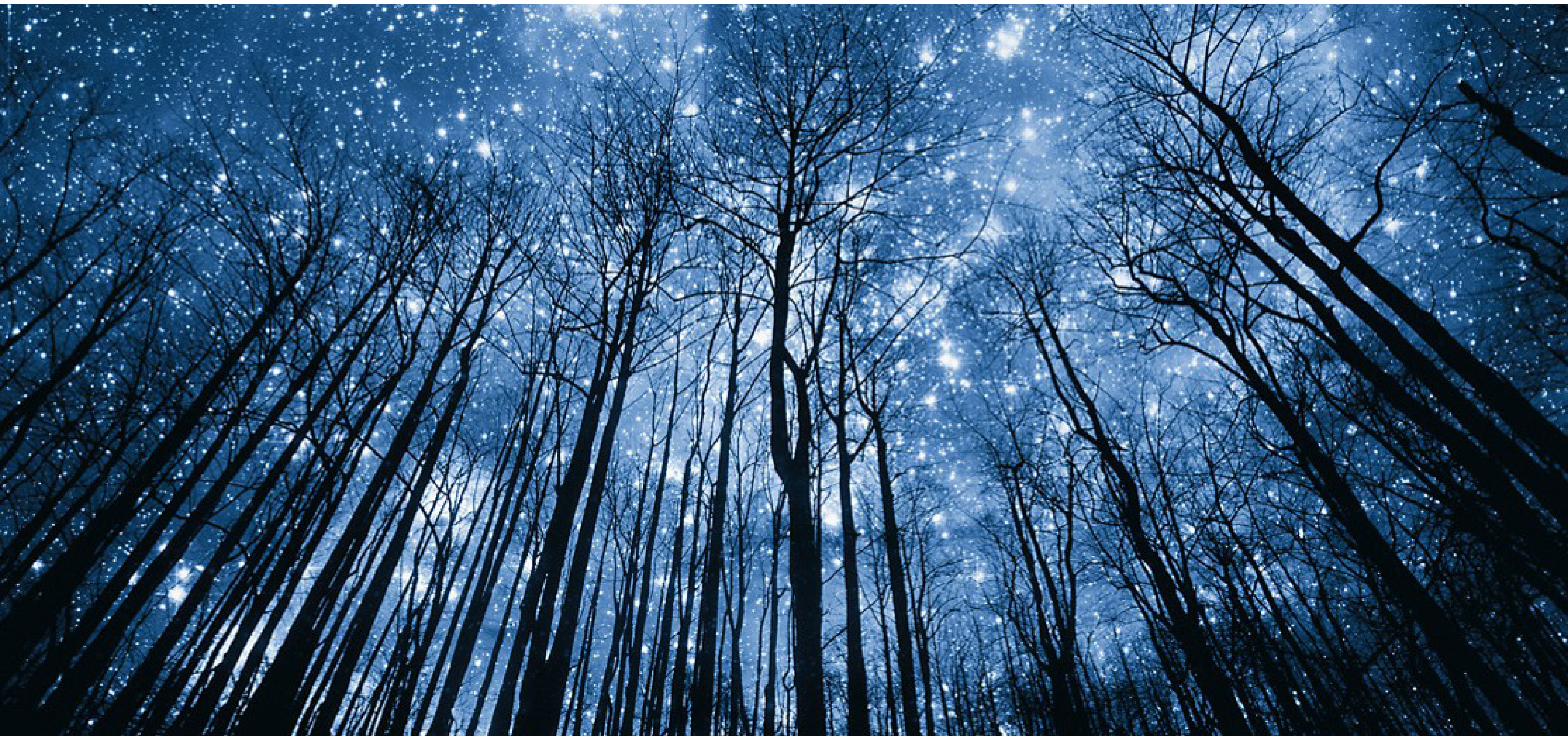
EDGE OF CONTAINERIZATION IN CLOUD COMPUTING



Ronit Mukherjee
Inside Product Specialist
Dell Technologies

Saipriya Swain

Knowledge Sharing Article

The Dell Technologies Proven Professional Certification program validates a wide range of skills and competencies across Dell's multiple technologies and products with both skill and outcome-based certifications.

Proven Professional exams cover concepts and principles which enable professionals working in or looking to begin a career in IT. With training and certifications aligned to the rapidly changing IT landscape, learners can take full advantage of the essential skills and knowledge required to drive better business performance and foster more productive teams.

Proven Professional certifications include skills and solutions such as:

- Data Protection
- Converged and Hyperconverged Infrastructure
- Cloud and Elastic Cloud
- Networking
- Security
- Servers
- Storage
- …and so much more.

Courses are offered to meet different learning styles and schedules, including self-paced On Demand, remote-based Virtual Instructor-Led and in-person Classrooms.

Whether you are an experienced IT professional or just getting started, Dell Technologies Proven Professional certifications are designed to clearly signal proficiency to colleagues and employers.

TABLE OF CONTENTS

INTRODUCTION

In a traditional environment of allocating resources to hosts in a cloud-based environment where, for instance, the IT Administrator would want to allocate resources to a certain number of hosts, virtualization is usually preferred and used. This is because virtualization allows a finite number of hardware resources to be distributed in a streamlined fashion to multiple processes that would require them, and Operating System virtualization is helpful to establish a virtual hosting environment for servers or applications. It is widely believed that virtualization is the foundation of cloud computing, as many IT organizations would want to deploy servers that are only running at a fragment of their capacity as they are dedicating their physical server to an application with a specific functionality. This is usually an inefficient mechanism as there is an excess of capacity there which is not being consumed, which in turn would lead to operational and IT costs which are on the higher end.

To address the above challenges, and to drive higher capacity utilization and reduction of costs, virtualization was created, and hence the technology of virtualization has been used so widely.

We know with most technologies, even though there may be many benefits, there could also be a certain number of shortcomings. In the field of virtualization technology, one key disadvantage that persists with the process of deploying or running virtual machines is the fact that a Guest Operating Systems, a virtual copy of the hardware that the OS requires to run, always needs to be deployed before any application could run within that Virtual Machine. Without a Guest Operating System, a Virtual Machine would be rendered unusable, so this makes Virtual Machines quite resource heavy. As in any environment, where virtualization technology needs to be utilized, a Guest Operating System is necessary, which would ask for quite a number of CPU cycles, RAM allocation, Capacity Utilization and so on. As a result, powering on Virtual Machines or Powering off Virtual Machines would be a time-consuming procedure, and working with virtual machines might turn out to be a bit sluggish.

Containerization solves this problem by not requiring the use of a Guest Operating System entirely. In other words, in the field of Containerization, a guest OS is completely omitted. The reason behind this is that, in virtualization, where we virtualize the underlying hardware, containers virtualize the operating system (typically Linux) so that each individual container contains *only* the application and its libraries and dependencies. The absence of the Guest OS is why containers are so lightweight, fast, and portable.

This article aims to explain the various attributes of virtualization and Containerization and their usages. It will discuss the very basics of these two technologies, and then discuss their differences and what advantages Containerization has over Virtualization.

Virtualization and Virtual Machines

We do understand that in the world of software development, where we are constantly developing applications and testing them in various environments, we might often need to lengthen the various offerings and functionalities of our Host Operation System, or Host OS, to add some additional attributes. We could also have scenarios where we might have projects that require maintenance and testing on a few devices that we may not have within our environment. Virtualization will help us out here. In such a situation, a virtual machine (VM)would empower us to run a guest OS on our own computers' hardware. Software developers today can easily install Cent OS on a Windows computer or run a Windows Server OS in a Mac OS environment.

VMs possess their own set of software and tools: the IT administrator acquires certain access to certain libraries, configurations files and other dependencies, which are specific in relation to this newly built virtual OS. Additionally, VMs use the absolute quantity of memory whether in GB or MB, that we allocate to it.

Now to sum up what exactly a Virtual Machine is**,** it is a machine or a Compute system that is not physical, or tangible, but virtual. However, for us to create a Virtual Machine, we need a physical compute system, called the Host, and our VM would be dependent on the underlying resources which are the CPU, Memory, Storage of that physical computer.

We will now look at the best possible features that Virtualization offers.

- **OS Isolation:** all the Operating Systems in a VM environment function in a mutually exclusive way.
- **Virtualization at a Hardware Level:** a multitude of Guest OS have the independence to split the resources of the underlying resources amidst each other.
- **Cost Savings:** Development costs would be reduced due to quick provisioning of servers, improvement in disaster recovery and reduction of footprints.

Let us now review what Containerization and Containers are in detail.

Containerization and Containers

Containerization has turned into a buzz word that we have been hearing a lot. According to business analysts, the software containerization market has been scoped out to grow at a rate of 30% per year. This is because whenever we want agility in our environments, containers are the way to go. To understand such statistics better, we need to first understand what containers are.

A Container is basically an application packaged together with all the necessary dependencies in an executable file. Here, the term Containerization is quite literally acquired from the word Container. Because, to put it simply, a container is an everyday object in our lives where we store food, accessories, tools, or anything of our choice that we can enclose. In software development, or cloud computing, a container is used to hold and transport applications from one environment to another.

Hence, Containerization pertains to the procedure of bundling all the variables that are required to run an application into a singular self-supporting file that can be operated in any environment.

Because each container only has access to the resources assigned to it individually, containerization offers a higher level of isolation. Malware and other harmful programs have a harder time spreading between containers as a result. Since each containerized application does not require installing or configuring a separate virtual machine, containerized applications can also be updated and deployed quickly.

Because numerous containerized apps can run on the same physical server or virtual machine, containerization can also help cut costs. This uses resources effectively and can lower the overall cost of ownership. So, it makes sense that containers have evolved into the fundamental building block of modern cloud-native apps.

Whether using Windows or Linux, we may immediately install containers on our operating system. The size of the containers is one of its most important characteristics and significant benefits. Installing a small container rather than a large VM that takes up a lot of storage on the server (in GB) is considerably simpler and quicker. Container mobility is another crucial factor.

Containers run on a host OS, and this fact alone makes managing them much easier. There is no need to monitor both the VM and the Host OS; only one OS needs to be kept under observation.

We will now take a look at the major Containerization types and their usages in relation to Cloud Computing.

Containerization Types and Usages for Cloud Computing

Consolidation of Servers

As a part of server consolidation, multiple server instances are run on a singular server, and this is a sort of containerization. In addition to lowering server expenses and increasing server usage, this can also make server management easier. Blade servers and server clustering are two of the various approaches to server consolidation. Containers can leverage shared memory and other computational resources across instances because they don't have numerous operating system memory footprints, which opens up the possibility of dense consolidation.

Containers as a Service (CaaS) and Multi-Tenancy

With the use of container-based virtualization, or CaaS, service providers may offer cloud services that include container engines, orchestration, and other computing resources. The development process is made more agile and straightforward with this methodology.

A single software instance can serve many end users inside a single tenancy thanks to multi-tenancy. Because of this, the supplier can automate the construction of any containerized program, speeding up deployment and providing simple maintenance.

For instance, a popular picture sharing platform transitioned to containerized technology for its platform to handle the increasing workload and fix operational problems

Hybrid and multi-cloud approaches

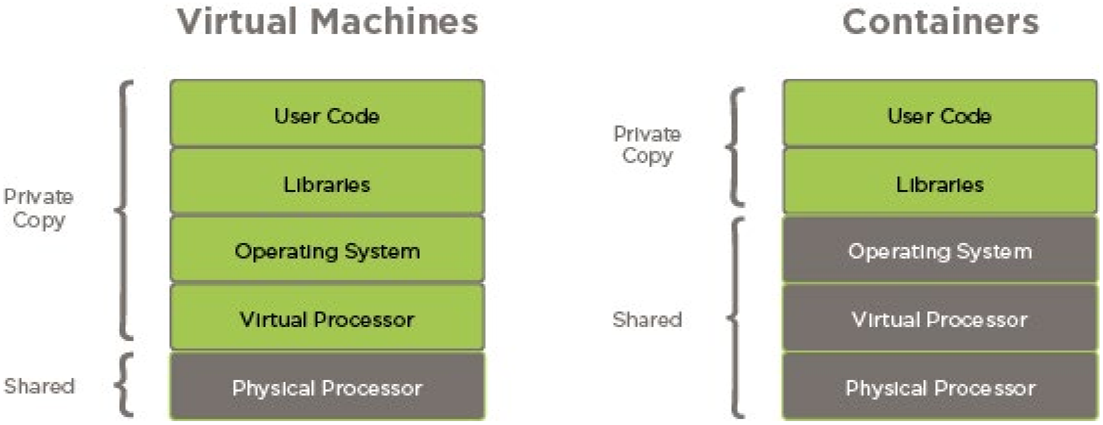
Organizations of all sizes, including startups and multinational corporations, are aware of the cost reductions, operational effectiveness, and agility offered by a hybrid cloud strategy. To address everything from app development and hosting to disaster recovery, most enterprises are exploring a hybrid and multi-cloud approach that uses containers.

Users utilize this pairing to benefit from the flexibility of various cloud infrastructures and programming languages based on multi-cloud management best practice. The flexibility of workloads that are hosted in infrastructures that are managed, whether on-prem or the public cloud, would always be a benefit.

Major Features of Containerization

- Virtualization of Host OS: On a Single OS, multiple containers would run
- Rapid Provisioning of Data: Due to their small size, containers are provisioned almost instantly
- Vernacular Performance: Because complete access to the underlying compute systems is provisioned to the container, exceptional performance is observed.

We will now look at the key differences between Virtual Machines and Containers.



As we can see in the picture above, instead of virtualizing the underlying hardware, containers virtualize the operating system (typically Linux) so each individual container contains only the application and its libraries and dependencies. The absence of the guest OS is why containers are so lightweight and thus fast and portable.

Major Differences between Virtualization and Containerization

Now that we know what these two technologies are, we can move on and look at the key differences between Virtualization and Containerization.

Host OS Separation

The way that virtualization and containerization isolate the host server OS from the applications that execute on it is the main distinction between them. Containerization isolates the applications from the OS whereas virtualization detaches and divides the OS into numerous "operating systems" using a hypervisor. This makes sure a container won't wreak havoc on the host operating system. Since each container is independent, installing and removing its own software will not have an impact on the other containers or the primary OS.

Operating System

Multiple operating systems can be run on the same hardware using virtualization, but something to note here is that the guest OS (the operating system of the virtual machine) is locked out of the underlying OS. Contrarily, containerization isolates each program within a separate operating environment.

Time Required in Booting Up

Beyond a shadow of a doubt, containers do start up and run quickly. On the other hand, a virtual computer must first boot the full virtual system. Each VM has a full copy of the operating system, therefore starting one up uses a lot of resources. Containers are far lighter than virtual machines and do not require the replication of OS code.

Running Environment

Virtual machines are generated on hardware, whereas containers operate on software. Virtual machines essentially run on a single physical machine. While each virtual machine (VM) demands its own operating system and a whole boot procedure, a container just needs a single operating system to function.

We will now try to understand the advantages of Containerization over Virtualization.

Advantages of Containerization over Virtualization

Portability

The fact that containerization enables organizations to keep their software programs portable by enabling developers to utilize them in various settings may be the most important benefit of containerization over virtualization. Transferring containers from one machine to another is significantly simpler. Developers may move them from one platform to another without worrying about compatibility problems. Virtual Machines, on the other hand, are bigger and call for a copy of the host operating system.

Deployment

Developers need not worry about where an application will be deployed using containers because it may operate on a private cloud today and a public cloud tomorrow. They are perfect for automated applications and DevOps workflows. Additionally, they provide continuous integration and deployment.

Less Resource consumption

Containers need far fewer resources than virtual machines since they use a common operating system kernel. Containers are substantially smaller than virtual computers. They also consume less resources since they only employ a small fraction of the operating system kernel on the host machine. In comparison to virtual computers, containers start up significantly faster.

Scalability

Virtualization, which restricts the number of instances per server or cloud instance, grows less well than containers. Additionally, any potential faults or failures cannot harm other containers because the program is isolated. Developers may specify security rights and automatically prohibit undesirable components using containerization, ensuring that programs are functioning in the appropriate environment.

Reduced Runtime

Runtime is cut by containers. They begin immediately as they are operating system-based. As virtual machines take a while to boot up, this is fantastic news for teams working on software. As a result, virtual machines frequently take a long time to start, and containerization shortens this time.

Memory Efficiency

Virtual machines cannot compare to the memory efficiency of containerized applications. This is so that the operating system does not have to be running in its entirety within the container. Additionally, systems with containerized workloads may determine when they need to scale out by adding and deleting containers according to how much memory they need. They have the option to run several apps on a single server thanks to this capability.

Shareability

Unlike virtual machines, container images can be easily shared between multiple servers. Virtual machines cannot leverage application sharing hubs or other sharing services.

Management of Containers

Container management is another virtue which gives it the edge over VM management as even though we may have hundreds or thousands of containers in an environment, by using orchestration software such as Dockers or Kubernetes, achieving this feat is again simplified and fast, and it is a need that is much called for in the cloud computing space.

Now that we have a fair understanding of the advantages of Containerization over Virtualization, we will consider the conclusion and understand why Containerization is better than Virtualization.

Conclusion

Is Containerization better than Virtualization?

This is a common and fair question that we had asked at the very beginning of this article. When we are speaking in context to a cloud environment, then a firm choice between the two would eventually depend on the end goal in mind as both Virtualization and Containerization are extremely powerful and prominent technologies, so evaluation of the specific project is required.

Although one objective inference we can surely make from our understanding of this article is that Virtualization does indeed stand out to be a common choice for applications which are demanding in nature and more often than not would demand the entire serviceability of the underlying OS and the hardware.

But, if we are looking for a solution that entirely eradicates the tribulation of servicing Virtual Machines that operate on the Host OS, while reducing inconsistencies as they are isolated from the Host OS, and are substantially lighter in weight, then Containerization would always have the upper edge over Virtualization. After all, in cloud computing, we are always looking forward to a smooth, fast, and efficient experience, and containerization does provide us with this benefit, helping not only the Information Technology engineering or administration team, but also our customers. This is an important consideration, as the actual decision is always contingent on our customers and which technology they would toil with.

References

- Containerization vs. Virtualization: Which Is Better? (sirinsoftware.com)
- Containerization vs Virtualization: What is the Difference? — ITcompanies.net
- https://www.youtube.com/watch?v=Gn7IoT_WSRA
- The Containerization Market in 2022 | Enterprise Storage Forum
- Containerization in the Cloud – Definition, Uses, and Value | Talend
- Containerization in the Cloud & How do you Manage Containers? (cloudlytics.com)

Dell Technologies believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

Disclaimer: The views, processes or methodologies published in this article are those of the authors. They do not necessarily reflect Dell Technologies' views, processes, or methodologies.