International Journal of Engineering Research and Modern Education (IJERME)
ISSN (Online): 2455 - 4200
(www.rdmodernresearch.com) Volume I, Issue I, 2016

IMPLEMENTATION OF AN ADAPTIVE AND DYNAMIC PORTFOLIO TO ALLOCATE WEB SERVICES IN THE CLOUD

C. Suganya* & K. Adlin Suji**

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

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

Abstract:

View the cloud as a marketplace for trading instances of web services, which can be "leased" by web applications. Argue that the application can "buy" diversity by selecting instances of web services from multiple cloud sellers in this market. The approach portfolio theory to construct a diversified portfolio of web service instances, which are traded from multiple cloud providers. With the growing rate of data used in applying the storage space along with the price for maintaining the data storage increases, thus everyone move towards cloud domain, hence an effective platform is needed for the purchase of the cloud domain and satisfy the following condition, Test the approach effectiveness in minimizing the risk of SLA violation. Simulate the dynamic and adaptive behavior of the approach in responding to changes in the market conditions and riskEvaluate the sensitivity of the allocation decisions to risk and its correlation with other candidates. Evaluate the scalability of the approach and its ramifications on risk reduction under extreme scenarios. Is use a portfolio based algorithm which satisfies the above condition and helps the user to get the required web service with the fluctuating changes in the market without changing the SLA.Selection of cloud-traded services is a relatively new area. However, none of the approaches have considered the problem of diversity in web services, selection through trading and portfolio thinking. The current web services allocation solutions in the cloud market.

Index Terms: Cloud-Based Market, Cloud Computing, Design Diversity, Modern Portfolio Theory, Risk of SLA Violations, Self-Adaptive & Web Service Selection

1. Introduction:

A cloud refers to a distinct IT environment that is designed for the purpose of remotely provisioning scalable and measured IT resources. The term originated as a metaphor for the Internet which is, in essence, a network of networks providing remote access to a set of decentralized IT resources. Prior to cloud computing becoming its own formalized IT industry segment, the symbol of a cloud was commonly used to represent the Internet in a variety of specifications and mainstream documentation of Web-based architectures. This same symbol is now used to specifically represent the boundary of a cloud environment. It is important to distinguish the term "cloud" and the cloud symbol from the Internet. As a specific environment used to remotely provision IT resources, a cloud has a finite boundary. There are many individual clouds that are accessible via the Internet.Whereas the Internet provides open access to many Web-based IT resources, a cloud is typically privately owned and offers access to IT resources that is metered.

Many approach have proposed implementing design diversity techniques to increase the reliability, availability and security of large-scale systems. However, none of them have explicitly linked the distribution of resources to risk and correlation between different candidate providers. The challenge would be to find an efficient and effective solution for inversiting in diversity while considering the risk and correlation between providers. Moreover, the dynamic nature of the cloud mptivates the need for adaptation in that solution.

Much of the Internet is dedicated to the access of content-based IT resources published via the World Wide Web. IT resources provided by cloud environments, on the other hand, are dedicated to supplying back-end processing capabilities and user-based access to these capabilities. Another key distinction is that it is not necessary for clouds to be Web-based even if they are commonly based on Internet protocols and technologies. Protocols refer to standards and methods that allow computers to communicate with each other in a pre-defined and structured manner. A cloud can be based on the use of any protocols that allow for the remote access to its IT resources.

2. Related Work:

Isdemonstrate how a cloud-based application can reduce the risks associated with web service allocation through simulating a hypothetical cloud market. A compared the risk reduction achieved by portfolio-based selection with the following non-diverse auction-based mechanisms of, where auctioning is based on risk or price algorithm and classical design diversity algorithm: Risk-Based auction that allocates all the required services from a single cloud provider that has lowest risk of SLA violations. Price-Based auction that allocates all the required services from a single cloud provider that have the lowest price.

Classical Design Diversity that implements diversity by evenly distributing the services among multiple providers without giving much consideration to the risk or correlation between the providers. The evaluation has been designed to test how our approach can be applied to a typical scenario of use where, for experiments 1-3, we have considered a scenario involving three candidate services for simplicity of exposition; nevertheless, the argument and the technique can be applied to more than three services. Furthermore, the number of considered providers tends to be limited to a few, as dealing with different providers may come with costs and overheads. Experiment 4 is aimed at stress-testing the mechanism for scalability.

Is used 400 candidate services. The choice of 400 is aimed to test/stress for scale and exhibit an extreme scenario. A systematic literature review had shown that the current approaches typically use 20 candidate services in dynamic selection and composition problem; henceforth, 400 goes far beyond the classics. Following is a description of the rationale for the choice of each experiment:

- ✓ The first experiment is designed to evaluate the effectiveness of the portfolio-based allocation in reducing the risk of SLA violation in comparison to price-based auction, risk-based auction and classical design diversity.
- ✓ The second experiment evaluates the self-adaptive behaviour of the portfolio-based allocation by changing the risk of SLA volition in the cloud market.
- ✓ The third experiment is designed to evaluate the sensitivity of the approach to changes in correlation between providers in the optimum portfolio selection. Compare the approach to the classical design diversity approach.
- ✓ The fourth experiment is designed to test the scalability of the approach.

It attempts to determine how the risk of SLA violation and execution time of the portfolio selection tend to be affected by increasing the number candidate services in the portfolio.

The Effectiveness of the Portfolio Based Allocation Recall the case of the budget flight booking website, Flight. Com, which provides the online booking web service Flight- Booking, which is a web service that leases variants of Concrete service through various clouds. As mentioned previously, the variants of the web service instances from various providers tend to provide the same core functionalities, but they differ in price and the way they address non functionalities such as availability, execution time and security. In high seasons, Flight.com has decided to scale up its services to support an anticipated load in the number of users through selecting and subsequently allocating 100 instances.

Let us assume that the minimum accepted value of QoS is as follows: 65 for (A) availability, 50 for (E) execution time and 80 for (F) security and that all of the QoS dimensions have .33 priority weights (see Section 3.3 for details about the pro- posed scale and QoS priority weights). Moreover, let us assume that the maximum price that they are willing to pay is $32 for each instance. , we can see a snapshot of the current Flight Booking web ser- vice offers that are available in the simulated cloud-based market which satisfy the QoS and price constraints as well as the risk of SLA violation associated with each web ser- vice. The correlation of SLA violation between the current Flight Booking web services. For this experiment, the simulation methodology consists of two steps:

- ✓ Performing web services allocation to the simulated market state using: a. Portfolio-based allocation. b. Classical design diversity allocation. c. Price-based allocation (minimum price). d. Risk-based allocation (minimum risk).
- ✓ Allocating the web services in the simulated market and calculating the risk of SLA violation. The results shown are the averaging of 30 simulations. Web service allocation using the portfolio-based method. When we use portfolio-based optimisation to allocate instances of web services, the first step is to find the percentage of instances that we should allocate from each of the candidate services in order to achieve the minimum portfolio risk. We can find the weights by applying portfolio optimisation, where the goal is to minimise the risk in constraints. The optimum weights of the optimisation process are depicted in Fig. 3. These weights imply that we will be able to construct the minimum risk portfolio of a web service by allocating 70 ser- vice instances from Flight Booking 2 at the cost of $32 for each service and 30 services from Flight Booking 3 at the cost of $31 for each service. This makes the total cost of constructing this portfolio to total $3170. After allocating the web service as in the optimum portfolio and running the Effectiveness of the Self- Adaptive Portfolio-Based Allocation In this experiment, we will evaluate the effectiveness of the self-adaptive behavior where the approach should system- apically evaluate the risks of the current portfolio and com- pare it to the optimal traded portfolio at a given time. It should then dynamically decide on a new portfolio and adapt the application accordingly, as detailed in algorithm. Correlation Sensitivity in Portfolio-Based Allocation and Classical Design Diversity Allocation In this experiment, we will evaluate the effect of the change in the correlation between providers on the selection process using portfolio-based allocation and classical design diversity allocation. To do this, we run 90 simulations with the configuration, except that correlation will change between Flight Booking 3 and Flight Booking.

3. Proposed Work:

Propose a novel, dynamic and adaptive approach for implementing design diversity in the cloud market.The approach uses portfolio theory to construct a diversified portfolio of web service instances, which are traded from multiple cloud providers.Illustrate the applicability of the approach. Evaluate the sensitivity of the allocation decisions to risk and its correlation with other candidates and evaluate the

International Journal of Engineering Research and Modern Education (IJERME)
ISSN (Online): 2455 - 4200
(www.rdmodernresearch.com) Volume I, Issue I, 2016

scalability of the approach and its ramifications on risk reduction under extreme scenarios.Controlled experiments are also used to

- ✓ Test the approach effectiveness in minimizing the risk of SLA violation;
- ✓ Simulate the dynamic and adaptive behavior of the approach in responding to changes in the market conditions and risk;
- ✓ Evaluate the sensitivity of the allocation decisions to risk and its correlation with other candidates and
- ✓ Evaluate the scalability of the approach and its ramifications on risk reduction under extreme scenarios.
- ✓ It's used to minimizing the risk of SLA violation.
- ✓ Its dynamic and adaptive behavior of the approach in responding to changes in the market conditions and risk.
- ✓ Its allocation decisions to risk and its correlation with other candidates.
- ✓ In proposed system have add the ratings and feedbacks for the user reference.
- ✓ If SLA, violation occurs on one of the candidate services, there is a great chance that it will not have any effect on the other of resources.

The scope of the project is buyer can buy the cloud in easy way by agent(Market regulator). The market regulator can also monitoring the Service Level Agreement (SLA) voilation complaince and probable risk. The main Scope of my project is to improve the SLA voilation by using portfolio-based optimization.

```
Begin
1: if (first adaptation cycle)
2: then set number of services required
3: set QoS and price constraints
4: set weight of quality attributes PWA, WE and PWF
5: end if
6: for each service Si in Marketnew
7: if Si satisfies all constraints
8: add Si to set of candidate services S
9: end if
10: end for
11: for each candidate services Si. 2 S
12: calculate aggregated QoS qi
13: get correlation pij
14: get risks Ri
15: end for
16: Newportfolio ¼ quadprog(mim(eq. (8)), S.t.(Eqs. (6), (7))
17: Calculate the risk of new portfolio Rpoptimum in Marketnew
18: Calculate the risk of currently allocated portfolio Rpcurrent in Marketnew
19: The potential improvement in risk Ic ¼ Rpcurrent Rpoptimum
20:ifðIcImÞ
21: then submit new optimum portfolio
22: else submit keep currently allocated portfolio
23: end if End
```

The algorithm in the first adaptation cycle asks the buyer to set the minimum accepted QoS, the required number of web services, the maximum price that (s)he is willing to pay Cmax as well as the weight of quality attributes PWA, WE and PWF of the web service in lines (1-5). Then the algorithm identifies the web services Si which offer the functionalities required by the user and satisfy the QoS and cost constraints in lines (6-10). After that, the buyer agent will accesses the Knowledge Base, which is maintained by the market regulator, to retrieve the likely risks Ri of SLA violation, qi aggregated QoS and correlation matrix p associated with each candidate web service in lines (11-15). Then the quadprog function, provided by the MatLab optimization toolbox, will return a vector that contains the weights of each web services in a new optimum portfolio. After that, we calculate the risk of the new optimum portfolio and the currently allocated portfolio in lines (16-18). Finally, we calculate the level of improvements in risk Ic and if it exceeds the minimum accepted improvement level Im, an adaptation will be triggered to allocate new set of web services based on the recommended weights from the new optimum portfolio in lines (19-23).

4. Experimental Analysis and Results:

To the best of our knowledge, neither portfolio-based design diversity nor dynamic adaptive systems has addressed the problem that explore in work. The combination of portfolio thinking with web instance diversification is a promising approach for dynamically and adaptively improving QoS and reducing risks associated with SLA violations.

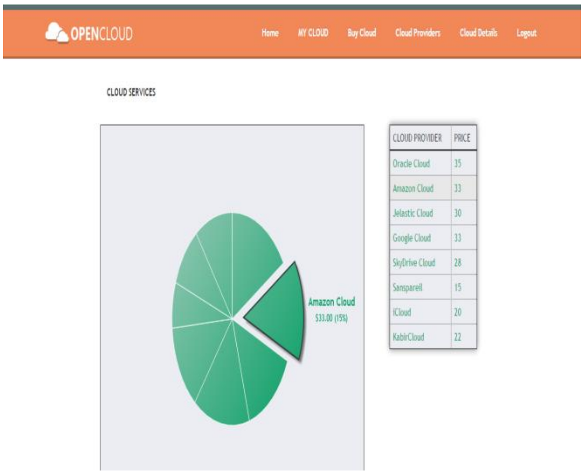
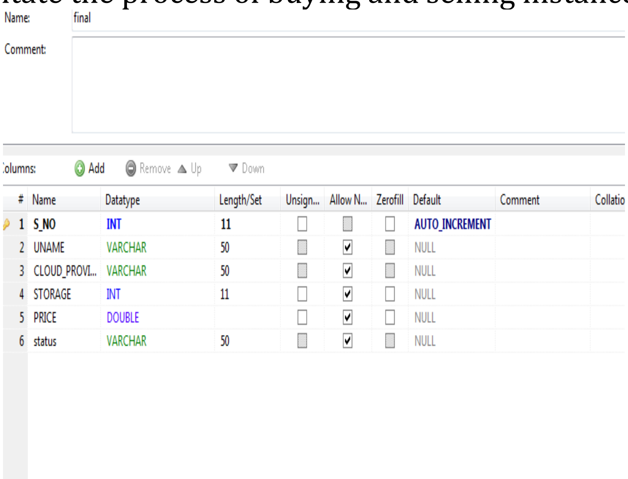


Figure 1: Cloud Ratings

The user must create the account for login to buy the cloud.The user can buy the cloud and it is user friendly. User Details have been stored in the database, with the time of sign in, log in and log out.User can get the cloud based on portfolio algorithm. A cloud marketplace will facilitate the process of buying and selling instances of web services.



The Admin can view the details of the registered users who were all sign up in the website.The Administrator can view the details of the user login and logout server.The Admin can view the Table of portfolio algorithm. The calculated values will be displayed to the particular user.The User details and login details will be stored in database.

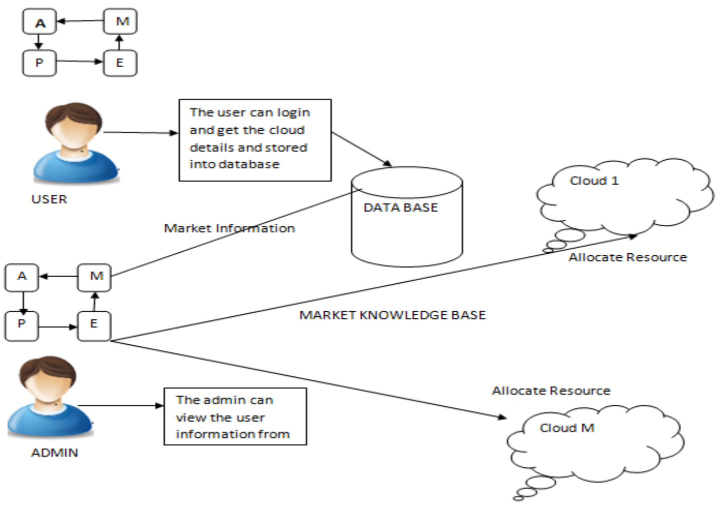


Figure 3: System Architecture

5. Conclusion and Future Enhancement:

Introduced a novel, dynamic and adaptive design diversity approach for web services selection and allocation in the cloud using portfolio thinking. Viewed the cloud as a marketplace for trading instances of web services, which cloud-based applications can explore trade and use as substitutable and compose able entities. Have used a portfolio-based optimization to improve SLA compliance by diversifying the selection and consequently the allocation of traded instances of web services from multiple providers. In this work, investing in diversity on a given type of cloud services, where for every abstract service in the architecture, there exist numbers of concrete candidate services.

6. References:

1. R. Buyya, C. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Gener. Comput. Syst., vol. 25, no. 6, pp. 599–616, Jun. 2009,
2. L. Zeng, B. Benatallah, A. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, "QoS-aware middleware for Web services composition," IEEE Trans. Softw. Eng., vol. 30, no. 5, pp. 311–327, May 2004.
3. L. Qia, W. Doua, X. Zhangc, and J. Chenc, "A QoS-aware composition method supporting cross-platform service invocation in cloud environment," J. Comput. System Sci., vol. 78, no. 5, pp. 1316–1329, Sep. 2012.
4. J.O. Gutierrez-Garcia and S. Kwang-Mong, "Self-organizing agents for service composition in cloud computing" in Proc. Int. Conf. Cloud Comput. Technol. Sci., Indianapolis, IN, USA, 2010, pp. 59–66.
5. N. Bonvins, T. G. Papaioannou, and K. Aberer, "An economic approach for scalable and highly-available distributed applications in cloud computing (CLOUD)," in Proc. IEEE 3rd Int. Conf., 2010, pp. 498–50.