International Journal of Current Research and Modern Education (IJCRME)
ISSN (Online): 2455 - 5428
(www.rdmodernresearch.com) Volume I, Issue I, 2016

HYBRID CLOUD SELECTION APPROACH TO AUTOMATE THE CLOUD SERVICE SELECTION BASED ON MULTI - CRITERIAN DECISION SUPPORT SYSTEM

K. Kayalvizhi* & A. Anitha**

* PG Scholar, Department of Master of Computer Applications, Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamilnadu

** Assistant Professor, Department of Master of Computer Applications, Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamilnadu

Abstract:

Cloud computing is the delivery of computing resources on demand with reduced management effort. One of the key problems in migrating multi-component enterprise applications to Clouds is selecting the best mix of VM images and Cloud infrastructure services. A migration process has to ensure that Quality of Service (QoS) requirements is met, while satisfying conflicting selection criteria, e.g. throughput and cost. The main problem in mapping software applications to cloud services is selecting the best and most compatible software components to ensure a cost- effective model. When selecting components to migrate the cloud, software engineers must consider many criteria and complex dependencies among other systems' components. Thus, a technique for locating components to be migrated without actually moving them is needed. To overcome these challenges, we propose an approach which can be used in the hybrid decision-making process based on a set of measurable factors in the pricing models of cloud providers. In the presented approach, coupling among different components of the system is measured. Then, a proposed cost measuring function is used to choose the optimal migration scenarios. And implement PSO algorithm to find the fitness values to select optimal cloud service system to overcome complexities in web application systems. VM images, Cloud infrastructure, Qos, Cost-effective model, Optimal migration scenario, PSO algorithm.

1. Introduction:

Cloud computing is a computing paradigm, where a large pool of systems are connected in private or public networks, to provide dynamically scalable infrastructure for application, data and file storage. With the advent of this technology, the cost of computation, application hosting, content storage and delivery is reduced significantly. Cloud computing is a practical approach to experience direct cost benefits and it has the potential to transform a data center from a capital-intensive set up to a variable priced environment. The idea of cloud computing is based on a very fundamental principal of „reusability of IT capabilities'. The difference that cloud computing brings compared to traditional concepts of “grid computing”, “distributed computing”, “utility computing”, or “autonomic computing” is to broaden horizons across organizational boundaries. Forrester defines cloud computing as: “A pool of abstracted, highly scalable, and managed compute infrastructure capable of hosting end customer applications and billed by consumption.”

Cloud services are popular because they can reduce the cost and complexity of owning and operating computers and networks. Since cloud users do not have to invest in information technology infrastructure, purchase hardware, or buy software licenses, the benefits are low up-front costs, rapid return on investment, rapid deployment, customization, flexible use, and solutions that can make use of new innovations. In addition, cloud providers that have specialized in a particular area (such as e-mail) can bring advanced services that a single company might not be able to afford or develop. Some other benefits to users include scalability, reliability, and efficiency.

Cloud computing QoS values of Web services can change dynamically due to the update of server hardware/software or workload change of servers. The Internet environment is highly dynamic. QoS values of web services can change dynamically due to the update of server hardware/software or workload change of servers. Moreover, some of the selected services may become unavailable suddenly at run-time while new service candidates may be launched. The pay-per-use business model promoted by cloud computing paradigm will enable service providers to offer their services to their users in different service levels [16]. Thus, service users will be soon faced with a huge number of variations of the same services offered at different QoS levels. The need for efficient Web service selection approaches is becoming more and more urgent. To address these challenges, we present a cloud model based web service selection approach. The main contributions of this paper can be summarized as follows.

- ✓ To address the problem of web service selection and demonstrate the influence of uncertainty of QoS on the service selection process.
- ✓ To propose a novel concept, called QoS uncertainty computing, to model the inherently uncertain of Web service QoS.

Next, a migration strategy needs to be defined and applied to make the transition from the local data center to the Cloud infrastructure service. A migration strategy defines the migration procedure in means of order and data transfer. The process of migrating an IT system to a Cloud infrastructure service comprises of five steps listed in presorted, modifiable order as following:

- ✓ Cloud infrastructure service selection
- ✓ Cloud VM image selection
- ✓ Cloud VM image customization
- ✓ Migration strategy definition
- ✓ Migration strategy application

2. Multi Criteria Decision Analysis:

One of the major challenges facing an entrepreneur in business prioritization entails coming-up with a reliable model that will rank the available business opportunities (where should we invest). A useful class of models that rank the opportunities is called multi-criteria decision analysis (MCDA) that deals with decisions involving the choice of a best alternative from several potential candidates, subject to several criteria such as those faced by a potential investor. Although MCDA technique seems to offer a natural mechanism to tackle problems of this nature, there is no evidence of their use. Partly this is because the problem is an inter-disciplinary in nature (i.e. Entrepreneurship and Management Science) but mainly because MCDA requires intervention of Management Science that lacks in an entrepreneur. So therefore propose to build the first MCDA business selection tool with inherent family of models to solve the problem.

In essence, one way of establishing this relevance is through a study involving statistical correlation analysis. Second, as individual investors have boundaries on the investment capability and preferences; wish to develop a framework that would account these limitations. This will only be possible if, for each question asked, we enumerate the possible responses from the investor. As the responses are directly linked to the model, it is essential that the validity of such responses is supported via a triangulation research technique. The third challenge entails weighting the responses from the potential investors and linking them to the MCDA model. In this case, weight normalization heuristic needs to be developed and embedded in the MCDA model. While numerous methods exist for weights normalization, we propose a popular rating method that requires the responses to be expressed on a numerical scale.

Finally, for the model to work we requires a database hooked into the MCDA model that enumerates possible business opportunities available within a region. Initially, a sample size based on stratified random sampling technique will be selected. The sample will then be analyzed and generalized for the remaining regions (population) through a fall-back principle that uses statistical induction. MCDA is a structured approach to decision-making that quantitatively evaluates alternatives, in this case, metrics, based on defined project criteria, expert opinions, and stakeholder preferences. It integrates a wide variety of information to evaluate project alternatives and rank them based on their aggregated value with respect to a set of criteria. It usually consists of four stages. The project team, incorporating expert and stakeholder opinions, must define: (1) the set of possible decision alternatives (in this case, metric alternatives) to be evaluated and ranked; (2) the criteria of the value tree that will influence the decision that these alternatives will be evaluated against; (3) the importance of each criterion relative to the others or their "weight" followed by a normalization of weights performed separately for each order of criteria (criteria of order one, criteria of order two (or sub-criteria), etc.); and (4) the value of each alternative with respect to each criterion. Depending on the specific MCDA method, (3) and (4) may also include uncertainty estimates. The hypothetical monitoring goal of the optimal set of metrics is twofold: (i) to select the best restoration alternative; and, (ii) to evaluate restoration project success by measuring the degree to which the intended objectives have been achieved following the project implementation period. The research has the following specific objectives:

Present a new model using MCDA techniques that will prioritize business opportunities.

- ✓ Analyze the criteria used by an entrepreneur for aiding investment decisions.
- ✓ Establishing a process of assigning weights to the different criteria identified.
- ✓ Weighting the preferences (answers) selected by the investors based on the criteria identified.
- ✓ Establishing a process of identifying the business types in a region (i.e. the investment opportunities available in the region).

3. Literature Survey:

3.1 Cloud Genius: Decision Support for Web Server Cloud Migration:

In this paper, introduce the Cloud Genius framework that lowers hurdles introduced by the complexity of the Cloud migration process. Cloud Genius offers a detailed process and comprehensive decision support that reduces a Web engineer's effort of finding a proper infrastructure service and VM image when migrating a Web application to the Cloud. The order reflects the fact that an image can be chosen for a certain Cloud infrastructure service only. Alternatively, selecting a Cloud VM image first restrains the number of eligible Cloud infrastructure services, typically to one. In more complex settings multiple components and databases must be migrated in parallel, what requires to apply the steps described above component wise. Additionally, interconnections and relations between the components must be considered. With Cloud Genius it propose an approach that translates both selection steps into multi-criteria decision-making problems to determine the most valuable combination of a Cloud VM image and a Cloud infrastructure service. The Cloud Genius framework defines a Cloud migration process. Within the process Cloud Genius offers a model and methods to determine the best combined choice of a Cloud VM image and a Cloud infrastructure service.

3.2 A Petri Net-Based Model for Web Service Composition:

Web Services as Petri Nets:

Petri nets (Petri 1962, Peterson 1981) are a well founded process modeling technique that have formal semantics. They have been used to model and analyze several types of processes including protocols, manufacturing systems, and business processes (Aalst 1999). A Petri net is a directed, connected, and bipartite graph in which each node is either a place or a transition. A Web service behavior is basically a partially ordered set of operations. Therefore, it is straight-forward to map it into a Petri net. Operations are modeled by transitions and the state of the service is modeled by places.

In this paper, propose a Petri net-based algebra for modeling Web services control flows. The model is expressive enough to capture the semantics of complex service combinations and their respective specificities. The obtained framework enables declarative composition of Web services. It shows that the defined algebra caters for the creation of dynamic and transient relationships among services. A Web service behavior is basically a partially ordered set of operations. Therefore, it is straight-forward to map it into a Petri net. Operations are modeled by transitions and the state of the service is modeled by places. The arrows between places and transitions are used to specify causal relations. It assumes that a Petri net, which represents the behavior of a service, contains one input place (i.e., a place with no incoming arcs) and one output place (i.e., a place with no outgoing arcs). A Petri net with one input place, for absorbing information, and one output place, for emitting information, will facilitate the definition of the composition operators and the analysis as well as the verification of certain properties (e.g, reach ability, deadlock, and liveness).

3.3 Portable Cloud Services Using TOSCA:

In the life cycle's production phase, the cloud management platform uses management plans to manage the service instance for compliance with the service-level agreements (SLAs) negotiated at subscription time. For example, the management platform assigns additional resources to the instance when the number of users increases, and removes them when users are no longer using the service. The cloud service provider or consumer can also trigger management plans manually for example, to back up or upgrade the service. Finally, when the cloud service consumer decides to get rid of the service or the subscription expires, the service instance terminates and all the resources go back into the resource pool. TOSCA describes composite applications and their management in a modular and portable fashion. It thus defines service templates that contain a cloud service's topology (for instance, an application is hosted on an application server, which is in turn hosted on an operating system) and its operational aspects (such as how to deploy, terminate, and manage this service). Service templates are interpreted by a TOSCA-compliant environment, which operates the cloud services and manages their instances. The creator of a cloud service captures its structure in a service topology a graph with nodes and relationships. Nodes represent the service's components, and relationships connect and structure nodes into the topology.

4. QoS Attributes:

A wide spectrum of metrics which attribute to quality of service has been put forth by the research community with often varying interpretations. Presented here is a list of these metrics with multiple definitions where applicable. Clouds aim to power the next generation data centers by architecting them as a network of virtual services (hardware, database, user-interface, application logic) so that users are able to access and deploy applications from anywhere in the world on demand at competitive costs depending on users QoS (Quality of Service) requirements [3]. Developers with innovative ideas for new Internet services no longer require large capital outlays in hardware to deploy their service or human expense to operate it [2]. It offers significant benefit to IT companies by freeing them from the low level task of setting up basic hardware (servers) and soft-ware infrastructures and thus enabling more focus on innovation and creating business value for their services.

Availability: Availability is the quality aspect of whether the Web service is present or ready for immediate use [4] represented as the percentage of uptime of a service in an observation period[9, 2, 1, 8] and related to its reliability [1].

Reliability: It is the probability that a request is correctly responded within a maximum expected time frame or simply the success rate of finish [8, 9].

Price: The monetary value of the service as set by the service provider [8].

Throughput: Throughput is the number of web service requests served in a given period of time minimising the response time [2]. QoS measures can include the maximum throughput or a function that describes how throughput varies with load intensity [10].

Response Time: The amount of time between sending the request and receiving a response [2] or the guaranteed average time required to complete a service request [1, 9]. Also referred to as execution duration, it is computed using the processing time and the transmission time [8]. It is sometimes referred to as latency [4].

Latency: Time taken between the service requests arriving and the request being serviced. The throughput of a system is related to its latency [1].

5. System Architecture:

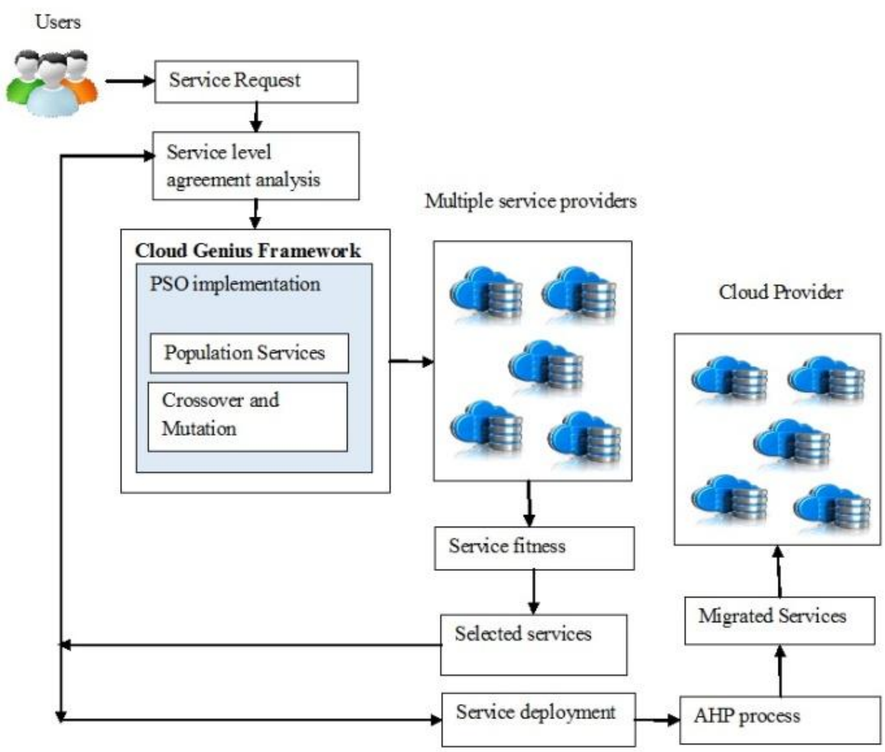

Figure 1: Architecture for cloud genius framework

However, PSO does not have genetic operators like crossover and mutation. Particles update themselves with the internal velocity. They also have memory, which is important to the algorithm. Compared with genetic algorithms (GAs), the information sharing mechanism in PSO is significantly different. In GAs, chromosomes share information with each other. So the whole population moves like a one group towards an optimal area.

6. Risks:

There are many benefits and risks involved in using public IaaS clouds. To get a holistic picture of the benefits and risk from an enterprise perspective, the weighted average of the benefits/risks can be calculated and charted on a radar graph, as shown in Figures 2 and 3. The weighted average can be calculated by multiplying the number of benefits/risks in each

Comparisons Between Genetic Algorithm and PSO:

Most of evolutionary techniques have the following procedure:

- ✓ Random generation of an initial population Reckoning of a fitness value for each subject.
- ✓ It will directly depend on the distance to the optimum.
- ✓ Reproduction of the population based on fitness values. If requirements are met, then stop. Otherwise go back to 2.

From the procedure, we can learn that PSO shares many common points with GA. Both algorithms start with a group of a randomly generated population, both have fitness values to evaluate the population. Both update the population and search for the optimum with random techniques. Both systems do not guarantee success. Category (organizational, legal, security, technical or financial) by the weight of each benefit/risk (unimportant = 1 … very important = 5), and dividing the result by the total number of benefits/risks in that category. Figure 2 shows the weighted average of the benefits for the case studies. It shows that in the case of the digital library, the technical benefits of using public IaaS clouds were more important than the organizational and financial benefits. Hence, the technical ability to deal with volatile demand patterns and cater for a growing number of users would be one of the main motivations for using the cloud. Whereas it is clear that their corporate IT department views financial and organizational benefits as more important than technical ones.

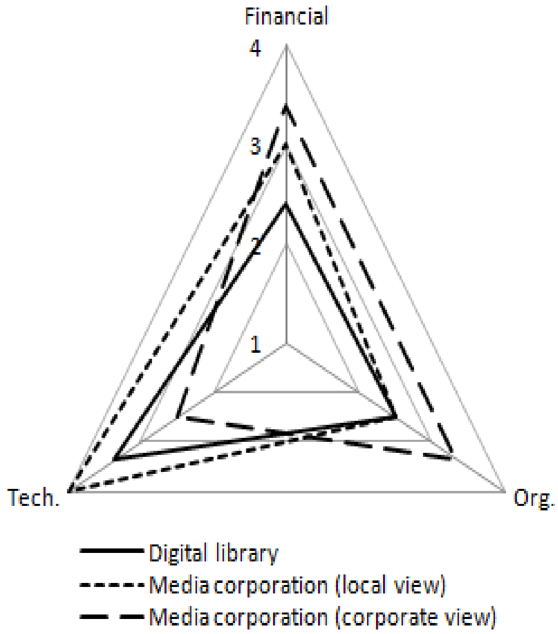

Figure 2: Importance of the different types of types of benifits of cloud migration.

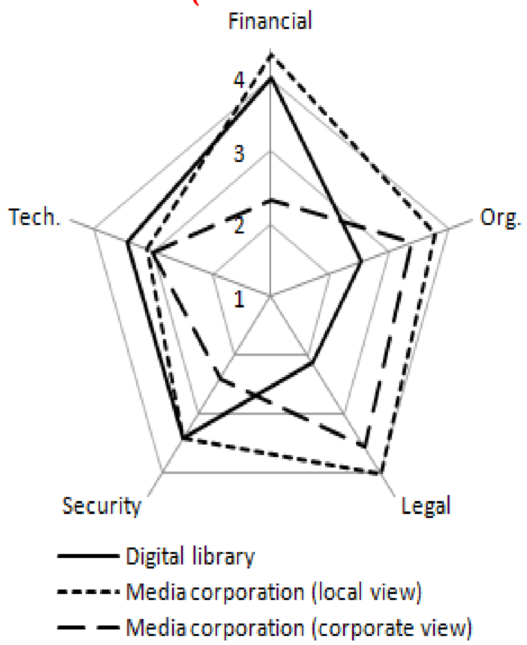

Figure 3: Importance of the different risks of cloud migration

7. Conclusion:

For solving the problem of discovering a user's optimal parameter portfolio for service level and evaluating the properties of any kind of candidate cloud services, we have proposed the cloud service selection model, Cloud Genius to evaluate the properties and select the optimal service which satisfies both user-specified service level and goals most. In hybrid migration, software engineers face the problem of locating the optimal set of components to be migrated statically before actually moving them and calculating the benefit associated with each available migration scenario. This is even more difficult for large systems such as enterprise applications. We conclude that migration scenarios can be guided statically by measuring the degree of coupling among migration candidates. Experimental results emphasize the opinion that less computation and more generic components are more suitable for migration.

8. Future Work:

In PSO, only gBest (or lBest) gives out the information to others. It is a one -way information sharing mechanism. The evolution only looks for the best solution. Compared with GA, all the particles tend to converge to the best solution quickly even in the local version in most cases. PSO is one of the important evolutionary algorithms in service selection. In future work, we can extend our approach to improve service as an android application, in which the service can be selected quickly and accurately.

9. References:

1. M. Menzel and R. Ranjan, "Cloud Genius: Decision Support for Web Server Cloud Migration," in Proceedings of the 21st International Conference on World Wide Web, ser. www '12. New York, NY, USA: ACM, 2012.
2. R. Hamadi and B. Benatallah, "A Petri Net-based Model for Web Service Composition," in Proceedings of the 14th Australasian database conference-Volume 17. Australian Computer Society, Inc., 2003, pp. 191–200.
3. T. Binz, G. Breiter, F. Leyman, and T. Spatzier, "Portable cloud services using tosca," Internet Computing, IEEE, vol. 16, no. 3, pp. 80–85, 2012.
4. S. Wang, Z. Zheng, and Q. Sun, "Cloud model for service selection," WKSHPS), 2011 IEEE . . . , no. 60821001, pp. 666–671, 2011.

5. R. Buyya, R. Ranjan, and R. Calheiros, "Inter cloud: Utility oriented federation of cloud computing environments for scaling of application services," Algorithms and Architectures for Parallel Processing, pp. 13–31, 2010.
6. "Web Application Hosting in the AWS Cloud Best Practices," http://media.amazonwebservices.com/AWS Web Hosting Best Practices.pdf, accessed 2013-07-19., Amazon Web Services.
7. M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica et al., "Above the Clouds: A Berkeley View of Cloud Computing," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS- 2009-28, 2009.
8. P. Mell and T. Grance, "The nist definition of cloud computing, recommendations of the national institute of standards and technology," NIST Special Publication, 2011.
9. "Bit Nami Cloud Images," http://bitnami.org/learn more/ cloud images, accessed 2011-12-02, Bit Nami.
10. "The Cloud Market," http://cloudmarket.com, accessed 2011- 10-19.
11. M. Godse and S. Mulik, "An approach for selecting software as- a-service (saas) product," in Proceedings of the 2009 IEEE International Conference on Cloud Computing, ser. CLOUD '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 155–158. [Online]. Available: http://dx.doi.org/10.1109/CLOUD.2009.74
12. C. Zeng, X. Guo, W. Ou, and D. Han, "Cloud computing service composition and search based on semantic," in Cloud Computing, ser. Lecture Notes in Computer Science, M. Jaatun, G. Zhao, and C. Rong, Eds. Springer Berlin Heidelberg, 2009, vol. 5931, pp. 290–300. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-10665-1 26
13. J. Kang and K.-M. Sim, "Ontology and search engine for cloud computing system," in System Science and Engineering (ICSSE),2011 International Conference on, June 2011, pp. 276–281.
14. R. Buyya, R. Ranjan, and R. Calheiros, "Inter cloud: Utility oriented federation of cloud computing environments for scaling of application services," Algorithms and Architectures for Parallel Processing, pp. 13–31, 2010. [Online]. Available: http://www.springerlink.com/index/B1JP674661364678.pdf
15. A. Ruiz-Alvarez and M. Humphrey, "An automated approach to cloud storage service selection," . . . workshop on Scientific cloud computing, pp. 39–48, 2011. [Online]. Available: http://dl.acm.org/citation.cfm?id=1996117
16. P. Saripalli and G. Pingali, "MADMAC: Multiple Attribute Decision Methodology for Adoption of Clouds," in Cloud Computing (CLOUD), 2011 IEEE International Conference on. IEEE, 2011, pp. 316–323. [Online]. Available: http: //ieeexplore.ieee.org/xpls/absn all.jsp?arnumber=6008725
17. A. Li, X. Yang, S. Kandula, and M. Zhang, "Cloud Cmp: comparing public cloud providers," in Proceedings of the 10th annual conference on Internet measurement, ACM. ACM, 2010,pp. 1–14. [Online]. Available: http://dl.acm.org/citation.cfm?id=1879143