*International Journal of Interdisciplinary Innovative Research & Development (IJIIRD)*
**ISSN: 2456-236X**
Vol. 03 Issue 01 | 2018

# Cloud Containers

[1]Ashish C. Verma, [2]Roshan M. Jaiswal
[1, 2] *Department of Computer Science Mumbai Educational Trust College, Mumbai, India*

**ABSTRACT**

***The management of an application in Cloud environment is particularly managed by light weight technology which known as containerization, have recently been successful. Containerization overcome the downside of virtual machine in which deploying and running distributed application without making run the entire virtual Machine for each and every application. A large amount of power is consumed to execute these computational units but containerization technology provide the better way in order to achieve low power consumptions in cloud environment which is also environment Friendly.***

***Keywords****-Virtual machines (VM), Container, Orchestration, Container, Cluster, Cloud and Virtualization*

## 1. INTRODUCTION

Containerization is an Operating System level Virtualization technique which is mainly for deploying and running distributed application without making run the entire virtual Machine for every application. Instead, a single control host is used to run the multiple isolated system and access a single kernel which provides huge benefit in terms of cost reduction and provides Lowering Power Consumption in Cloud Environment. Components such as libraries, files and environment variables which are required to run the particular software are hold by Container.

In containerization, the Different Container uses the Same Operating System. Unlike, Using Cloned Operating System For Each Virtual Machine which Tend To increase in cost .Containerization also provide the Solution to mobile security issue for multi-use phones, wherein employees are allowed to use their personal device in many companies .However, containerization could work to wall off sensitive data of company on a multi-use phone, it would not work against a certain kinds of compulsion inherent in the mobile device O/S itself or jail breaking.

To achieve elasticity in large-scale shared resources ,cloud uses virtualization technique which provide the lightweight alternate that has resulted in a significant uptake in cloud application management . Virtual Machine plays a most important role in virtualization technique which is typically a backbone of infrastructure layer.

This paper is structured as follows. Section II describes the architecture of the container and how the container is managed. Section III explains about the virtual machine and cloud containers and also provides a characterization framework for cloud container orchestration.

## 2. CONTAINER ARCHITECTURES AND THEIR MANAGEMENT

Containerization is an O/S level Virtualization technique. To achieve elasticity in large-scale shared resources, cloud uses virtualization technique which provides the lightweight alternate that has resulted in a significant uptake in cloud application management. Virtual Machine plays a most important role in virtualization technique which is typically a backbone of infrastructure layer.

### 2.1 Container Technology Principles

A container holds loaded self-contained, ready-to-deploy parts of applications and, if required middleware and business logic to run the applications. Docker is a type tool which is built around container engines where containers act as portable means to package applications. This provides the outcome in the need to manage

dependencies between containers in multi-tier applications. An orchestration plan can describe components, their dependencies and their lifecycle in a layered plan. A PaaS cloud can then execute the work flows from the plan through agents like a container engine. PaaS clouds can consequently support the deployment of applications from containers. Orchestration subsumes here their coordinated construction, deployment and ongoing management.

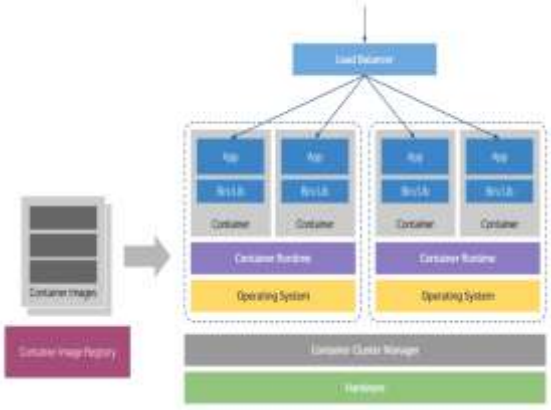


Fig.1 : Container Cluster Architectures

### 2.2 Cloud-based Container Architectures

Container orchestration not only deals with making the applications start or stop and moving them among servers. Orchestration is described as constructing and continuously managing possibly distributed clusters of container based applications .When the multiple number of container applications are deployed, Container Orchestration permit the users to describe how to correlate the containers in the cloud environment. Container orchestration not only describes the initial deployment of containers, but also the management of the multiple containers as a single unit. Availability, Scaling and Networking of containers are also taking cared by the Container orchestration. Essentially cloud-based container construction is a form of orchestration within the distributed cloud environment.

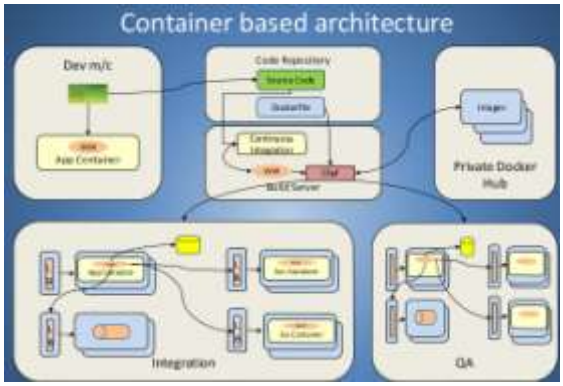



Fig.2: Container Architectures

## 3. CLOUD CONTAINER AND VIRTUAL MACHINE

### 3.1 Cloud Container

Cloud Container is all about putting the entire thing together and deploying it at once. Containers are similar to Virtual Machines in that they provide isolation and separate space for app to execute in memory and storage to reside and also provide the appearance of individual system, so that each container can have its own groups of users and system admins. Isolation means that if any of the containers having any issue i.e. excessive consumption of the resources by the process to achieve any task, it doesn't affect the other containers which

getting executed parallels. It also resolves the compatibility issues between the applications that reside on the same operating system.

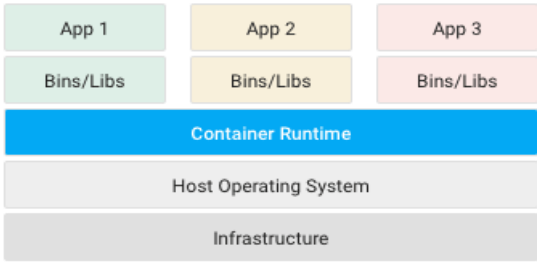
Operation System Level Virtualization methodology are known as container-based virtualization technique and application containerization is usually for deploying the distributed applications and running it without launching an whole Virtual Machine for Each and every application which is also Known As Containerization.

Unlike a Virtual Machine, in a container they don't need a full Operating System to be installed within the container, you are not running a complete instance or image of operating system drivers, with kernels and shared libraries and they don't need a virtual copy of the host server's hardware. Containers provide the ability to perform the task with the minimum amount of resources; this can mean just a few pieces of software, libraries and the basics of an OS. This results into as many containers being able to be deployed on a server than virtual machines.

Instead, a whole stack of containers, whether it be hundreds or even thousands are be able to run on top of the single instance of the host OS, in a tiny fraction of time of a comparable VM running the same application. Cloud containers are also very portable -- once the container has been created, it can be deployed to different servers very easily. From a software lifecycle perspective this is great, as containers can be copied to create development, test, and integration and live environments very quickly, and do not require the usual configuration. From a software- and security-testing perspective this has a large advantage, because it ensures that the underlying OS is not causing a difference in the test results.

One drawback of containers is the complication of dividing your virtualization into lots of smaller chunks. When there are just a few numbers of containers which are included, it's an advantage since you know exactly what configuration you're deploying and where. However, if you fully invest in containers it's quite possible to soon have so many containers that it becomes difficult to manage. Do you have the ability to deploy patches to hundreds of different containers? If a specific library needs updating inside a container because of a security concern, do you have an easy way to do this? Problems of container management are a common complaint, even with container management systems such as Docker. Virtual machines are generally considered easy to manage, primarily because there are significantly fewer VMs compared to containers.

Containers can be deployed in one of two ways: either by downloading a previously created image, such as from Docker Hub or by creating an image to run in a container. Although Docker is the company which is the only container platform provider to address every application across the hybrid cloud and provide the service that helps to drive the container movement and the only container platform provider to address every application across the hybrid cloud. However, Docker has become closely associated with containerization. Originally built on a technology called LXC, Docker has become the predominant force in the world of containers. The library of previously created images in Docker Hub is huge , and must allow most standard specification to be met with minimal effort.




**Containers**

### 3.2 Virtual Machine

Virtual machines are an abstraction of physical hardware turning one server into many servers. The hypervisor allows multiple VMs to run on a single machine. Each Virtual Machine includes a full copy of an operating system, one or more applications, necessary binaries and libraries - taking up tens of GBs. VMs can also be slow to boot.


Virtual Machines

A VM is a software based environment geared to simulate a hardware-based environment, for the sake of the applications it will host. Standard applications are designed to be supervise by an operating system and executed by a batch of processor cores. Such applications can run within a VM without any re-architecture.
With Virtual Machines, a software unit called a hypervisor reacts as a factor between the Virtual Machine environment and the underlying hardware, providing the prerequisite layer of abstraction. A hypervisor which is responsible for executing the virtual machine assigned to it and can execute several simultaneously is a VMware ESXi, Other Known hypervisors which are as follows KVM, Citrix Xen, and Microsoft Hyper-V. In the most recent Virtual machine environments, modern processors having the ability of interacting with hypervisors directly, providing them with channels for pipelining instructions from the Virtual Machine in a manner that is completely opaque to the applications running inside the VM. They also include sophisticated network virtualization models such as VMware NSX.

The scalability of a VM server workload is achieved in much the same way it is achieved on bare metal: With a Web server or a database server, the programs responsible for delivering service are distributed among multiple hosts. Load balancers are inserted in front of those hosts to direct traffic among them equally.

## 4. CONCLUSION

Container based virtualization gives you ability to run guest machine as close as a virtual machine without the need of separate kernel and hardware simulation. In containers, you don't need hardware virtualization module i.e. intelvt or amd-v.Simply install the openvz/lxc/docker packages, reboot and create vms. There is no specific hardware required and you can even run it on a virtulabox/workstation vm.Instead of runningthe entire virtual Machine for every application A single control host is used to run the multiple isolated system and access a single kernel which provides huge benefit in terms of cost reduction and provides Lowering Power Consumption in Cloud Environment. Container helps to simplify the development and deployment by able to run the container virtually anywhere which can on linux, windows and mac OS or even on virtual machine or on public cloud. Containers provide the developers the ability to create predictable environments that are isolated from other applications.

## REFERENCES

1. Pahl, C. (2015). Containerization and the PaaS Cloud. IEEE Cloud Computing, 2(3):24–31.
2. Manu, A., Patel, J., Akhtar, S., Agrawal, V., and Murthy, K. (2016). Docker container security via heuristics-based multilateral security-conceptual and pragmatic study. In Intl Conf on Circuit, Power and Computing Techn, pages 1–14.
3. Pahl, C., Helmer, S., Miori, L., Sanin, J., and Lee, B. (2016). A container-based edge cloud paas architecture based on raspberry pi clusters. In 4th IEEE Intl Conf on Future Internet of Things and Cloud Workshops.