Cost Optimization Strategies for Running Java Applications on AWS

Hareesh Kumar Rapolu

hareeshkumar.rapolu@gmail.com

Abstract

The research paper has analysed the different ways in which JAVA applications are run efficiently on the Amazon Web Services platform. In order to ensure that the applications are running smoothly, it requires constant monetary support. Therefore, it becomes necessary to implement a few cost optimisation strategies to lower the the maintenance cost. The study has evaluated the different categories of JAVA applications. In the final section of the paper, different cost optimisation techniques and their respective benefits have been thoroughly identified. The derivations from this study can be crucial forboth individuals and businesses.

Keywords: Cloud computing, JAVA applications, Cost optimisation, AWS

I. INTRODUCTION

Different types of individuals and companies use AWS, which is one of the leading cloud computing services. With the use of this, they are able to scale their business and conduct their operations in a realiable and affordable manner. A wide range of JAVA applications can be run in AWS. However, it takes a lot of money to manage these services. Therefore, they need to implement a range of cost optimization strategies to manage the JAVA applications properly. This particular research work will analyse the different ways in which different kinds of JAVA applications can be run on the AWS platform.In the final portion of the paper, it will also explore the different advantages that that a user can gain from using the cost optimisation strategies. The study will provide a lot of insights that can be useful to both individuals and businesses.

II. EXPLORING THE CONCEPT OF JAVA APPLICATIONS

The JAVA programming language is mainly used to code different JAVA applications. The flexibility and adaptability of the applications make them so attractive. All the JAVA applications can properly run on different kinds of operating systems. Therefore, the user do not face much trouble crafting and running these applications on their hardware. The JAVA applications can be segregated into four different types, which are enterprise, standalone,web and mobile applications. All of them have different purposes and uses cases where they can be properly utilised. The standalone JAVA application is able to run on a server without the need for any particular type of operating system[1]. On the other hand, a web JAVA applications only runs on the web serverit is compatible with. The enterprise JAVA applications serve the needs of a small or a large enterprise. These applications are fundamental in ensuring the enterprise's success. Finally, mobile JAVA applications are mainly used for the purpose of cross-platform development[2]. It offers the developers a lot of scope to make these applications highly portable and secure.

III. HOW JAVA APPLICATIONS ARE RUN ON AWS

Different types of JAVA applications are run on AWS in several ways. These methods different from one another on the basis of a few factors like cost, scalability and other kinds of conditions. In tthe first method, Amazon EC2 can be used to run a JAVA application on a virtual machine. In this regard, am Amazon Machine Image is required to properly set up and boot the Amazon EC2 instance[3]. After this, JAVA with all the required dependencies are to be installed in order to run the application on the virtual machine. The JAVA application is then deplayed in the machine. The appliction traffic is properly managed with the help of an Elastic Load Balancer. It is important to mention that the auto-scaling option needs to be turned on for ensuring the the application is highly available[4].In another method, JAVA applications can be run inside AWS containers. For this particular method to work, the JAVA application needs to be packaged in a Docker container[5]. It is then deployed on Amazon ECS like Fargate or Amazon EKS like Kubernetes. The AWS Elastic Beanstalk console or the Elastic Beanstalk command line interface (EB CLI) can also be used to run JAVA applications.

Figure 1: Logo of Amazon Web Services

IV. IDENTIFYING THE COST OPTIMIZATION STRATEGIES FOR RUNNING JAVA APPLICATIONS ON AWS

An individual or a company needs to be highly aware of their cloud expenses in order to operate efficiently. Therefore, they have to consider a number of factors and implement a variety of strategies to select the right cloud computing plan for their business requirements. In this regard, the percentage of total cost reduction is measured by subtracting the changed cost from the initial one and dividing the product by the initial cost. This entire product is then multiplied by 100 to arrive at the desired result. Furthermore, the total cost of ownership is measured by adding the complete cost required for computation, networking, storage, monitoring and licensing. There are different strategies that are used for minimising the overall cloud expenses on AWS while running JAVA applications on it.

Right-sizing of resources

A particular JAVA application requires a definite amount of resources. Therefore, the individual or business running the application needs to have complete knowledge of the actual needs of the application. In this manner, they can allocate their resources properly and avoid over-provisioning. Therefore a lot of money can be saved, which can ultimately hamper the position of the involved business or the individual. In order to do this, different kinds of monitoring tools can help to gather important insights on the instance types and sizes that are required for running the Java applications on AWS.

Auto-scaling

This can be described as an automatic service of AWS that helps the individual or a business to adjust the total number of instances in real-time. The resources are automatically adjusted on the basis of the needs of the one running the JAVA applications[6]. This is instrumental for saving a lot of money and managing unpredictable workload pressures.

Regular monitoring and optimisation of cloud storage

The business or individual has to implement some data lifecycle policies that can help to identify which kind of data requires which storage unit. This categorisation can be beneficial for minimising the cloud computing cost.

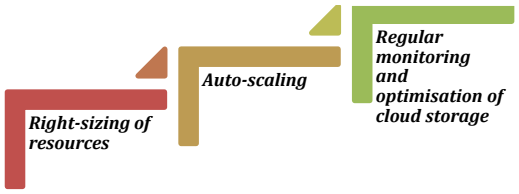

Figure 2: Cost optimization strategies for running JAVA applications on AWS

V. BENEFITS OF THE COST OPTIMIZATION STRATEGIES FOR RUNNING JAVA APPLICATIONS ON AWS

Improved resource utilisation

The cost optimization strategies help in appropriate resource management. Therefore, the individual or company is able to allocate resources efficiently according to their demands and requirements.

Enhanced security of the JAVA applications

If these strategies are properly implemented, the security of the JAVA applications can be enhanced. This is mainly due to the better resource allocation and storage unit choices[7].

VI. CONCLUSION

From the discussion, it can be concluded that an individual or a business needs to implement proper cost optimization strategies like right-sizing of the resources, auto-scaling and monitoring of the cloud storage. These strategies can not only help the JAVA applications to run seamlessly but also enable the business to get the best results out of them and achieve their goals.

Abbreviations and acronyms

- AWS - Amazon Web Services
- EC2 - Amazon Elastic Compute Cloud
- ECS - Elastic Container Service
- EKS - Elastic Kubernetes Service
- EB CLI - Elastic Beanstalk command line interface

Units

- Memory - GB or MB
- Memory utilisation and CPU utilisation - %

Equations

- Cost Reduction(%)=(Cbefore−Cafter/Cbefore)×100
- TCO=Ccompute+Cstorage+Cnetwork+Clicensing+Cmonitoring

REFERENCES

[1] G. Bollella and R. Belliardi, *The Real-Time Specification for Java*, vol. 33. Computer, Jun. 2000. Available: http://www.staff.city.ac.uk/c.kloukinas/IN2P3/material/rtsj-V1.0.pdf

[2] G. Juve, E. Deelman, G. B. Berriman, B. P. Berman, and P. Maechling, "An Evaluation of the Cost and Performance of Scientific Workflows on Amazon EC2," *Journal of Grid Computing*, vol. 10, Mar. 2012, doi: https://doi.org/10.1007/s10723-012-9207-6

[3] I. H. Kazi, H. H. Chen, B. Stanley, and D. J. Lilja, "Techniques for obtaining high performance in Java programs," *ACM Computing Surveys*, vol. 32, no. 3, pp. 213–240, Sep. 2000, doi: https://doi.org/10.1145/367701.367714

[4] J. Baron and S. Kotecha, "Storage Options in the AWS Cloud," *Amazon Web Services*, Oct. 2013, Available: https://www.academia.edu/download/62020653/storage_options_in_the_aws_cloud_020200206-75635-320zzw.pdf

[5] M. S. Aslanpour, M. Ghobaei-Arani, and A. N. Toosi, "Auto-scaling web applications in clouds: A cost-aware approach," *Journal of Network and Computer Applications*, vol. 95, pp. 26–41, Oct. 2017, doi: https://doi.org/10.1016/j.jnca.2017.07.012

[6] S. Mathew, "Overview of Amazon Web Services," *Amazon Whitepapers*, vol. 105, no. 1, Nov. 2014, Available: https://www.sysfore.com/Assets/PDF/aws-overview.pdf

[7] W. S. El-Kassas, B. A. Abdullah, A. H. Yousef, and A. M. Wahba, "Taxonomy of Cross-Platform Mobile Applications Development Approaches," *Ain Shams Engineering Journal*, vol. 8, no. 2, pp. 163–190, Jun. 2017, doi: https://doi.org/10.1016/j.asej.2015.08.004