

Los diferentes orquestradores

DigitalHouse >
Coding School



**Certified
Developer**
The Ultimate Tech Degree



Los **orquestadores** son sistemas de automatización del despliegue, ajuste de escala y manejo de aplicaciones en *contenedores*.



Características

Un *orquestador de contenedores* se ocupa de cuestiones como:

- Configuración automática.
- Despliegue y "levantado" automático de servicios basados en contenedores.
- Balanceado de carga.
- Autoescalado y autoreinicio de contenedores.
- Control de la "salud" de cada contenedor.
- Intercambio de datos y networking.
- Mantenimiento de parámetros "secretos" y configuraciones.



Kubernetes

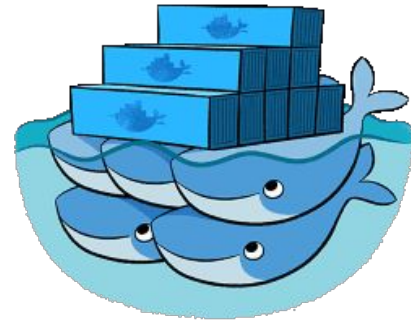
Es el motor de orquestación de contenedores más popular que existe en el mercado. Comenzó siendo un proyecto de Google. Actualmente, miles de equipos de desarrolladores lo usan para desplegar contenedores en producción. La herramienta funciona agrupando contenedores que componen una aplicación en unidades lógicas para una fácil gestión y descubrimiento.



kubernetes
..Google

Docker Swarm

Swarm es la solución que propone Docker ante los problemas de los desarrolladores a la hora de orquestar y planificar contenedores a través de muchos servidores. Viene incluido junto al motor de Docker y ofrece muchas funciones avanzadas integradas —como el descubrimiento de servicios, balanceo de carga, escalado y seguridad—.



Mesosphere **DC/OS**

El sistema operativo Mesosphere Datacenter (DC/OS) es una plataforma de código abierto, integrada para datos y contenedores desarrollados sobre el kernel de sistema distribuido Apache Mesos. Se ha diseñado para gestionar múltiples máquinas dentro de un centro de datos con uno o más clústeres, ya sea en la nube o usando software en servidores en local. DC/OS puede desplegar contenedores y gestionar tanto aplicaciones sin estado como protocolos con estado en el mismo entorno. Es capaz de funcionar con Docker Swarm y Kubernetes.



HashiCorp **Nomad**

Soportada por Linux, Mac y Windows, Nomad es una herramienta binaria única capaz de planificar todas las aplicaciones virtualizadas en contenedores o independientes. Nomad ayuda a mejorar la densidad, a la vez que reduce costos, ya que es capaz de distribuir de manera eficiente más aplicaciones en menos servidores.



Amazon ECS

El servicio de AWS es un sistema de gestión muy escalable que permite a los desarrolladores ejecutar aplicaciones en contenedores. Está formado por muchos componentes integrados que permiten la fácil planificación y despliegue de clústeres, tareas y servicios del contenedor.



Amazon
EC2 Container
Service (ECS)

Amazon Elastic Kubernetes Service

Amazon EKS facilita la implementación, la administración y el escalado de aplicaciones en contenedores mediante Kubernetes en AWS. Ejecuta la infraestructura de administración de Kubernetes por el usuario en varias zonas de disponibilidad de AWS para disminuir errores. Las aplicaciones que se ejecutan en cualquier entorno estándar de Kubernetes son totalmente compatibles y pueden migrar fácilmente a Amazon EKS.



Azure **Kubernetes** Service (AKS)

El servicio de Azure es código abierto y está optimizado para su uso en las máquinas virtuales de Azure, denominadas Azure Virtual Machines. Proporciona las herramientas necesarias para crear, configurar y gestionar la infraestructura de contenedores Docker abiertos. AKS ofrece desarrollo simplificado de aplicaciones basadas en contenedores y despliegue con soporte para Kubernetes, Mesosphere DC/OS o Swarm para la orquestación.



Google **Kubernetes** Engine (GKE)

Montado sobre Kubernetes, permite desplegar, gestionar y escalar aplicaciones de contenedores en la nube de Google. El objetivo de GKE es optimizar la productividad del departamento de desarrollo al mejorar la gestión de las cargas de trabajo basadas en contenedores. Oculta tanto las tareas de gestión simple como aquellas más complejas detrás de herramientas de líneas de comando, usando interfaces transparentes y fáciles de usar.

Obviamente, Kubernetes es la columna vertebral de GKE. Aunque no es estrictamente necesario dominar Kubernetes para usar GKE, ayuda mucho si al menos conocemos sus fundamentos básicos.



DigitalHouse>
Coding School