# ENHANCED AUTO SCALING COMPARISON BETWEEN AMAZON AWS AND MICROSOFT AZURE

N.Manimozhi[1], R.Sakthidevi[2], S.Ramya[3], G.Suguna[4], Dr.P.Senthil Pandian[5]

[1]Assistant Professor, Department of Computer Science and Application, RAAK Arts and Science College, Tamil Nādu, India.
[2]Assistant Professor, Department of Computer Science and Application, RAAK Arts and Science College, Tamil Nādu, India.
[3]Assistant Professor, Department of Computer Science and Application, RAAK Arts and Science College, Tamil Nādu, India.
[4]Assistant Professor, Department of Computer Science and Application, RAAK Arts and Science College, Tamil Nādu, India.
[5]Assistant Professor, Department of Computer Science and Application, RAAK Arts and Science College, Tamil Nādu, India.

**ABSTRACT**

*In the Modern world many companies are preferring systemize environment on their own IT business. It may not be feasible for every startup to invest huge amount of money for procuring servers, IT infrastructure and recruiting staff who can maintain these servers and IT Infrastructure. To satisfy the user demand by increasing or decreasing the virtual machine computing services. Recently Many cloud service providers provide cloud services. This paper describes about auto scaling comparison on Amazon AWS and Microsoft Azure which predicts most of the people preferring Amazon AWS cloud computing service.*

**Keyword**: - *Auto scaling, Instance, Virtual machine, Scaling Plans*

## 1. INTRODUCTION

Today, cloud computing provides a highly reliable, scalable, low-cost infrastructure platform in the cloud that powers hundreds of thousands of businesses in 190 countries around the world began offering IT infrastructure services to businesses as web service. Now commonly known as cloud computing. One of the key benefits of cloud computing is the opportunity to replace upfront capital infrastructure expenses with low variable costs that scale with your business. With the cloud, businesses no longer need to plan for and procure servers and other IT infrastructure weeks or months in advance. Instead, they can instantly spin up hundreds or thousands of servers in minutes and deliver results faster by different cloud service provider such as AWS (Amazon Web Services), Microsoft Azure, Google Cloud IBM Cloud, Oracle Cloud, Alibaba Cloud etc.

## 2. BASIC CONCEPT OF AUTO SCALING

Auto Scaling is a systematic method in cloud computing that enables organizations to automate the number of computational resources according to active users' demands. Systems will automatically set aside the right number of resources at different periods based on defined scaling rules. There are two major ways to autoscale: vertically and horizontally. Vertical scaling involves scaling resources up and down, which changes their capacity. Horizontal scaling, also known as in-and-out scaling, controls the instances of a resource.

.

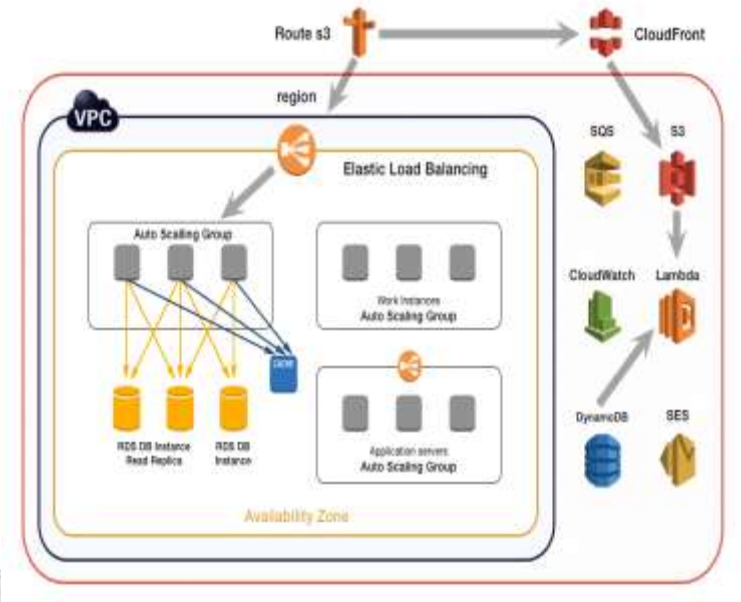


**Fig -1**: Auto Scaling Group with AWS services

**2.1 AWS Auto Scaling supports the use of scaling plans for the following services and resources:**

- Amazon Aurora – Increase or decrease the number of Aurora read replicas that are provisioned for an Aurora DB cluster.
- Amazon EC2 Auto Scaling – Launch or terminate EC2 instances by increasing or decreasing the desired capacity of an Auto Scaling group.
- Amazon Elastic Container Service – Increase or decrease the desired task count in Amazon ECS.
- Amazon DynamoDB – Increase or decrease the provisioned read and write capacity of a DynamoDB table or a global secondary index.
- Spot Fleet – Launch or terminate EC2 instances by increasing or decreasing the target capacity of a Spot Fleet.

**2.2 Working with scaling plans create, access and manage your scaling plans using any of the following interfaces:**

- AWS Management Console – Provides a web interface that you can use to access your scaling plans. If you've signed up for an AWS account, you can access your scaling plans by signing into the AWS Management Console, using the search box on the navigation bar to search for AWS Auto Scaling, and then choosing AWS Auto Scaling.
- AWS Command Line Interface (AWS CLI) – Provides commands for a broad set of AWS services, and is supported on Windows, macOS, and Linux. To get started, see AWS Command Line Interface User Guide. For more information, see autoscaling-plans in the AWS CLI Command Reference.
- AWS Tools for Windows PowerShell – Provides commands for a broad set of AWS products for those who script in the PowerShell environment. To get started, see the AWS Tools for Windows PowerShell User Guide. For more information, see the AWS Tools for PowerShell Cmdlet Reference.
- AWS SDKs – Provides language-specific API operations and takes care of many of the connection details, such as calculating signatures, handling request retries, and handling errors. For more information, see AWS SDKs.
- Query API – Provides low-level API actions that you call using HTTPS requests. Using the Query API is the most direct way to access AWS services. However, it requires your application to handle low-level details such as generating the hash to sign the request, and handling errors. For more information, see the AWS Auto Scaling API Reference.

- AWS CloudFormation – Supports creating scaling plans using CloudFormation templates. For more information, see the AWS::AutoScalingPlans:: ScalingPlan reference in the AWS CloudFormation User Guide.

### 2.3 To build a Scalable Application upto 1 million users on AWS

Auto scalability of an application is equally important as its features and user interface. It becomes even more important if your app is going to serve more than a million users in the future. In the Fig2, you will pursue how to scale your app up to 1 million users on AWS.


**Fig – 2:** Million user Scalable Application with AWS services

## 3. MICROSOFT AZURE AUTOSCALE CONCEPTS

Autoscaling provides the capability to run your application or workload with the required resources (resources, in this case, are virtual machines) without interruption. It assures you that the virtual machines you requested for your application are always available and up. If the virtual machines are interrupted, autoscaling replaces those faulty virtual machines with new ones.

Types of autoscaling

In general, there are two types of autoscaling –

- Time-Based Autoscaling.
- Metrics-Based Autoscaling.

Time-Based autoscaling is nothing but scaling based on the scheduled time. This type needs some extent of manual prediction of your demand. For example, suppose you know that your application experiences high traffic during certain times of the day, week or month and the number of virtual machines needed to meet that demand. In that case, you can configure the rules to spin up and shut down those needed virtual machines only during that specific time period.

On the other hand, Metrics-Based autoscaling enables the scaling activity to be based on the key performance metrics of your resource like CPU, Memory, Thread Count, etc.

**Here the main concepts behind on Azure Auto scaling:**

- **Resource metrics—Azure VM scale sets use telemetry data from Azure diagnostics agents. You can** get telemetry for web applications and cloud services directly from the Azure infrastructure. You can get data about resources, including CPU and memory usage, thread counts, disk usage, and queue length.
- Custom metrics—you can configure your applications to send custom metrics to Application Insights, a feature of Azure Monitor which provides Application Performance Management (APM). You can then scale VMs according to this information.

- Rules—Azure lets you create metric-based rules and time-cased rules. Additionally, you can create as many autoscale rules as you need, and set them up to overlap during certain scenarios.
- Actions and automation—you can use rules to trigger one or multiple types of actions, including scaling VMs, sending emails to relevant stakeholders, and triggering automated actions via webhooks.
- Horizontal vs vertical scaling—autoscale uses horizontal scaling only. This means you can set up rules that increase or decrease the amount of VMs. This process provides the flexibility needed to run hundreds and thousands of VMs. Vertical scaling, on the other hand, lets you maintain the same amount of VMs, while increasing or decreasing the CPU and memory resources. You usually need to stop VMs when using vertical scaling.

### 3.1 Work of Azure Autoscaling

Azure Autoscaling initiates events based on predefined settings. You can set up rules that define how VMs should be scaled during unexpected or regular, predictable events.
You can define a scale set, which is a group of VMs with a minimum and a maximum number of instances. The minimum number of instances will always run, irrespective of loads. The maximum number is a limit, which will put a cap on the total cost per hour. The scale set automatically adjusts between these two extreme values according to the rules you set.

**When rule conditions are met, you can perform one or more auto-scaling actions, including:**

- Adding VMs (scale out)
- Removing VMs (scale in)
- Sending notifications
- Using webhooks to run other automated events, including automated runbooks, Azure Functions, and third party systems

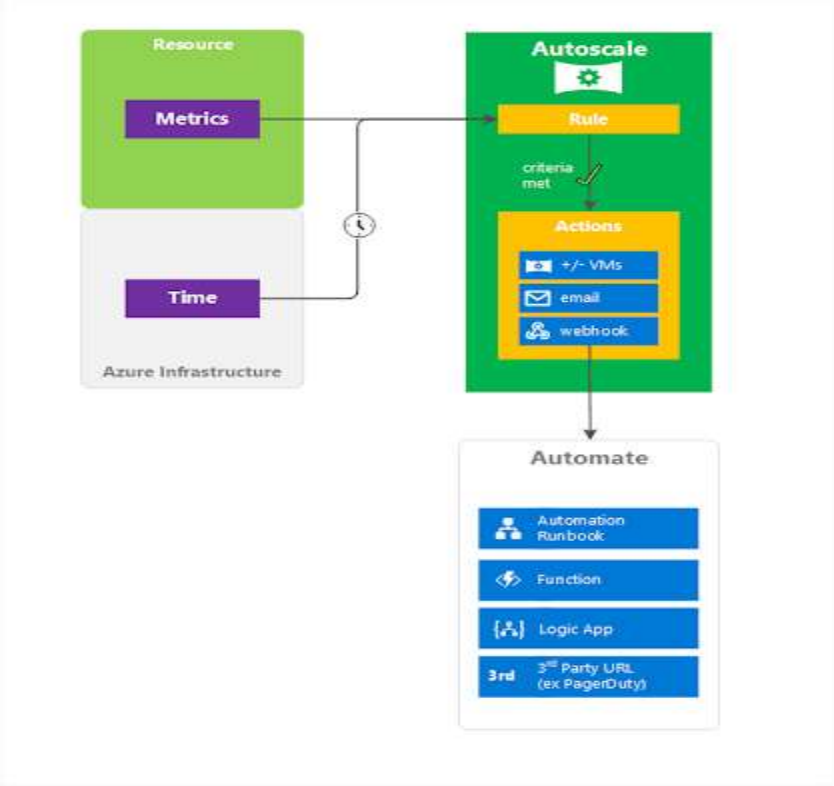


**Fig-3 Source:** Azure Auto scaling

### 3.2 Azure Autoscaling Best Practices

The following best practices will help you use Azure Autoscaling more effectively:

- If you can predict application loads, use scheduled autoscaling to add and remove instances according to known workload fluctuations.
- If you cannot predict loads, use autoscaling based on resource metrics.
- Initially provision some extra capacity, especially when starting to use autoscaling for application. This will allow you to monitor resource metrics and tune autoscaling behavior, without disrupting the application.
- Monitor and fine tune autoscaling rules. Keep in mind that autoscaling is an iterative process—it will take some trial and error to identify the resource metric and threshold that will achieve the best scaling behavior.
- You can have multiple rule profiles. Azure Autoscaling processes one rule profile at a time, and only after processing all custom rules, uses the default profile. This means the most important rules should be in custom profiles, if used. Within a profile, scale out is executed if any rule is met, and scale in (removing instances) is only performed if all rules are met.

The SDK is more flexible than Azure Portal, allowing more detailed scheduling options, and also lets you use custom metrics and counters as a trigger for autoscaling.

## 4. UNDERSTANDING AWS and AZURE AUTO SCALING

### 4.1 Amazon AWS

The AWS autoscaling feature is free to use and conveniently set up with the AWS Management Console, CLI (command-line interface), or SDK (software development kit). Users only need to pay additional fees for used resources and CloudWatch monitoring, which provides data and actionable insights on AWS.
Through the AWS autoscaling feature, users can look forward to scaling multiple resources across servers within a short time frame.

**Advantages:**

- Provides autoscaling groups, which enable users to categorize instances into logical groupings for more convenient scaling and management.
- Enhanced fault tolerance, driving quick response in detecting and replacing faulty instances.
- Runs predictive scaling that applies machine learning technology in estimating expected traffic for proactive provision of compute.
- The system's cooldown feature may cause inaccuracies without proper precautions. Short cooldowns may result in "over-scaling" or "under-scaling." Users need to ensure that a cooldown period equals the time taken for a metric to fulfill a scaling event.

### 4.2 Azure Autoscale:

Microsoft Azure offers a built-in autoscale feature that enables users to schedule system alerts based on any defined metric such as CPU status, user response rates, and event triggers. Azure users can benefit from key performance metrics that moderate system performance for optimal results.

**Advantages:**

- Responsive scaling allows users to scale automatically without manual administration.
- Customized metrics provide improved flexibility in autoscaling. Users may define beyond resource metrics by applying preset instances.
- . Features more availability zones (AZs) than the AWS infrastructure, reducing the likelihood of server downtime.

**Downside:**

There is a required learning curve for other programs (for example, Azure PowerShell).
Both autoscaling services provide businesses with the capabilities to optimize the cost-effectiveness of their servers. Ultimately, the choice lies in individual user needs and workload demands. Complex processes may require a multi-cloud approach. Regardless, a unified automated solution can significantly boost standard autoscaling services.

## 5. GROWTH RATE OF CLOUD SERVICE PROVIDER

According to the reported quarterly earnings for 2021, Microsoft's Azure cloud revenue has been observed to, once again, outperform both AWS and Google Cloud combined.

In spite of the Goliath-like stature of Amazon's AWS, Microsoft's Azure cloud outperformed its competitors with its US$17.7 billion (50% revenue growth over the previous quarter) in commercial-cloud revenue as per the fiscal earnings report. While Amazon's AWS reported US$13.5 billion in cloud business revenue for the quarter (revenue grew 32% in the quarter), Google Cloud had a modest US$4.05 billion.

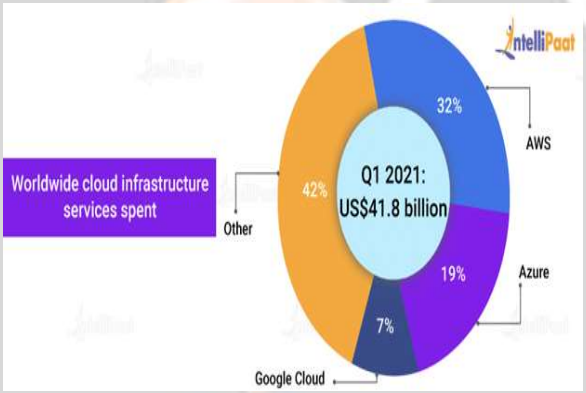


**Chart -1**: *Growth rate top of the cloud Vendors*

## 6. CONCLUSION

Through Autoscaling, organizations can optimize steady and predictable server performance at the lowest cost. AWS and Azure are two of the biggest names in cloud computing that offer built-in autoscaling capabilities providing an excellent reason for cloud migration over on-premise solutions. This paper outlined a mostly used in cloud computing service provider of Amazon AWS. This paper explains about the hope and also considered as a starting point identifying opportunities for future reference. A report by Canalys mentions that as of April 2021, the global cloud market grew 35% this quarter to $41.8 billion. AWS covers 32% of the market, followed by Azure at 19% and Google at 7%.

## 7. REFERENCES

[1] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A. & Zaharia, M. (2010). A view of cloud computing. Communications of the ACM, 53(4), 50-58.

[2] Moyo, Tumpe, and Jagdev Bhogal. "Investigating Security Issues in Cloud Computing." Complex, Intelligent and Software Intensive Systems (CISIS), 2014 Eighth International Conference on. IEEE, 2014.

[3] R. Buyya, C. S. 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 Generation Computer Systems, Vol. 25, No. 6, pp. 599-616, 2009.

[4] Y. F. Li, W. Li, and C. F. Jiang, "A Survey of Virtual Machine System: Current Technology and Future Trends," IEEE International Symposium on Electronic Commerce and Security, 2010.