# Automate the Deployment of Any Custom Web Applications Across Several Clouds using Kubernetes Clusters

**Ms. Amrapali Chavan[1], Ms. Mansi Deshmukh[2], Mr. Vishnu Parikh[3],**
**Mr. Ajinkya Khandave[4], Ms. Suvarna Pawar[5]**
Faculty, Department of Computer Engineering[1]
Students, Department of Computer Engineering[2,3,4,5]
AISSMS Institute of Information Technology, Pune, Maharashtra, India
amrapali.chavhan@aissmsioit.org@gmail.com, mansid411@gmail.com,
vishnuparikh12345@gmail.com, ajinkyaakap11@gmail.com, pawarsuvarna2017@gmail.com

**Abstract:** *During COVID period, most the businesses are worked in online mode. For deploying their businesses need to keep their web apps up all throughout the clock. Every successful small-scale business has its own website or [3 ]web application. To deploy the website with every modification needed, a team of DevOps Engineers is needed. The role of those engineers is to take the website, run it on a server and make it available to the end-user. This entire process can therefore be automated in a single click with this system. In this system, we are integrating the DevOps tools such as Ansible, Terraform, Kubernetes, Docker and Cloud Servers. This system when used will save more time and would be much more efficient. It will help businesses cost-effectively achieve zero downtime. Along with these benefits, their website or web app would be secured as it would be deployed and monitored on Cloud. This system will help businesses to grow more.*

**Keywords:** Ansible, AWS, Azure, Docker, Kubernetes

## I. INTRODUCTION

DevOps is a set of practices that aims to reduce the barriers between development and operations teams. The function that development team members are planning, analysis, programming, and other activities throughout the project. Operations engineers install, configure, operate, and evolve common infrastructures such as the network, servers, and your external services (e.g. the cloud). So, the Dev in DevOps is for Developers whose main work is to develop a product, and the Ops in DevOps is for Operations Engineer whose job is to monitor, re-lease, deploy and operate. During the COVID period, most of the businesses are functioning in online mode. People running their businesses need to keep their web apps up all throughout the clock. It is much necessary to use the automated scripts. Businesses and freelancers will be able to use this automated procedure to deploy their web apps across different clouds for scaling, stability, durability, reusability, and security. During the COVID period, most of the businesses are worked in online mode. For deploying their businesses need to keep their web apps up all throughout the clock. Everyone in the IT sector requires their apps, whether web or mobile, to be deployed and available at all times.

For an app to function with all its available functionalities, it must be able to handle traffic from all over the world and must be available at all times. So, for the app to be always up and running without any downtime, it can be deployed on several clouds using Terraform and then managed using Kubernetes in those instances also known as servers or nodes, thus allowing web-app users to access it from anywhere. It is a time- consuming process to manually deploy an app. But, if we use DevOps Automation, it will be faster, more reliable, and more efficient. In a production environment, you need to manage the applications and ensure that there is no downtime. For example, if a container goes down, another container needs to start. Wouldn't it be easier if this behavior was handled by an automated system? The deployments are automatically managed which means if a container having the web application goes down then the deployments automatically create the same one.

DOI: 10.48175/IJARSCT-7335

## II. METHODOLOGY

### 2.1 Bare Metal

A physical server devoted to a single tenant is known as a bare metal server. The server's tenant can optimize the server to meet its performance, security, and reliability requirements. A hypervisor server, which allows several users to share a virtual server's computation, storage, and other resources, is an alternative to a bare metal server. Bare metal servers are also known as single-tenant server farms or managed dedicated hosting. On a bare metal server, the operating system is placed directly on the server, removing layers and continuous improvement.

### 2.2 Virtualization

Virtualization is the bare metal hypervisor which is the most commonly deployed type of hypervisor. Virtualization is made feasible by virtualization, which translates requests between physical and virtual resources. A bare-metal virtualization is installed directly on a real server's hardware, between both the hardware and the software (OS). At the same level as the motherboard's basic input/output system, certain bare-metal hypervisors are incorporated in the firmware (BIOS). For some systems, this is required in order for the operating system to acquire and use virtualization software. The program no longer relies on or is confined to certain hardware devices or drivers since the bare metal virtualization separates the OS from the hardware resources. As a result, bare metal VM’s enable operating systems and their related applications to run on a wide range of hardware.

### 2.3 Cloud

Cloud hosting is the concept of outsourcing an organization's disk storage infrastructure to a service supplier that delivers infrastructure on a utility basis. The cloud provider is in charge of the installation, cloud infrastructure, security, and maintenance, with clients having the option to customize hardware and applications as well as scale servers online. In a cloud infrastructure architecture, compute and storage resources are distributed across hundreds of virtual servers (VMs) that load balance I/O needs. [4]Cloud hosting is a less expensive alternative to traditional dedicated server hosting, which requires businesses to create and run their own data centers. Servers and storage, with hardware platforms and virtual resources, remain on-premises in the traditional paradigm and can be a costly investment and ongoing expense for businesses.

### 2.4 Containerization

Containers have increased in popularity as a major standard for virtualizing physical resources in cloud data centers since 2013. Containers employ runtime engines like Docker, Apache Mesos, or Hyper run for virtualization partitions, whereas VPS partitions use hypervisor software to sharing OS kernel support on a web server. Through automation, web server stack software disc images are installed in elastic cluster orchestration. Containers are used in web hosting to provide multi-tenant datacentre system throughput with network resource allocation that go beyond a single single hosting in web usage support.

### 2.5 Containerization in Cloud Computing

Containers are a type of operating system virtualization used in cloud computing. This allows the user to work with a program and its dependencies via segregated resource procedures. The application's code can be grouped into bundles with configurations and dependencies.[5]Containers are used in cloud computing to provide building blocks that help with operational excellence, versioning, developer reliability, and environmental consistency. As a result, the user may anticipate dependability, consistency, and speed regardless of the distributed platform. The infrastructure has been improved since it allows for more control over resource granular activities. The use of containers in online services improves storage security, reliability, and elasticity in cloud computing.

## III. ADVANTAGES AND DISADVANTAGES

### 3.1 Advantages of a Container in Cloud Computing

- **Cloud Storage Consistency:** It eliminates organisational and technical stumbling obstacles, allowing the program to finish the entire procedure. It provides a base for the files and needs that programs and software

DOI: 10.48175/IJARSCT-7335

servers require. This is something that can be distributed across any resource. As a result, the manual configuration of each server is totally avoided, letting people to introduce a new functionality.

- **Version Control for Applications:** Users can see the latest iteration of the program code as well as their requirements using containers in cloud computing.

### 3.2 Disadvantages

The count of servers will be defined by the client, if the client does not know the exact traffic generation we have to approximately allocate the servers to various data centers, this might cause wastage of resources. In the first stage, this product is only compatible with websites and will not be able to deploy mobile and desktop applications.

## IV. IMPLEMENTATION

### 4.1 Creation of Slave and Master Node

This section with thr help of [7] it gives the output of each phase in the process. On the Azure cluster, a Kubernetes cluster including one master as well as other slave nodes was built first. The nodes are all of kind t2.micro and are in the us- east-1 area. It took roughly a few minutes to complete this operation. We are launching all the instances on various cloud platforms using terraform which is an open source tool.We are configuring mater and slave node using ansible via various modules present in ansible.

### 4.2 Master Node Configuration

Two servers make up our Kubernetes cluster. [8]One of them will take on the role of Kubernetes Master and other will be worker node. The servers are all connected to the same local network. Kubernetes, Docker, and Nginx-ingress-controller are the software components that the cluster relies on.

We are configuring ansible in the master node cluster and generating SSH key on a master node after that copy the SSH key from the master node to the worker or slave node for connection establishment. At the end set up the cluster using yaml file via ansible-playbook. The master node controller is present for maintaining podes.

### 4.3 Slave Node Configuration

Worker node present in the cluster of azure where ssh key of master is should be copied. In slave node network configuration and pods are present with other addons like DNS, UI, etc. which are useful for outside connection with the internet.

## V. EFFICIENCY OF KUBERNETES CLUSTER

Cloud service providers are adopting container-based virtualization technologies like Docker and Kubernetes because they are easier to deploy, perform better, and have a smaller memory footprint than hypervisor-based virtualization. Basic replication for availability is supported by Kubernetes, although it does not ensure integrity and consistency and may damage the application state in the event of a failure. This paper describes a Kubernetes state machine replication technique that delivers high availability, integrity, and consistency. Replica coordination is offered as a service, with simple application integration. [1],[2] gives the solution's feasibility is demonstrated by the experimental results. The frequency with which you modify your programme and provide new features for users. When updating software or launching a significant deployment back in the day, there have been a lot of downtimes, and they were usually done after midnight or on the weekends when user traffic was minimal. When adding new features on a regular basis, however, bare in mind that delay does not increase velocity

Containers and Kubernetes empower developers to create distributed systems based on immutable infrastructure principles. An artifact developed in an immutable infrastructure will not be modified by user modifications. With changeable infrastructure, the typical way of doing things was to allow change to arrive on top of the existing entities as continuous updates. As a result, the current state of the infrastructure could be represented as a single entity, but rather as a collection of incremental updates and modifications. When kubernetes receives a desired state configuration, It doesn't simply take a single step to match the current state to the desired state; it takes steps to ensure that it stays that way throughout time.

DOI: 10.48175/IJARSCT-7335

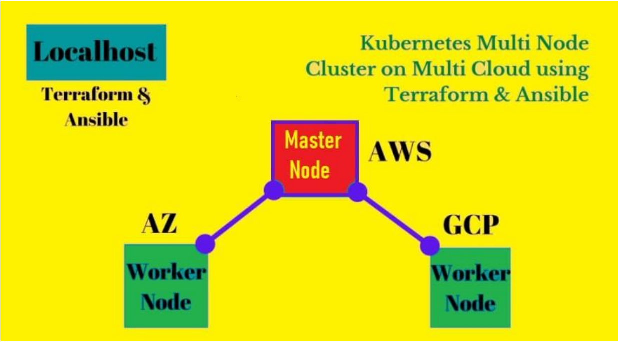

**Figure 3**: Multinode Cluster

## VI. CONCLUSION

The entire product is focused to ease the process of deployment of Web Applications. These web apps will be deployed over multiple major clouds such as AWS, Azure, GCP, etc. This will aid in lowering the failure rate and increasing the overall system's reliability. This process of deployment is carried out using various DevOps tools and hence requires a lot less time which will help developers to focus on development and not on the operations part. The infrastructure will be able to scale at any time without any human involvement thanks to the Kubernetes Cluster, resulting in a speedier delivery process.

## ACKNOWLEDGEMENT

The authors would like to thank the Principal of AISSMS IOIT Dr. P.B. Mane for their constant support during the project.

## REFERENCES

**[1].** Arnaldo Pereira and Ferreira; Richard Sinnott,"A Performance Evaluation of Containers Running on Managed Kubernetes Services", 2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), Sydney, NSW, Australia, 2019, doi:10.1109/CloudCom.2019.00038.

**[2].** Ruchika Muddinagiri; Shubham Ambavane; Simran Bayas, "SELF HOSTED KUBERNETES: DEPLOYING DOCKER CONTAINERS LOCALLY WITH MINIKUBE", 2019 International Conference on Innovative Trends and Advances in Engineering and Technology (ICITAET), Shegaon, India, 2019, doi: 10.1109/ICITAET47105.2019.9170208.

**[3].** Daniel Vladusic, Dragan Radolovic, Infrastructure as Code for Heterogeneous Computing, 2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2020, doi: 10.1109/ SYNASC51798.2020.00011.

**[4].** T. Deepa, Dr. Dhanaraj Cheelu, "A Comparative Study of Static and Dynamic Load Balancing Algorithms in Cloud Computing", International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017), Chennai, India, 2018, doi: 10.1109/ ICECDS.2017.8390086.

**[5].** Aayush Agarwal; Subhash Gupta; Tanupriya Choudhury, "Continuous and Integrated Software Development using DevOps", 2018 International Conference on Advances in Computing and Communication Engineering(ICACCE), Paris, France, doi:10.1109/ICACCE.2018.8458052.

**[6].** Nikhil Marathe, Ankita Gandhi, Jaimeel M Shah, "Docker Swarm and Kubernetes in Cloud Computing Environment", 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, doi: 10.1109/ ICOEI.2019.8862654.

**[7].** Leonardo Reboucas de Carvalho; Aleteia Patricia Favacho de Araujo, Performance Comparison of Terraform and Cloudify as Multicloud Orchestrators, 2020, doi: 10.1109/ CCGrid49817.2020.00-55.

**[8].** Nishant Kumar Singh, Sanjeev Thakur, Himanshu Chaurasiya, Himanshu Nagdev, Automated provisioning of application in IAAS cloud using Ansible configuration management, 2016, doi: 10.1109/NGCT.2015.7375087.