Is the hype around containers justified, or are they simply an alternative form of virtual machine? 451 Research believes containers are better placed, at least theoretically, to achieve lower TCO than traditional hardware virtualization. In fact, we have found that double-digit resource savings are achievable even with relatively simple implementations.

By reducing duplication, server resources are freed to be allocated to other requirements. In other words, container technology is likely to more efficiently 'sweated' – resources being shared, with the asset used to the fullest – than hardware virtualized counterparts. The asset-sweating stretches beyond just servers – bandwidth, time, bits, bytes and labor are all likely to be better utilized with containers, according to our research.

THE 451 TAKE

Rarely are decisions in IT based purely on cost. Cost is of course a factor, but this is balanced against the value achieved for that cost. Virtual machines are unlikely to be as cost-efficient as containers, but they do provide value in other ways (which we'll cover in a follow-up report). However, the economic advantage of containers suggests they're not slowing down anytime soon – by their very nature, they have an economic edge over hardware virtualization, and this is likely to be taken advantage of by vendors, providers and end users. Over time, software vendors will seek to improve containers, so their value proposition will only increase against virtual machines.

VIRTUALIZATION ECONOMICS

The primary economic benefit of traditional server virtualization is the 'sweating of assets' through the consolidation of hardware – it roughly means 'getting as much use as possible out of what you already possess.' Originally, one server meant one operating system, typically delivering one workload. Through virtualization, one server can hold multiple operating systems, each one operating a logically separated workload. Before virtualization, perhaps just a tiny fraction of the asset (the server and its resources) might be used at any one time. Through virtualization, we can multiplex multiple applications together, so that resources are shared and the asset is fully used.

If a server at a total cost S was previously able to hold just one workload, but can now support n workloads, the cost per workload plummets from S to S/n. If n is 16, a fairly reasonable level of consolidation, that's a 93% average cost savings per workload. The greater the value of n, the greater the savings. It is clear why virtualization is so commonplace today.

The cloud was the next step up from virtualization, providing the benefits of consolidation with the flexibility of being able to dynamically create and move resources to suit different business requirements. None of this is rocket science, and most in IT have theoretical and practical experience in the subject. But containers have seriously rocked the boat. They're the new kids on the block, but are they all they're cracked up to be?

A CONSOLIDATED NEW WORLD

Hardware virtualization means that operating systems (and their applications) share hardware resources such as compute, storage and memory from a single asset, be it a server or even a pool of servers. Container technology, specifically system containers, is essentially operating system virtualization – workloads share operating system resources such as libraries and code.

Containers have the same consolidation benefits as any virtualization technology, but with one major benefit – there is less need to reproduce operating system code. Hardware virtualization means each workload must have all its underlying operating system technology. If the operating system takes up 10% of a workload's footprint, then in a hardware virtualized platform, 10% of the whole asset is spent on operating system code. This is regardless of the number of workloads, n, being run on the asset.

In the same environment utilizing containers, the operating system only takes up 10% divided by the number of workloads, n. In a nutshell, our server is running 10 workloads, but only one operating system, in our container environment – in the virtualized environment, the server would be running 10 workloads and 10 operating systems.

Let's say we want to locate 10 applications on a hardware server costing $1,000. If we use virtualization, where each operating system takes up 10% of the VM, each server costs $100. If we use containers, each server also costs $100, but it requires less space to host. So containers effectively give us the same capability with fewer resources.

An estimate of the effective resource savings of choosing containers over virtual machines can be calculated as follows, based on a homogeneous environment of workloads: (n-1)/n*(percentage footprint size of OS).

This is shown in the table below for a number of options.

		FOOTPRINT OF O/S PER WORKLOAD				
		10%	20%	30%	40%	50%
WORKLOAD PER SERVER	1	0%	0%	0%	0%	0%
	2	5%	10%	15%	20%	25%
	4	8%	15%	23%	30%	38%
	8	9%	18%	26%	35%	44%
	16	9%	19%	28%	38%	47%
	32	10%	19%	29%	39%	48%
	64	10%	20%	30%	39%	49%
	128	10%	20%	30%	40%	50%

As an example, let's say in our environment, we need eight workloads supported on a single asset, with 10% of each application holder dedicated to operating system. By choosing containers over virtualization, we are poised to achieve resource savings on the order of (7/8*10%) = 9% of the server.

The more workloads per server, the greater the savings, although this increase is relatively small; what really matters is operating system footprint. The larger the footprint, the greater the savings made by choosing containers, and this can be significant. Just using two containers, instead of two virtual machines where the footprint is 50%, can create savings of 25% on resources. The point is that operating system virtualization does a better job of consolidation than hardware virtualization, simply because there is less duplication and therefore less resource consumption.

The above is a very simple analysis, based on the crude assumption that the resource requirements for containers and/or virtual machines are fixed, and don't grow or shrink depending on the demand placed on them. However, both containers and virtual machines possess this ability, so it's unlikely to significantly impact our general findings. We've also made the assumption that no operating system elements are duplicated in a container environment, although some duplication is likely. Conversely, most hardware virtualization platforms do attempt to reduce duplication of virtual machine code.

However, the point is that containers, by their very nature, are likely to be better at asset-sweating than hardware virtualization simply because they reduce duplication, thereby providing more capability from the same asset. The reduction in duplication doesn't just free up resources on the server, it frees up other resources too. This is among the primary drivers of containers in enterprise IT – to reduce overhead and boost efficiency.

For a start, spinning up apps is likely to be quicker because there is less need to instantiate a new operating system for each workload. And when transferring or migrating workloads, less needs to be sent and received – this applies to updating workloads as well. Another ancillary benefit would be less cumbersome, and fewer, OS security updates. The size of snapshots is reduced, as is labor; with less duplication, there is less to be managed.

So economically (at least in theory), containers make a lot of efficiency and economic sense. Could this represent the dawn of a new age? Is this the death of the virtual machine? So why isn't everyone using them? As we will cover in more depth in a subsequent report on the pros and cons of containers, and the comparison to VMs, there are still cases where VMs are the right match for applications, including those that require OS components and functionality.

There are also benefits to VMs in more mature security, tooling, management and process, which is why we expect – just as there are still physical machines alongside today's VMs – there will likely be VMs alongside containers. Nevertheless, efficiency, scalability and performance factors (including the cost advantage illustrated in this report) will undoubtedly mean more disruption to VMs from the advent of containers.