Resource Allocation for Maximum Performance and Minimum Cost for 3-tier SaaS Application in Azure

Sasko Ristov[1] and Marjan Gusev[2] Bojana Koteska[3], Goran Velkoski[4],

***Abstract*—Windows Azure is one of the most common commercial clouds, which offers a scalable and elastic platform to host applications. The cloud customers can choose from various number of different type instances according to their needs in order to avoid application bottleneck. Scaling the resources can be done on two ways, i.e., either to increase the instance with additional resources (CPU, RAM, HDD), or to add an additional instances, each with the same resources as the first one. Since the customer should pay different price for the scaling, we are interested which scaling provides better performance, and if we can generalize it. The goal of this paper is to determine which resource organization produces better performance and cost, as well as better price-performance ratio, i.e., if it is better to host the web application in a large number of "smaller" application instances or small number of "bigger" instances. The results show that the latter provides much better performance for less price.**

***Index Terms*—Cloud Computing; Performance; SaaS.**

I. Introduction

Cloud computing is a paradigm that usually reduces the customers' costs if they migrate their services into the cloud. However, the performance of the services hosted in the cloud are usually discrepant due to the additional virtualization layer and more important, the cloud multitenant and shared resources environment. The performance depends on the rented hardware resources, platform environment, the number of active virtual machines (VMs) on the same physical server, the total number of active VMs in the cloud, and so on. Therefore, the existing pay-as-you-go model is a little bit unfair for the cloud customers.

The Cloud has another advantage compared to the traditional IT hosting platforms, that is, it is scalable and elastic. The price scales similar to the scaled resources, but how does the performances scale? In this paper we analyze these issues for one of the most common commercial clouds, that is, the Microsoft Windows Azure. Azure is a realistic platform for parallelization, especially for communicationless applications [13], but it is not appropriate for tightly-coupled applications compared to Amazon EC2 or cluster [14].

[1] S. Ristov is with University Ss Cyril and Methodius, Faculty of Computer Science and Engineering from 2008, Rugjer Boshkovik 16, PO Box 393, 1000 Skopje, Macedonia, Email: sashko.ristov@finki.ukim.mk

[2] M. Gusev is with University Ss Cyril and Methodius, Faculty of Computer Science and Engineering from 1989, Rugjer Boshkovik 16, PO Box 393, 1000 Skopje, Macedonia, Email: marjan.gushev@finki.ukim.mk

[3] B. Koteska is with University Ss Cyril and Methodius, Faculty of Computer Science and Engineering from 2007, Rugjer Boshkovik 16, PO Box 393, 1000 Skopje, Macedonia, Email: bojana.koteska@finki.ukim.mk

[4] G. Velkoski is with Innovation Dooel, Vostanichka 118-18, 1000 Skopje, Macedonia from 2013, Email: goran.velkoski@innovationl.com.mk

We are trying to determine which organization of the resources in the Azure cloud will provide the best performance for the most common three tier application (Web, Application and Database Servers) by varying the application load (the number of requests) and the number of instances and their sizes (the number of CPUs, RAM and HDD within the VMs). We continue with the research to determine the cost performance trade-off as well. Similar behavior have the web services in the cloud, i.e., up to 10 times better performance is achieved when they are hosted in many small VMs, instead of using one huge VM with all resources allocated with in [4].

However, the cloud is multi-tenant virtual environment and there are many challenges about the performance. The same instance of a VM behaves differently for a certain load, among other active VMs [8]. Therefore, the cloud multi-tenant virtual environment must be isolated in terms of performance [16]. If a cloud service provider (CSP) adds more nodes and thus under-utilizes the CSP's resources, then the performance can be improved by implementing more parallelism [6].

The paper is organized as follows. Related work in the area of cloud cost and performance is given in Section II. Section III presents the testing methodology, plan and infrastructure. The results from the experiments are described in Section IV. The conclusion and future work are specified in Section V.

II. Related Work

The performance of various cloud applications and services by using different resources are analyzed by many authors.

Agarwal and Prasad [1] extend the research and present the AzureBench - a benchmark suite for Windows Azure platform's storage services. Windows Azure can decrease the costs and time for deployment, as well as can support the efficient TCP data transfers [15]. Lu et al. [9] examine several pitfalls in the Windows Azure cloud during several days of performing the experiments: Instance physical failure, Storage exception, System update. They also discovered several pitfalls resulting in waste of active VM idling. The performance of VMs, storage and SQL services in Windows Azure are analyzed by Hill et al. [5].

In our recent research, we have analyzed several different server loads, platforms [12], clouds and applications. Scaling the resources in Windows Azure can lead to superlinear speedup, that is, the performance is scaled more than the scaled resources [2]. However, it is better to use many smaller VM instances than to use parallelization in Windows Azure [3].

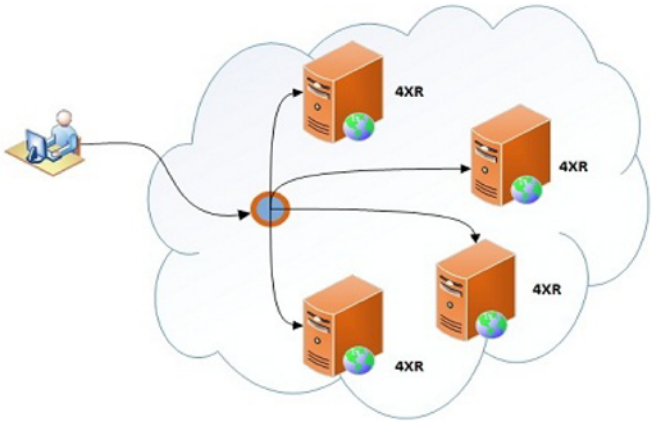

Fig. 1. 4x4 environment

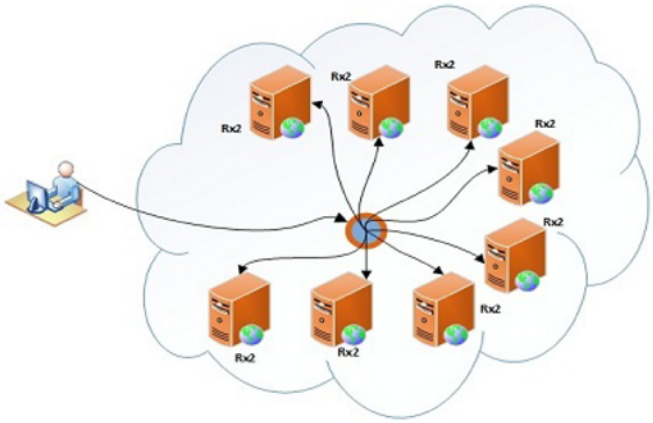

Fig. 2. 8x2 environment

III. Testing Methodology

We use client-server testing environment hosted in Windows Azure cloud. The server side consists of the SaaS application "PhluffyFotos" [10] (Figure 3), which is a sample cloud application developed for public use. This sample uses several technologies: ASP.NET MVC 4, Windows Azure SQL Databases and Windows Azure Storage, including Tables, Blobs, and Queues. The database consists of 100 albums with one picture per album. The client uses Apache JMeter [7] to test the SaaS application performance by varying the load (the number of requests) and by using different number of instances with various number of resources. The client and the cloud are placed in the same data center (North Europe) in order to minimize the network latency.

Our testing is consisted of reading information from the SQL database, i.e., accessing a web page which displays the albums that have been created from all users. We conduct two experiments in different cloud environments using the same total amount of resources, but allocated in different number of application instances.

Figure 1 depicts the first cloud environment where the 4 instances of the SaaS application are hosted on four Large

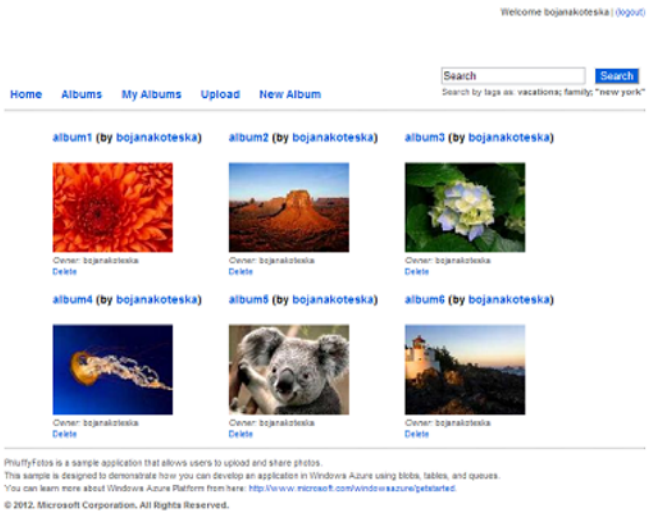

Fig. 3. PhluffyFotos web page showing the first six created albums

VMs, each allocated with four CPU cores. Figure 2 depicts the second cloud environment where eight instances of the SaaS application are hosted on eight Medium VMs, each allocated with two CPU cores.

Each experiment consists of 5 test cases with different throughput (number of http requests per second) $N = 1, 250, 500, 750$ and 1000. The range of parameter N is selected such that web servers in instances work in normal mode without replying error messages and all requests are being completed.

The performance of the SaaS application is calculated by measuring the performance of the throughput, i.e., the average response time T for various throughput (the number of requests). We introduce *Relative Throughput* $R = T_{8x2}/T_{4x4}$ in order to compare which resource organization provides better throughput.

IV. Experimental Results

In this section, we present and analyze the experimental results of testing the SaaS performance represented as average response time for a given throughput. Further on, the analysis continues by analyzing which is the cheaper environment and to determine the cost-performance ratio.

A. Throughput Performance

Figure 4 depicts the measured response time as a function of a given throughput of the SaaS application. The results show that increasing the throughput also increases the average response time, as expected. But more important conclusion is that better performance is achieved when the number of instances of the application is smaller, and the number of cores is larger, i.e., when the SaaS application is hosted in cloud environment with 4 application instances hosted on 4 VMs, each allocated with 4 CPU cores. This is emphasized when the throughput is $N = 500$.

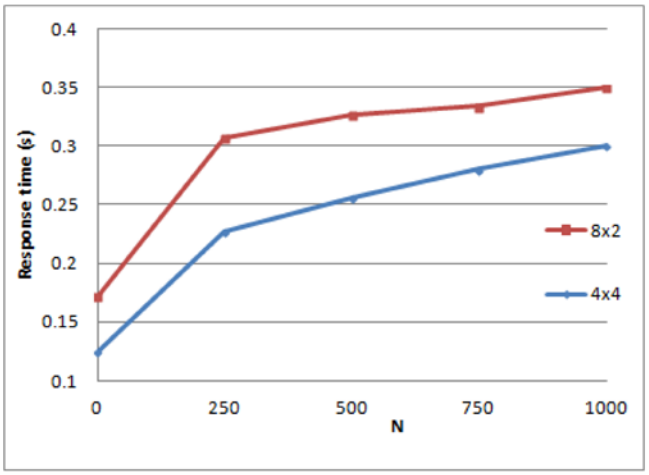

Fig. 4. Throughput performance

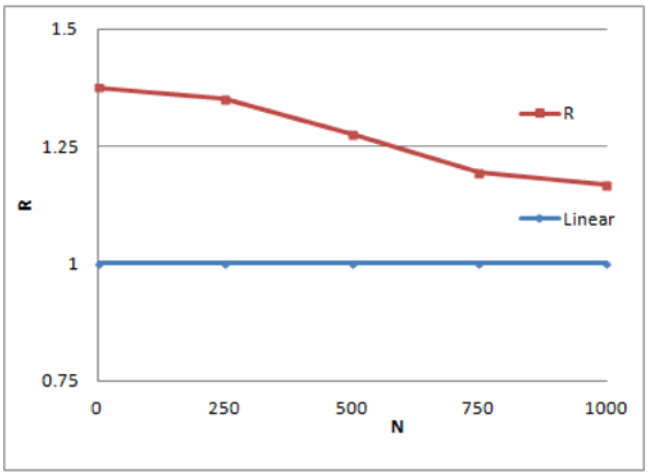

Fig. 5. Relative throughput

B. Relative Throughput R

Better comparison of both cloud environments is depicted in Figure 5, which depicts the relative throughput R as a function of a certain throughput. We clearly observe that the relative throughput $R > 1$ for each throughput N and it continuously declines when the throughput increases. The maximum achieved relative throughput is $R_{max} = 1.38$ for 1 message in a second, while the average R for conducted test cases is $R_{AVG} = 1.273$.

C. What about the Customers' Cost?

The results show that using smaller number of greater VMs (4x4) achieves better performance rather than using greater number of smaller VMs (8x2), when the same total amount of resources is used. Now, let's analyze what is the price for both environments. Does the price follow the performance, i.e., does a client pay less or more if he/she hosts the 3-tier application in 4x4 or 8x2 environment.

TABLE I
THREE YEAR COST REPORT USING PLANFORCLOUD

Instances x Cores	Three year cost
4x4	78,584.52 $
8x2	83,319.24 $

In order to measure the deployment costs, we used the cloud cost calculator PlanForCloud [11], which provides a detailed 3-year cloud cost forecast for each cloud deployment simulation. By four Medium VM and Web/Worker Role, plus SQL Server Web were used for 8x2 environment, and by four Large VM and Web/Worker Role, with SQL Server Web for 4x4 environment.

The simulation for storage included 1000 reads and 500 writes per month, and 5GB for Drive, BLOB, Queue Storage and Table Storage. Database type was Web Edition (up to 5GB, 24h/day; 1000 transactions/month), while data transfer included 30GB/180GB to/from VMs, 20/120 to/from web/worker roles, and by 100GB monthly for blob.

The three year costs for both combinations are shown in Table I. We can conclude that not only using smaller number of greater instances is better environment considering the performance, but it is cheaper solution, as well. That is, the 8x2 environment is 1.06 times more expensive.

We must note that although the price for renting compute VMsfor IaaS (Infrastructure as a Service) is linear, the price is not linear for PaaS and SaaS because the price of VM with SQL server is not linear.

V. Conclusion and Future Work

Windows Azure offers different resource organizations for application instances on SaaS layer. The client can choose between the smaller number of greater VMs or the greater number of smaller VMs. In this paper, we analyze the behavior of 3-tier cloud SaaS application in both environments, represented in performance and cost as a function of a certain throughput.

The results show that the environment with smaller number of greater application instances achieves much better results than its counterpart with the same amount of the total resources. The gap between the performance of both environments is greater for smaller throughput, and it reduces when the throughput increase.

The same environment is better also for the cost. That is, for 3 year simulation, using the greater number of smaller VM instances is 1.06 times more expensive. Therefore, the overall conclusion of this paper is to use greater instances for SaaS applications in Windows Azure, as to achieve better performance for less cost.

Windows Azure's pricing model differentiates also for Linux-based instances, which are cheaper. We will analyze the cost-performance ratio for some other platform independent applications, which can be hosted in different platforms in Windows Azure in order to determine the best platform considering the the performance and the price.

REFERENCES

[1] D. Agarwal and S. K. Prasad, "Azurebench: Benchmarking the storage services of the azure cloud platform," in *Proceedings of the 2012 IEEE 26th International Parallel and Distributed Processing Symposium Workshops & PhD Forum*, ser. IPDPSW '12. IEEE Computer Society, 2012, pp. 1048–1057.

[2] M. Gusev and S. Ristov, "Superlinear speedup in windows azure cloud," in *2012 IEEE 1st International Conference on Cloud Networking (CLOUDNET) (IEEE CloudNet'12)*, Paris, France, 2012, pp. 173–175.

[3] ——, "Resource scaling performance for cache intensive algorithms in Windows Azure," in *Intelligent Distributed Computing VII*, ser. SCI, F. Zavoral, J. J. Jung, and C. Badica, Eds. Springer International Publishing, 2014, vol. 511, pp. 77–86.

[4] M. Gusev, S. Ristov, G. Velkoski, and M. Simjanoska, "Optimal resource allocation to host web services in cloud," in *Cloud Computing (CLOUD), 2013 IEEE 6th International Conference on*, June 2013, pp. 948–949.

[5] Z. Hill, J. Li, M. Mao, A. Ruiz-Alvarez, and M. Humphrey, "Early observations on the performance of Windows Azure," in *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*, ser. HPDC '10, 2010, pp. 367–376.

[6] R. Iakymchuk, J. Napper, and P. Bientinesi, "Improving high-performance computations on clouds through resource underutilization," in *Proc. of the 2011 ACM Symp. on Applied Computing*, ser. SAC '11. USA: ACM, 2011, pp. 119–126.

[7] A. JMeter, "Functional testing tool for load testing," Mar. 2010. [Online]. Available: http://jmeter.apache.org/

[8] Y. Koh, R. Knauerhase, P. Brett, M. Bowman, Z. Wen, and C. Pu, "An analysis of performance interference effects in virtual environments," in *Performance Analysis of Systems Software, 2007. ISPASS 2007. IEEE International Symposium on*, april 2007, pp. 200 –209.

[9] W. Lu, J. Jackson, J. Ekanayake, R. S. Barga, and N. Araujo, "Performing large science experiments on azure: Pitfalls and solutions," in *CloudCom'10*, 2010, pp. 209–217.

[10] Microsoft, "Picture gallery service," Apr. 2008. [Online]. Available: http://phluffyfotos.codeplex.com/

[11] RightScale, Inc. (2013, Apr.) Plan for cloud. [Online]. Available: http://www.planforcloud.com/

[12] S. Ristov and M. Gusev, "Choosing optimal platform to host web services with xml signature and xml encryption," *J. of Next Generation Information Technology (JNIT)*, vol. 4, no. 5, pp. 110–124, July 2013.

[13] E. Roloff, F. Birck, M. Diener, A. Carissimi, and P. Navaux, "Evaluating high performance computing on the windows azure platform," in *Cloud Computing (CLOUD), IEEE 5th Int. Conf. on*, 2012, pp. 803–810.

[14] V. Subramanian, H. Ma, L. Wang, E.-J. Lee, and P. Chen, "Rapid 3d seismic source inversion using Windows Azure and Amazon EC2," in *Services (SERVICES), IEEE World Congress on*, 2011, pp. 602–606.

[15] R. Tudoran, A. Costan, G. Antoniu, and L. Bougé, "A performance evaluation of azure and nimbus clouds for scientific applications," in *Proceedings of the 2nd International Workshop on Cloud Computing Platforms*, ser. CloudCP '12. ACM, 2012, pp. 4:1–4:6.

[16] W. Wang, X. Huang, X. Qin, W. Zhang, J. Wei, and H. Zhong, "Application-level cpu consumption estimation: Towards performance isolation of multi-tenancy web applications," in *Cloud Computing (CLOUD), 2012 IEEE 5th Int. Conf. on*, june 2012, pp. 439 –446.