**Virtualization** enables you to run multiple operating systems on the hardware of a single physical server, while **containerization** enables you to deploy multiple applications using the same operating system on a single virtual machine or server.

The container shares the kernel of the host OS with other containers, and the shared part of the OS is read-only. Therefore, the containers are lightweight, so you can deploy multiple containers on a single server (or a VM)—no more dedicating an entire server to a single application. And, you only have one OS to maintain. Scaling up becomes fast and easy, without the need for more server space.

Comparing Containers and Virtual Machines

Containers and virtual machines have similar resource isolation and allocation benefits, but function differently because containers virtualize the operating system instead of hardware. Containers are more portable and efficient.

#### **CONTAINERS**

Containers are an abstraction at the app layer that packages code and dependencies together. Multiple containers can run on the same machine and share the OS kernel with other containers, each running as isolated processes in user space. Containers take up less space than VMs (container images are typically tens of MBs in size), can handle more applications and require fewer VMs and Operating systems.

#### **VIRTUAL MACHINES**

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