Measuring the Scalability of Cloud-based Software Services

Amro Al-Said Ahmad
School of Computing and Mathematics
Keele University, UK
a.m.k.al-said.ahmad@keele.ac.uk

Peter Andras
School of Computing and Mathematics
Keele University, UK
p.andras@keele.ac.uk

Abstract— **Measuring and testing the performance of cloud-based software services is critically important in the context of rapid growth of cloud computing. Scalability, elasticity and efficiency are interrelated aspects of performance of cloud-based software services. Here we present a work that is focused on measuring the scalability of cloud-based software services in technical terms. We introduce technical scalability metrics inspired by earlier technical metrics of elasticity.**

Keywords-measurement; performance; scalability; Software-as-a-Service (SaaS); metrics;

I. INTRODUCTION

The performance assessment and testing of cloud-based software services is critically important in order to support the Service Level Agreement (SLA) compliant quality of delivery of these services, especially in the context of rapidly expanding the quantity of service delivery [1]. There are three interrelated cloud-specific performance aspects that are key determinants of service quality delivery: scalability, elasticity and efficiency [2, 10].

Following [3] we adopt the following definition of the scalability aspect. Scalability is the ability of the cloud layer to increase the capacity of the software service delivery by expanding the quantity of the software service that is provided. This definition focus on the technical side of cloud-based software services, however we note that alternative, utility oriented (i.e. economic cost/benefit focused), approaches are also used in the literature [4, 5].

Measuring and testing scalability of cloud-based software services from a technical perspective are key for the assessment and testing of performance [6]. This will help in designing appropriate test scenarios and identifying options for changes and upgrades that can improve the scalability performance of the system.

Here we follow ideas proposed in the context of measurements and metrics for cloud elasticity [7-9] to propose technical metrics for the scalability of cloud-based software services. The proposed scalability metrics address both volume and quality scaling of cloud-based software services. The rest of the paper is structured as follows. first we present our approach to measure and quantify scalability of cloud-based applications. Finally, present our conclusions and future work section.

II. SCALABILITY PERFORMANCE MEASUREMENT

In principle the increase of capacity could happen either by increasing the volume of service requests served by a single instance of the service provision software or by deploying multiple instances of the service software, or by a combination of these two approaches. In general, we expect that if a service scales up ideally then the increase in demand for service should be matched by proportional increase in the provision of the service instance(s) such that the quality of the service does not change. Here quality of the service may be seen for example in terms of average response time. This ideal scaling behavior of the system should be valid over a sufficiently long time scale, i.e. short-term mismatches between provision and demand, which are the subject of elasticity, are not relevant from the perspective of scalability. If the system does not scale according to the ideal manner, it recruits insufficient resources to deliver the increased volume of service (instances) without change in the quality of the service. In general, real systems are expected to operate below the level of the ideal scaling behavior and the aim of measuring scalability is to quantify the extent to which the real system behavior differs from the ideal behavior.

To deliver the ideal scaling we expect that the system increases the number of instances of the software proportionally with the increase in demand for the software services and also that the system maintains the quality of service in terms of maintaining the same average response time irrespective of the volume of service requests. Formally, let us assume that D and D' are two service demand volumes, $D' > D$. Let I and I' be the corresponding number of software instances that are deployed to deliver the service, and let t_r and t'_r be the corresponding average response times. If the system scales ideally we expect that for any levels of service demand D and D'.

$$D' / D = I' / I \qquad (1)$$

$$t_r = t'_r \qquad (2)$$

Equation (1) expresses that the volume of software instances providing the service scales up with the demand for the service. Equation (2) expresses that the quality of service, in terms of average response time, remains unchanged for any level of service demand.

To measure the values of I and t_r the system must perform the delivery of the service over some sustained time, such that short-term variations, due to elastic response of the system, do not influence the system measurements. In practice this means that the number of software instances and the average response time should be calculated by averaging over a number of measurements during the execution of a demand scenario (e.g. every second), and a number of repeated applications of the same demand scenario, i.e. a pattern of demand presentation, which may include variation in the demand.

The difference between the ideal and the actual scaling behavior of the system offers the possibility of defining technical scalability metrics for cloud-based software services.

In terms of provision of software instances for the delivery of the services the scaling is deficient if the number of instances is lower than the ideally expected number of software instances. To quantify the level of deficiency we pick a demand scenario and start with a low level of demand D_0 and measure the corresponding volume of software instances I_0. Then measuring the number of software instances I_k corresponding to a number (n) of demand levels D_k following the same demand scenario, we can calculate how close are the I_k values to the ideal I^*_k values ($I_k < I^*_k$). Following the ideal scalability assumption of equation (1) we get for the ideal I^*_k values:

$$I^*_k = (D_k / D_0) \cdot I_0 \qquad (3)$$

Considering the ratio between the area defined by the (D_k, I_k) values, $k = 0,\ldots,n$, and the area defined by the (D_k, I^*_k) values we get a metric of service volume scalability of the system:

$$A^* = \sum_{k=1,\ldots,n} (D_k - D_{k-1}) \cdot (I^*_k + I^*_{k-1}) / 2 \qquad (4)$$

$$A = \sum_{k=1,\ldots,n} (D_k - D_{k-1}) \cdot (I_k + I_{k-1}) / 2 \qquad (5)$$

$$\eta_I = A / A^* \qquad (6)$$

where A and A^* are the areas under the curves calculated for actual and ideal I values and η_I is the volume scalability performance metric of the system. If η_I is close to 1 the system is close to ideal volume scalability, if it is close to 0, then the volume scalability of the system is much less than ideal.

Similarly, we can define the quality scalability of the system by measuring the service average response times t_k corresponding to the demand levels D_k. We approximate the ideal average response time as t_0, following the ideal assumption of equation (2). The quality scalability of the system is less than ideal if the average response times for increasing demand levels increase, i.e. $t_k > t_0$. By considering the ratio between the areas defined by the (D_k, t_k) values, $k = 0,\ldots,n$, and the area defined by the (D_k, t_0) values we get a ratio that defines a metric of service quality scalability for the system:

$$B^* = \sum_{k=1,\ldots,n} (D_k - D_{k-1}) \cdot t_0 = (D_n - D_0) \cdot t_0 \qquad (7)$$

$$B = \sum_{k=1,\ldots,n} (D_k - D_{k-1}) \cdot (t_k + t_{k-1}) / 2 \qquad (8)$$

$$\eta_t = B^* / B \qquad (9)$$

where B and B^* are the areas under the curves calculated for actual and ideal t values and η_t is the quality scalability performance metric of the system. If η_t close to 1 the system is close to ideal quality scalability, if it is close to 0 the quality scalability of the system is much less than ideal.

VI. Conclusions And Future work

In this paper we present two scalability metrics for cloud-based software services. One of these addresses the volume scalability of the service, while the other the quality scalability of the service. The metrics are based on simple principles of proportional scaling of the service volume and constant provision of the service quality, and are defined using the differences between the real and ideal scaling curves for both the volume and quality scalability.

We believe that the proposed technical scalability metrics can be used to perform and design scalability testing of cloud-based application services with the aim to identify system components that critically contribute to the technical scalability performance. Furthermore, the proposed metrics can be extended, by considering alternative service quality features, and combined with a range of demand scenarios to support the fine-tuning of the system, the identification of quality-of-service trade-offs, and estimation of realistic scalability performance expectations about the system depending on demand scenarios.

Future work will include the application of the metrics using cloud-based software applications run on public cloud platforms, using multiple demand scenarios to show how these impact on the scalability performance of cloud-based software services.

Acknowledgment

This research is supported by a PhD scholarship from Philadelphia University – Jordan for Amro Al-Said Ahmad. We thank our colleagues, Fiona Polack and Pearl Brereton, for useful comments and suggestions.

References

[1] J T. Atmaca, T. Begin, A. Brandwajn and H. Castel-Taleb, "Performance Evaluation of Cloud Computing Centers with General Arrivals and Service," in IEEE tran. on Parallel and Distributed Systems, vol. 27, no. 8, 2016, pp. 2341-2348.

[2] M. Becker, S. Lehrig, and S. Becker, "Systematically deriving quality metrics for cloud computing systems." In the 6th Int. Conference on Performance Engineering, ACM, 2015, pp. 169-174.

[3] S. Lehrig, H. Eikerling and S. Becker, "Scalability, elasticity, and efficiency in cloud computing: A systematic literature review of definitions and metrics," 2015 11th Int.ACM SIGSOFT Conference on Quality of Software Architectures, Montreal, QC, 2015, pp. 83-92.

[4] R. Buyya, R. Ranjan, R.N. Calheiros, "InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services". in the 10th Int. conference on Algorithms and Architectures for Parallel Processing - Volume Part I (ICA3PP'10). Springer-Verlag, Berlin, 2010, pp 13-31.

[5] K. Hwang, Y. Shi and X. Bai, "Scale-Out vs. Scale-Up Techniques for Cloud Performance and Productivity," 2014 IEEE 6th Int. Conference on Cloud Computing Technology and Science, Singapore, 2014, pp. 763-768.

[6] K. Blokland, J. Mengerink, and M. Pol, Testing Cloud Services: How to Test SaaS, PaaS & IaaS. Rocky Nook Inc., California, USA, 2013.

[7] D. Jayasinghe, et al., "Variations in Performance and Scalability: An Experimental Study in IaaS Clouds Using Multi-Tier Workloads," IEEE Trans. Serv. Comput, vol. 7, no. 2, 2014, pp. 293-306.

[8] D. Jayasinghe, et al., "Variations in Performance and Scalability When Migrating n-Tier Applicat. to Different Clouds," in IEEE Int. Conf. on Cloud Computing, Washington, DC, 2011, pp. 73 - 80.

[9] J. Gao, P. Pattabhiraman, X. Bai and W. Tsai, "SaaS performance and scalability evaluation in clouds,", in th IEEE 6th Int. Symp. on Service Oriented System Engineering, Irvine, CA, 2011, pp. 61-71.

[10] N. Herbst, S. Kounev, and R.H. Reussner. "Elasticity in Cloud Computing: What It Is, and What It Is Not." In ICAC, vol. 13, 2013, pp. 23-27.