exploits the information offered by IIF framework.
- The second step, the main focus of this paper, is dispatching the workload on the available CPs effectively. The job allocation method is based on iterative Genetic Algorithm [4]. **Figure 2** shows the GA based model for distributing the jobs receiving from IIF framework on Cloud Providers.

Defining an applicable *fitness function* is essential and having strong effect on the convergence rate of GA and achieving the optimal solution. This paper considered two main factors to define the fitness function:

1. Each IaaS CP's performance: The framework allocates a performance history variable ph_k to each IaaS Cloud Provider CP_k. IIS framework sends a test workload to each Cloud Provider CP_k periodically and updates the performance variable ph_k according to the CP_k's resource availability, CP_k's response time, and CPU throughput.
2. The cost: The IIF framework has the SLA repository based on the agreement between CS and CPs that includes the price lists for different computing resource offering. The $cost_{ik}$ is the cost of computing resource offering from Cloud Provider CP_k for the requirement of job j_i.

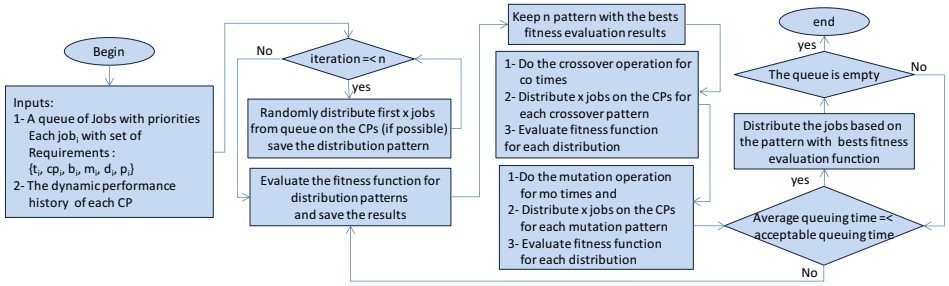

Figure 2. The GA based model for distributing jobs on Cloud Providers.

3. Evaluation

The evaluation of proposed Computing Resource Selection Approach is through agent based simulation. In the simulation process, there are three types of agents:

- *An agent for Cloud Subscriber* : CS agent has number of computing resources including SingleCore, DualCore, QuadCore, and OctoCore processors with variety of attached RAM and different network bandwidth speed. It is possible to provide different combination of available resources that are specified in the SLA of Cloud Subscriber.
- *Predefined number of agents for Cloud Providers*: Each CP agent is specified by different service combinations and their prices. In addition there is a performance history variable ph_k for each IaaS Cloud Provider CP_k agent.

Agents for Jobs produced with the rate of Poisson distribution. The workload characterization is based on selected computational tasks of construction industry. Specification for Cloud hosting is based on the infrastructure for FITMAN[1] project. The overall simulation is modeled within the scope of the scenarios being implemented by UNINOVA and CONSULGAL[2] for FITMAN project. According to the simulation results, following achievement are possible using IIF framework with GARS approach:

- Reduction in waiting queuing time
- Better Quality of Service
- Cost Reduction
- Resource Sharing

In this paper we just show the simulation results for reduction in waiting time. The job-selection module, selects the jobs from waiting queue in CS that are not depend on a specific resource of CS. Additionally the deadline d_i of each selected job j_i is longer than network delay to get service from other CPs. Figure 3 (a) shows the number of jobs, blue line, waiting to get CRs in a single cloud environment are increasing during the time. With similar configuration, Figure 3 (b) shows the number of jobs waiting to get CRs in a multiple cloud providers environment (one CS and four CPs) are close to zero.

[1] Future Internet Technologies for MANufacturing industries : http://www.fitman-fi.eu/
[2] http://www.consulgal.pt/en/

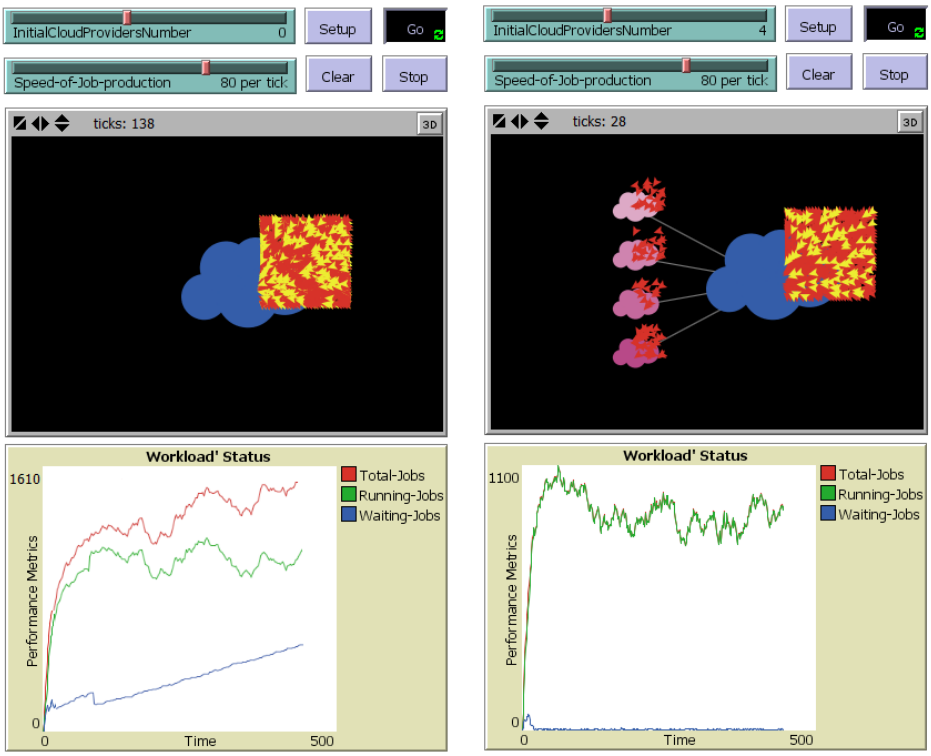

Figure 3. (a) The simulation results for single cloud provider environment that shows the number of jobs waiting to get resources is increasing. (b) The simulation results for multi-cloud provider environment.

4. Conclusion

Current cloud systems do not fully support inter-cloud interoperability. An effective computing resource selection method is required to achieve inter-cloud interoperability for IaaS service cloud providers. Based on our conceptual model for Inter-Cloud Interoperability (IIF), this paper discuss on a genetic algorithm based computing resource selection approach that can be exploit for interoperability in dynamic operation dispatching for IaaS clouds. The resource selection approach has two steps: first, identifying the available CPs which meet the current work-flow requirements, and second, dispatching the workload on the available CPs effectively. The job allocation method is based on GA and considering two main factors to define the fitness function: The performance of each IaaS CP that is measured according to the CP's resource availability, response time, and CPU throughput, and the minimum cost. The evaluation process used the agent based mode which adds dynamic workload to the multi cloud providers' environment and dispatches the waiting workload from CS to CPs. The simulation results show the waiting time to get required resources reduces to very small number when there are 4 CPs compare to a single cloud environment.

MDA and SOA approaches are identified as appropriate approaches to develop the IIF framework.

References

[1] R. Jardim-Goncalves, K. Popplewell, and A. Grilo, "Sustainable interoperability: The future of Internet based industrial enterprises," *Comput. Ind.*, vol. 63, no. 8, pp. 731–738, 2012.

[2] C. Coutinho, A. Cretan, and R. Jardim-Goncalves, "Sustainable interoperability on space mission feasibility studies," *Comput. Ind.*, vol. 64, no. 8, pp. 925–937, Oct. 2013.

[3] R. Jardim-Goncalves, C. Agostinho, F. Lamphataki, Y. Chalarabidis, and A. Grilo, "Systematisation of Interoperability Body of Knowledge: The foundation for EI as a science," *Enterp. Inf. Syst. J.*, 2012.

[4] D. Goldberg, *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley Professional, 1989, p. 432.

[5] D. Bernstein, "The Intercloud : Cloud Interoperability at Internet Scale," *Sixth IFIP Int. Conf. Netw. Parallel Comput.*, 2009.

[6] D. Bernstein, E. Ludvigson, K. Sankar, S. Diamond, and M. Morrow, "Blueprint for the Intercloud - Protocols and Formats for Cloud Computing Interoperability," *2009 Fourth Int. Conf. Internet Web Appl. Serv.*, no. Mdi, pp. 328–336, 2009.

[7] A. Parameswaran and A. Chaddha, "Cloud interoperability and standardization," *SETlabs briefings*, 2009.

[8] A. Celesti, F. Tusa, M. Villari, and A. Puliafito, "How to Enhance Cloud Architectures to Enable Cross-Federation," *IEEE 3rd Int. Conf. Cloud Comput.*, Jul. 2010.

[9] D. Bernstein and D. Vij, "Using XMPP as a transport in Intercloud Protocols," *2nd USENIX Work. Hot Top. Cloud Comput.*, 2010.

[10] L. Badger, R. Bohn, R. Chandramouli, T. Grance, T. Karygiannis, R. Patt-Corner, and J. Voas, "Cloud Computing Use Cases," *National Institute of Standards and Technology,*, 2010.

[11] V. S. K. Nagireddi and S. Mishra, "An ontology based cloud service generic search engine," in *2013 8th International Conference on Computer Science & Education (ICCSE)*, 2013, pp. 335–340.

[12] A. Abouzamazem and P. Ezhilchelvan, "Efficient Inter-cloud Replication for High-Availability Services," in *IEEE International Conference on Cloud Engineering (IC2E)*, 2013, pp. 132–139.

[13] F. Pop, V. Cristea, N. Bessis, and S. Sotiriadis, "Reputation Guided Genetic Scheduling Algorithm for Independent Tasks in Inter-clouds Environments," in *27th International Conference on Advanced Information Networking and Applications Workshops*, 2013, pp. 772–776.

[14] Y. Demchenko, C. Ngo, C. de Laat, J. AntoniGarcia-Espin, S. Figuerola, J. Rodriguez, L. M. Contreras, G. Landi, and N. Ciulli, "Intercloud Architecture Framework for Heterogeneous Cloud Based Infrastructure Services Provisioning On-Demand," *27th Int. Conf. Adv. Inf. Netw. Appl. Work.*, 2013.

[15] T. Nodehi, S. Ghimire, and R. Jardim-goncalves, "Toward a Unified Intercloud Interoperability Conceptual Model for IaaS Cloud Service," in *International Conference on Model-Driven Engineering and Software Development (MODELSWARD 14)*, 2014.

[16] "FI-WARE : FUTURE INTERNET Core Platform," *Funded by: Seventh Framework Programme (FP7) and European Commission*. [Online]. Available: http://www.fi-ware.eu/.

[17] A. Cretan, C. Coutinho, B. Bratu, and R. Jardim-Goncalves, "NEGOSEIO: A framework for negotiations toward Sustainable Enterprise Interoperability," *Annu. Rev. Control.*, 2012.

[18] OMG, "MDA Guide Version 1.0.1," no. June, 2003.

[19] A. M. Jimenez, "Change propagation in the MDA: A model merging approach," The University of Queensland, 2005.

[20] OMG, "Meta Object Facility (MOF) 2 . 0 Query / View / Transformation Specification," 2011.

[21] A. Agrawal, A. Vizhanyo, Z. Kalmar, F. Shi, A. Narayanan, and G. Karsai, "Reusable Idioms and Patterns in Graph Transformation Languages," *Electron. Notes Theor. Comput. Sci.*, Mar. 2005.

[22] OMG, "OMG Meta Object Facility (MOF) Core Specification (Version 2.4.1)," 2011.

[23] IBM, "SOA," 2008. [Online]. Available: http://www-01.ibm.com/software/solutions/soa/.

[24] S. Güner, "Architectural Approaches, Concepts and Methodologies of Service Oriented Architecture," Technical University Hamburg Harburg, 2005.