

On the surface, containers and virtualization appear to accomplish similar things, but they actually operate on very different levels and can be used synergistically for maximum benefit.

The Synergies Between Containers and Virtual Machines

November 2020

Written by: Gary Chen, Research Director, Software Defined Compute

Introduction

The adoption of containers and the Kubernetes orchestration platform is accelerating in the enterprise as businesses grapple with agility and the ability to support their digital transformation efforts. Containers are gaining traction as an efficient and developer-friendly way to ship applications faster and for their agile and highly automated management. Enterprises are using containers to encapsulate microservices-based applications and for modern workloads such as artificial intelligence and machine learning (AI/ML) and edge.

However, in the enterprise, containers aren't used just for new applications. IDC data shows that existing applications make up 55% of containerized footprints today. Most enterprises are modernizing legacy apps, in addition to net-new cloud-native development, to close gaps in their digital transformation strategy. The refactoring of existing applications is a key part of that strategy.

Over the past two decades, server virtualization has become the default infrastructure in datacenters and the cloud. Containers may seem to overlap with virtualization because each provides a modern, agile vessel in which to wrap software, but they are actually quite different and work at different levels of the stack. Hypervisors are a hardware construct that emulates server hardware in software to partition the resources of a server. Containers, on the other hand, are a packaging technology for applications and a way to sandbox different applications on the same operating system (OS) host during runtime.

A quintessential example of the use of virtualization and containers together is the public cloud, where nearly all containers run inside virtual machines (VMs). The primary reason for this is the need for secure multitenancy because the container boundary is not strong enough to isolate tenants and the boundaries between tenants must be enforced with VMs. Additionally, VMs partition and provision the underlying hardware for heterogeneous customer environments, while containers are then used to manage each customer's apps within that VM slice. Consolidating heterogeneous

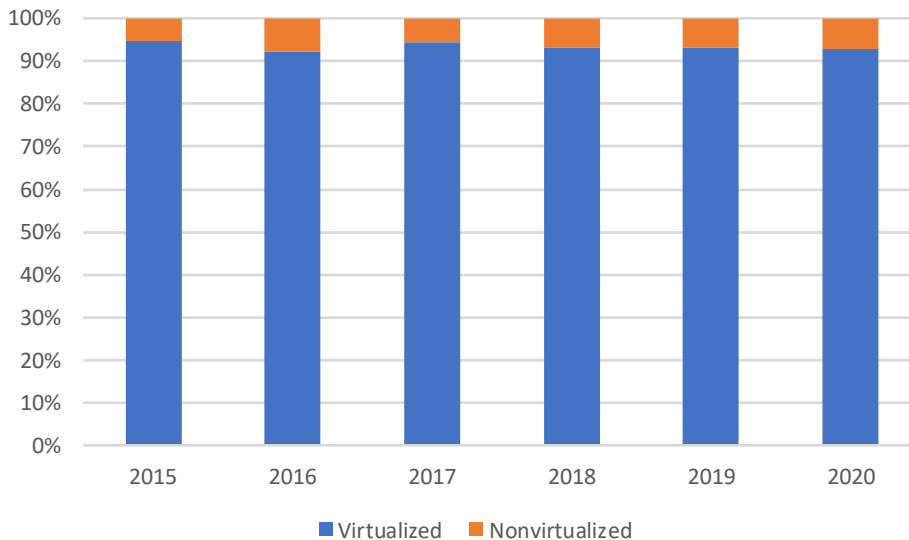
AT A GLANCE

KEY TAKEAWAYS

- » Containers operate at the application level, while virtualization operates at the hardware level, which means the two can work together rather than one being a replacement for the other.
- » Virtualization can increase security, manageability, utilization, reliability, availability, and scalability for containers.

workloads on the same infrastructure also maximizes utilization. Virtualization is the default in the enterprise datacenter today, which means the majority of on-premises containers deployed are in VMs (see Figure 1).

FIGURE 1: *Enterprise Container Instances Installed Base by Virtualized/Nonvirtualized, 2015–2020*



Source: IDC, 2020

Benefits

Virtualization and containers fundamentally solve problems at different layers of the stack and thus work extremely well together. The virtualization capabilities built over the past 20 years to create an agile, resilient, and more manageable infrastructure will still largely apply to a Kubernetes world. Containers can still benefit from virtualization for several reasons, including:

- » **Manageability.** Containers and Kubernetes were developed to be better application deployment and management solutions and do not address management of the underlying infrastructure. Kubernetes does not address the underlying virtual or physical infrastructure but expects the user to present a robust infrastructure on which it can operate. Faster and more agile provisioning is a strength of virtualization, as well as one of the primary reasons why customers moved from physical to virtual, and this benefit still applies to containers. Containers and VMs will coexist in enterprises for the foreseeable future, often within the same logical application. Integrated and converged management of both VMs and containers can prevent the creation of more silos and unify skills and tooling.
- » **Security.** As previously mentioned, public clouds use VMs to securely enforce multitenancy rather than relying on containers by themselves. While enterprises may not have the same type of multitenancy as the public cloud, enterprises still have many isolation requirements between different groups, users, and applications, often dictated by compliance requirements.

- » **Protection.** Used together, containers can actually enhance isolation by adding another layer of protection. An attacker would have to first break out of the container and then would also have to break out of a VM in order to compromise the host. Layered security is a best practice that can provide extra protection in case of failure, software flaws, or misconfiguration.
- » **Performance and utilization efficiency.** While containers on bare metal may seem attractive to customers from a performance standpoint, there are trade-offs. For most customers, only a small percentage of applications would benefit from the removal of any hypervisor overhead. The primary factor for the majority of workloads is manageability, not performance, and virtualization more than delivers on manageability benefits at the expense of a small level of performance. Enterprises should keep in mind that virtualization overhead continues to drive toward near zero. CPUs have added virtualization acceleration in silicon, removing nearly all of the CPU and memory overhead for virtualization. The rise of hardware accelerators in servers is also bringing native I/O performance to hypervisors.

Virtualization greatly increased server utilization rates, and containers can raise those rates further still. Most enterprises will run a mix of large-scale and small-scale apps and will still need to carve up a physical server into smaller pieces. Additionally, enterprises run a mix of various operating systems and multiple versions/patch levels. Containers all share the same host OS, so all containers on a bare metal server must be for the same OS version. With a large number of OS flavors and version levels, this can be difficult to organize and consolidate efficiently. Being able to mix and match different OSs with virtualization provides more flexibility.

In addition, all Kubernetes implementations have a pod limit, which could be anywhere from 100 to 500 pods per node. Given the powerful servers in use today with increasing core counts, even a 500-pod limit could result in an underutilized server if implemented on bare metal. Virtualization's ability to dynamically allocate and balance resources can be applied to Kubernetes clusters to resolve this challenge and better utilize available capacity.

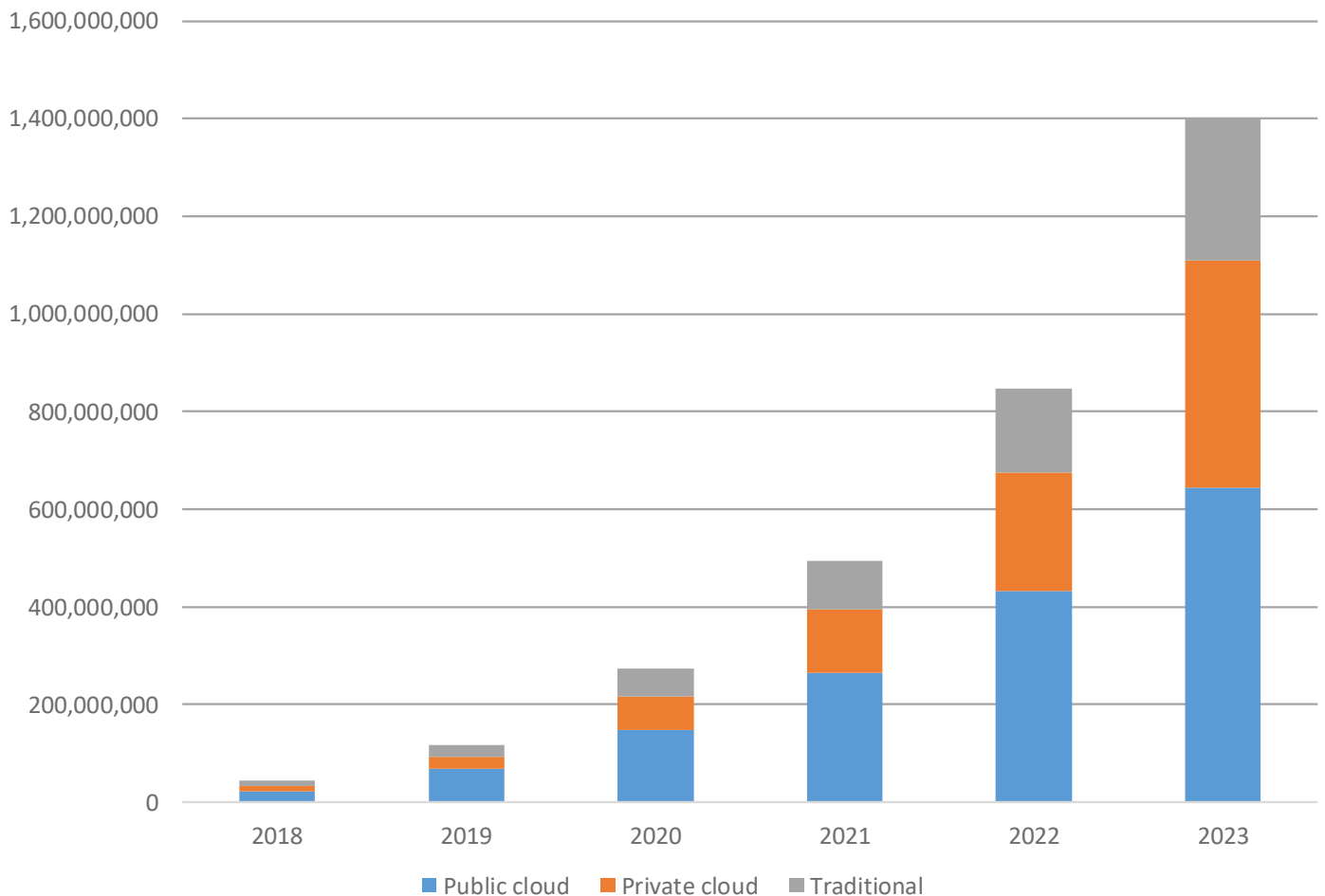
- » **Reliability, availability, scalability (RAS).** Kubernetes provides many new and enhanced application RAS capabilities, but it enhances rather than replaces hypervisor RAS features, which operate at an infrastructure level. Kubernetes focuses on container orchestration and thus provides RAS from an application point of view, such as making sure the application is always running and scaling the number of instances. Virtualization provides infrastructure-based RAS with features such as server-based high availability, nondisruptive maintenance, and live migration. Certain containerized workloads and the Kubernetes control plane can still benefit from resilient infrastructure underneath as server failures can still cause chaos for Kubernetes. Additionally, while Kubernetes can increase pod counts easily, virtualization would still be needed to increase the size of nodes or provision new nodes.
- » **Storage and networking subsystems.** Storage and networking are key elements to the scalability and performance of workloads. Storage in particular is a notable issue for containers and Kubernetes, which were originally designed for stateless apps. However, many key stateful apps are still needed, and many customers want to containerize them, driving the need for data persistence, for which Kubernetes has recently been improving support. Hypervisors already have mature storage and software-defined networking (SDN) subsystems and interfaces, and by running containers on a hypervisor, users can ease the integration of these elements in their container systems. With VMs and containers coexisting for the foreseeable future, having a shared storage and networking plane can help VMs and containers connect to each other and share data.

Trends

The IT industry is shifting to containers and Kubernetes as a new construct for shipping and managing applications, but this is likely to be a long transition. Customers need to prepare for the long-term coexistence of VM- and container-based apps and formulate strategies on that convergence and how to manage it. Some of the challenges are as follows:

- » The need to connect VMs and containers as applications, which can cross both domains
- » Management of containers and VMs in a hybrid/multicloud world, where they will be running on premises and in multiple public clouds (IDC's forecast for container deployment models shows a hybrid environment nearly evenly split between on-premises datacenters and the public cloud [see Figure 2].)

FIGURE 2: **Worldwide Enterprise Container Instances Installed Base by Deployment Model, 2018–2023**



Source: IDC's Container Infrastructure Software Market Assessment: x86 Containers Forecast, 2018–2023

Ideally, many of these challenges can be addressed by sharing as much as possible to avoid creating silos between VMs and containers. This approach can include sharing common infrastructure across core compute, storage, and networking as well as a shared common control plane that can manage VMs and containers across multiple environments. Convergence can also help with reducing skills complexity, tool consolidation, and tool reuse.

Containers are changing not only how enterprises do their own software development but also how software vendors are delivering software. According to IDC's *Container Infrastructure Software Survey*, 48% of enterprise containers are used to run packaged software. Independent software vendors (ISVs) are gravitating to distributing software as containers because the environment is highly standardized with Kubernetes, and they can deliver a better experience by including deployment and operational information with the software. Additionally, this software can optionally be fully managed by the ISVs as a remote cloud service, delivering a SaaS-type offering in any location the customer wishes.

Considering VMware

VMware is widely known as a leader in server virtualization, with the vSphere hypervisor powering the majority of enterprise datacenters today. However, the company is a major vendor for container infrastructure software with its Tanzu line of products. VMware is also a top contributor to the Kubernetes open source project and drives many container-related open source projects, such as Harbor, Velero, Contour, Antrea, and Spring. The core of Tanzu is Tanzu Kubernetes Grid, a Kubernetes distribution that works in multiple on-premises and cloud environments.

Tanzu Kubernetes Grid has particularly deep integration with vSphere. It is not simply Kubernetes bundled with or running on top of vSphere but a complete redesign of vSphere with Kubernetes as the embedded control plane, while exposing both Kubernetes and traditional vSphere APIs. This integration provides developers and cloud-native teams with a modern Kubernetes interface and API while enabling infrastructure administrators to extend support for containers via the familiar vCenter management console. This approach provides a converged path to modernization by leveraging existing skills and tools and managing both VMs and containers on a single platform. Additionally, the hypervisor and OS scheduler can be very differently optimized, and VMware has identified certain cases where leveraging both schedulers in tandem can result in faster performance compared with bare metal.

Tanzu Kubernetes Grid is available both in standalone editions that can work in non-vSphere environments such as the public cloud and as part of vSphere or VMware Cloud Foundation. By integrating Kubernetes into vSphere, VMware has extended Kubernetes to manage VMs, allowing for a modern and powerful infrastructure API to encompass all VMware workloads. vSphere with Tanzu includes this integrated Kubernetes, and customers have the choice to plug in their own storage and networking systems. VMware Cloud Foundation provides a more full-stack experience, with integrated vSAN storage and NSX-T networking, along with a highly automated deployment and life-cycle manager that can ease large-scale deployments. vSAN has been enhanced to provide a data persistence platform for containers, and NSX-T networking is key to bridging VMs and containers.

Tanzu Kubernetes Grid is the core platform for the larger Tanzu portfolio. Together, they create a modern platform for containers. Various Tanzu editions (Basic, Standard, Advanced, and Enterprise) bundle the various Tanzu products for easy consumption at different maturity stages. Key Tanzu products include the following:

- » **Tanzu Mission Control** is a multicluster Kubernetes management solution that can manage across on-premises and cloud, including management of any Kubernetes conformant cluster. It provides global visibility and policy management.

- » **Tanzu Observability** provides modern observability from applications to infrastructure with metrics, traces, span logs, and analytics.
- » **Tanzu Service Mesh** provides microservices networking to connect, monitor, and secure cloud-native applications.
- » **Tanzu Application Catalog** is a customer-curated and VMware-maintained library of open source software such as application components, databases, and runtimes that are continuously tested and maintained.
- » **Tanzu Application Service** is a Cloud Foundry-based PaaS that provides a modern runtime for Java, .NET, and Node applications, providing developers with application velocity and operations with platform automation efficiencies.

Challenges

VMware's greatest challenge in integrating containers and VMs will be one of perception and communicating deeply technical and complex issues. As discussed in this paper, there are many technical reasons to run containers on a hypervisor, such as for security, availability, and manageability. All of the public clouds continue to run containers on a hypervisor for many of these reasons. However, customers continue to be attracted to the perception of removing cost and simplifying the stack by removing the hypervisor, and this mindset is further encouraged by most of VMware's competition as a strategic move. VMware's challenge will be to simplify a complex topic and effectively message to the market the differences between containers and VMs and why a hypervisor is still needed. Although containers do not invalidate the need for hypervisors, VMware and the virtualization market will have to adapt to a changing reality, and Tanzu is a major investment into containers and integration into the VMware platform.

Conclusion

The topic of virtualized versus bare metal containers is extremely complex, with numerous technical nuances. While containers improve many things at the application level, the lower-level infrastructure still needs to be managed by something. Removing the hypervisor means reverting to previrtualization bare metal management and all the complexities and headaches that virtualization solved. A smaller stack does not always mean simplification. Today's modern stack is much larger than the stack of the past, but those additions have added valuable functionality, abstractions, and automation. The hardware abstraction, partitioning, and management provided by hypervisors still deliver value underneath containers, and this manageability benefit is something that must be balanced against any perceived cost and simplification benefits, especially when operating at scale.

About the Analyst



Gary Chen, Research Director, Software Defined Compute

Gary Chen is IDC's Research Director, Software Defined Compute. His research focuses on server virtualization, container infrastructure and management, and cloud system software (system software used to build IaaS clouds such as OpenStack).

MESSAGE FROM THE SPONSOR

VMware has put considerable emphasis on building out its Kubernetes management capabilities via VMware Tanzu, accelerating app and infrastructure modernization and providing a simple and fast way to get started with Kubernetes and modernize workloads traditionally run in data centers. VMware recently announced VMware Tanzu editions – Tanzu Basic, Tanzu Standard, Tanzu Advanced, and Tanzu Enterprise – packaging capabilities of the Tanzu portfolio into solutions that directly address common customer challenges. All four Tanzu editions make the best possible use of various open source projects while putting Kubernetes at their core. With a ubiquitous Kubernetes distribution across the portfolio, customers can run any app across multiple clouds – including vSphere, public cloud, and edge. The Tanzu editions are packaged to bring Dev and Ops together in a shared effort to continuously deliver a better path from software to production. To learn more about Tanzu editions and the value they bring to your organization, please visit <https://tanzu.vmware.com/tanzu>.



The content in this paper was adapted from existing IDC research published on www.idc.com.

IDC Research, Inc.

5 Speen Street
Framingham, MA 01701, USA

T 508.872.8200

F 508.935.4015

Twitter @IDC

idc-insights-community.com

www.idc.com

This publication was produced by IDC Custom Solutions. The opinion, analysis, and research results presented herein are drawn from more detailed research and analysis independently conducted and published by IDC, unless specific vendor sponsorship is noted. IDC Custom Solutions makes IDC content available in a wide range of formats for distribution by various companies. A license to distribute IDC content does not imply endorsement of or opinion about the licensee.

External Publication of IDC Information and Data — Any IDC information that is to be used in advertising, press releases, or promotional materials requires prior written approval from the appropriate IDC Vice President or Country Manager. A draft of the proposed document should accompany any such request. IDC reserves the right to deny approval of external usage for any reason.

Copyright 2020 IDC. Reproduction without written permission is completely forbidden.