

[Previous](#)

Unit 5 of 14

[Next](#)

✓ 100 XP



Describe Azure Containers

6 minutes

While virtual machines are an excellent way to reduce costs versus the investments that are necessary for physical hardware, they're still limited to a single operating system per virtual machine. If you want to run multiple instances of an application on a single host machine, containers are an excellent choice.

What are containers?

Containers are a virtualization environment. Much like running multiple virtual machines on a single physical host, you can run multiple containers on a single physical or virtual host. Unlike virtual machines, you don't manage the operating system for a container. Virtual machines appear to be an instance of an operating system that you can connect to and manage. Containers are lightweight and designed to be created, scaled out, and stopped dynamically. It's possible to create and deploy virtual machines as application demand increases, but containers are a lighter weight, more agile method. Containers are designed to allow you to respond to changes on demand. With containers, you can quickly restart if there's a crash or hardware interruption. One of the most popular container engines is Docker, which is supported by Azure.

Compare virtual machines to containers

The following video highlights several of the important differences between virtual machines and containers:

Azure Container Instances

Azure Container Instances offer the fastest and simplest way to run a container in Azure; without having to manage any virtual machines or adopt any additional services. Azure Container Instances are a platform as a service (PaaS) offering. Azure Container Instances allow you to upload your containers and then the service will run the containers for you.

Use containers in your solutions

Containers are often used to create solutions by using a microservice architecture. This architecture is where you break solutions into smaller, independent pieces. For example, you might split a website into a container hosting your front end, another hosting your back end, and a third for storage. This split allows you to separate portions of your app into logical sections that can be maintained, scaled, or updated independently.

Imagine your website back-end has reached capacity but the front end and storage aren't being stressed. With containers, you could scale the back end separately to improve performance. If something necessitated such a change, you could also choose to change the storage service or modify the front end without impacting any of the other components.

Next unit: Describe Azure Functions

[Continue >](#)