



Introduction

2 minutes

The use of containers to develop and deploy software has become popular over the last few years. Containers make it easy to package and deploy an application with all its services to any computing environment. When your application meets higher demand, you can easily scale out your services by deploying additional container instances. Containers are also less resource-intensive than virtual machines. This efficiency helps you make better use of computing resources and save money.

The standard container management runtime focuses on managing individual containers. But there are times when you want to scale and have multiple containers working together. Managing multiple containers is challenging because several factors need consideration. Suppose you need to handle load balancing, security, network connectivity, and deployment. It's common to use a container management platform such as Kubernetes to make the management process easier.

Suppose you work at a drone management company. Your company provides a drone tracking solution to customers worldwide. The solution is built and deployed as microservices and includes several major applications:

- A web front end that includes maps and information about tracked drones.
- A cache service that stores frequently requested information displayed on the website.
- A RESTful API where tracked drones send data about their status, such as a GPS location and battery charge levels.
- A queue that holds unprocessed data collected by the RESTful API.
- A data processing service that fetches and processes data from the queue.
- A NoSQL database that stores processed tracking data and user information captured from the website and the data processing service.

You're using containerized instances to quickly deploy into new customer regions and scale resources as needed to meet customer demands. You want to use a container orchestration platform that simplifies the development, deployment, and management of these containerized applications.

Here, you'll see how you can use Kubernetes to orchestrate containerized applications. The goal is to help you decide if Kubernetes is a good choice as an orchestration platform for your business.

Learning objectives

In this module, you will:

- Evaluate whether Kubernetes is an appropriate orchestration platform for your workload
- Describe the difference between master nodes and nodes in a Kubernetes cluster
- Describe how the components of a Kubernetes cluster work and support compute container orchestration

Prerequisites

- Basic understanding of microservices
- Basic understanding of how Docker containers work
- Basic knowledge of how to install software by using a command-line interface

Next unit: What is Kubernetes?

[Continue >](#)