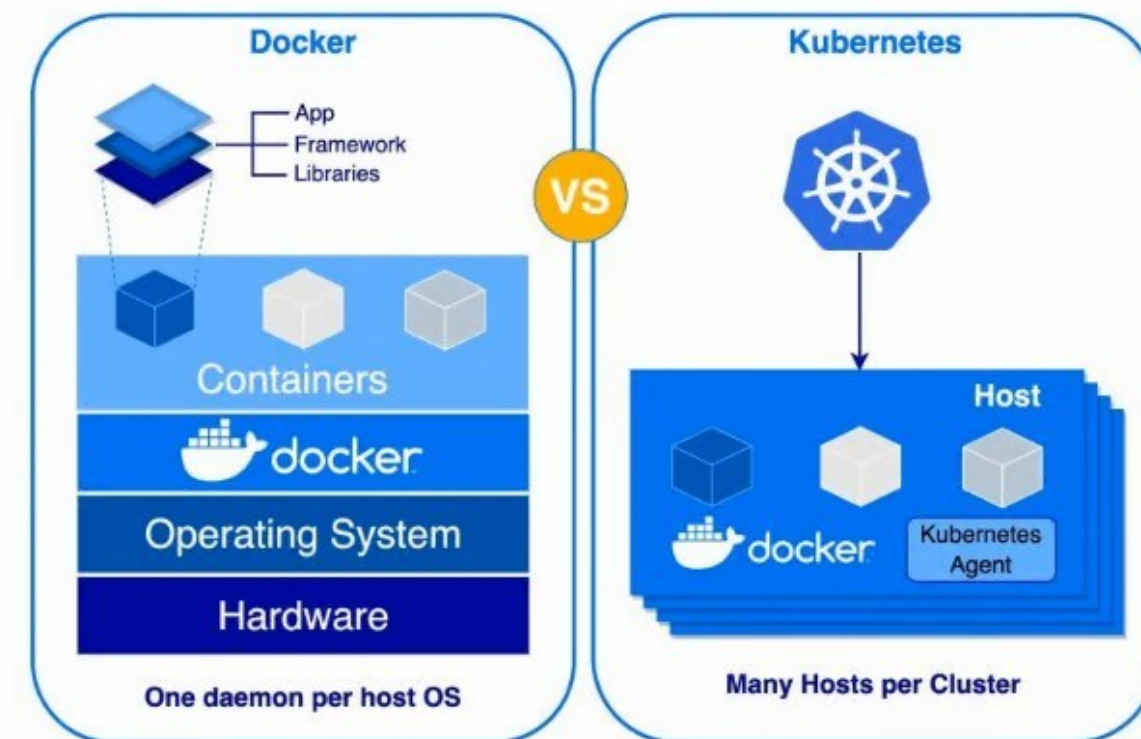


Introducció a Docker i Kubernetes

Docker i Kubernetes són dues tecnologies fonamentals en la gestió i desplegament d'aplicacions basades en contenidors. Mentre que Docker facilita la creació, l'execució i la distribució de contenidors, Kubernetes s'encarrega d'orquestrar i gestionar-los a escala. Aquestes eines han revolucionat el desenvolupament i l'operació de sistemes distribuïts, aportant agilitat, eficiència i escalabilitat.



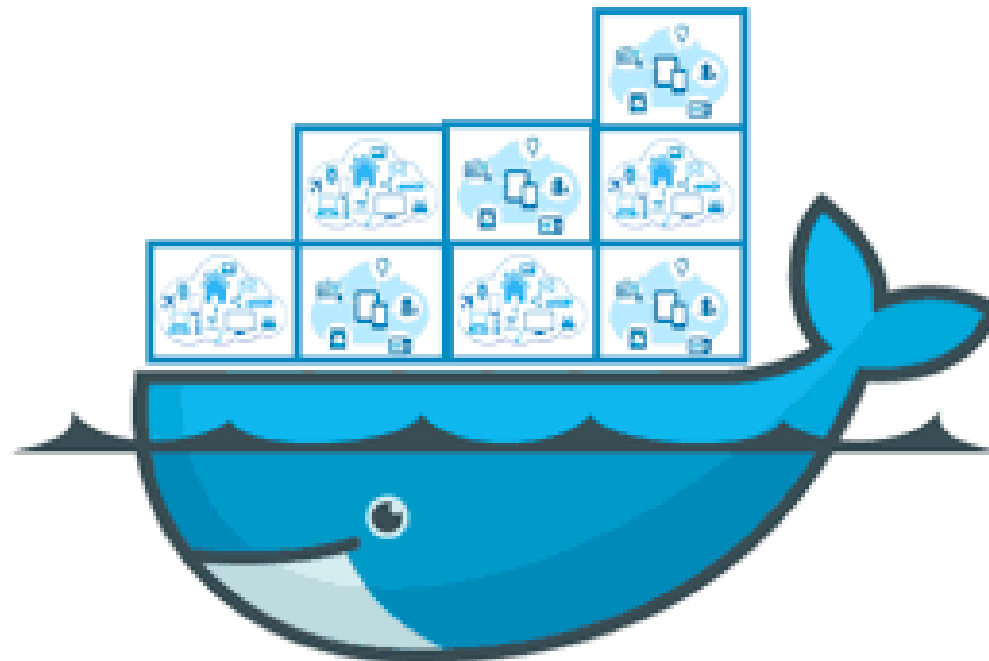
Docker Vs Kubernetes



Docker: Contenedors lleugers per a aplicacions

Què és Docker?

Docker és una plataforma de virtualització a nivell de sistema operatiu que permet permet empaquetar aplicacions i les seves dependències en contenidors lleugers, lleugers, portables i independents de la infraestructura subjacent. El principal avantatge de Docker és que garanteix la coherència de l'entorn entre el desenvolupament, desenvolupament, les proves i la producció, eliminant els clàssics problemes de "funciona de "funciona a la meva màq



Components clau de Docker

1 Docker Engine

És el motor d'execució d'execució de contenidors. Inclou un servei de fons (daemon), una API i una interfície de línia de comandaments (CLI).

4 Docker Hub

Un registre públic (o privat) on es poden allotjar i compartir imatges de Docker.

2 Docker Images

Una imatge és un conjunt de fitxers que conté tot el necessari necessari per executar una aplicació: el codi, les dependències, les configuracions, configuracions, etc. Es basen en capes, cosa que permet l'eficiència en l'emmagatzematge i distribució.

5 Docker Compose

Una eina que permet definir i gestionar aplicacions multi-contenedor mitjançant un fitxer docker-compose.yml.

3 Docker Containers

Són instàncies en execució d'una imatge. A diferència de les màquines virtuals, comparteixen el nucli del sistema operatiu host, fent-los els més lleugers i ràpids.

Com funciona Docker?

1

Dockerfile

Es crea una imatge amb un Dockerfile, que especifica les instruccions d'instal·lació i configuració.

2

Docker build

Es construeix la imatge amb docker build.

3

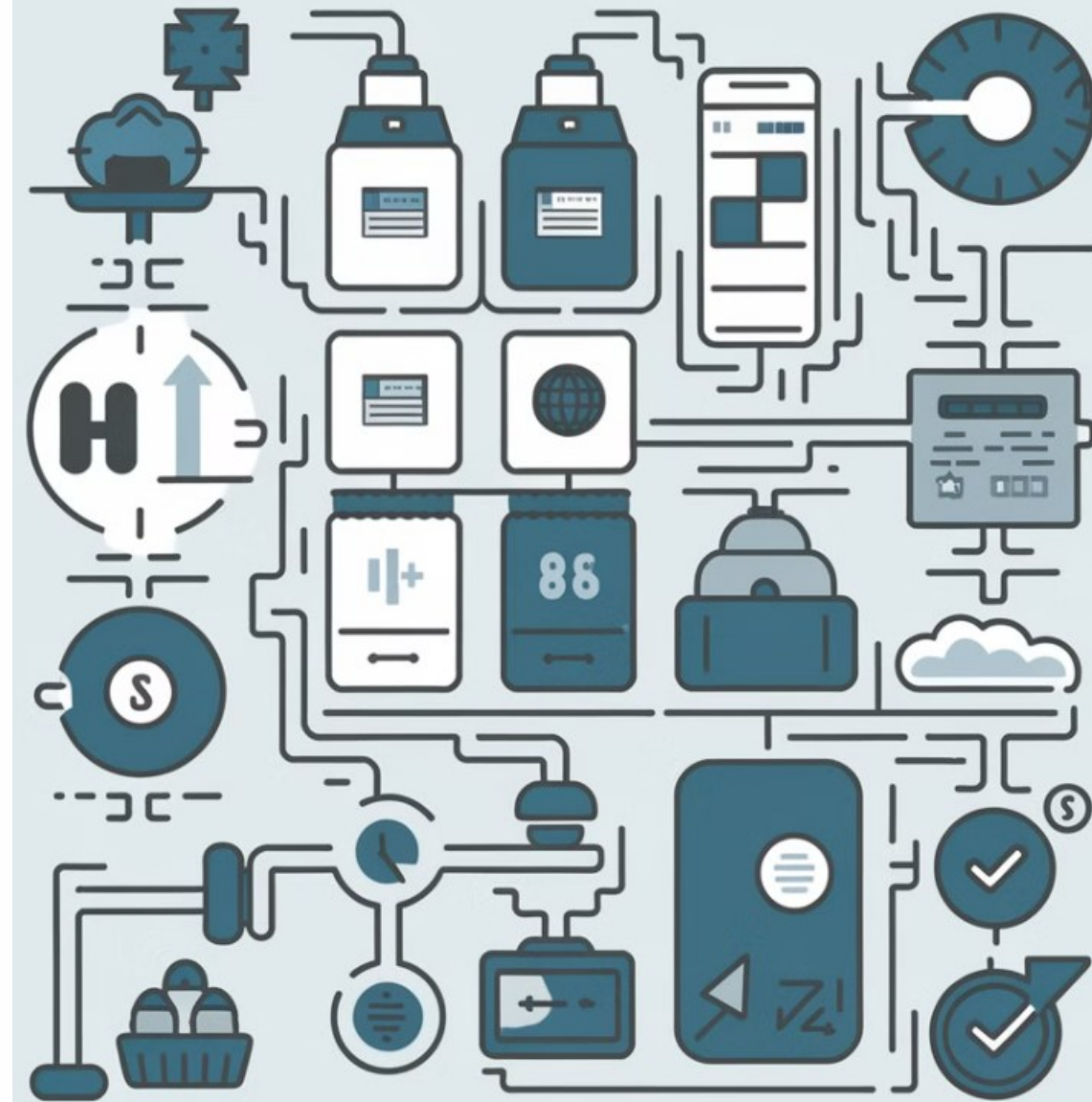
Docker run

Es desplega un contenidor amb docker run.

4

Gestió

Docker gestiona l'execució, el networking i l'emmagatzematge del contenidor.





Avantatges de Docker



Portabilitat

Els contenidors poden executar-se en qualsevol sistema que tingui Docker.



Eficiència

Consumeixen menys recursos que les màquines virtuals.



Ràpid desplegament

Es poden iniciar i destruir contenidors en segons.



Escalabilitat

Permet l'execució de múltiples instàncies d'una aplicació fàcilment.

Kubernetes: Orquestració de contenidors a gran escala

Què és Kubernetes?

Kubernetes (o K8s) és una plataforma d'orquestració de contenidors de codi obert desenvolupada desenvolupada originalment per Google i ara mantinguda per la Cloud Native Computing Foundation (CNCF). Permet automatitzar el desplegament, l'escalat i la gestió d'aplicacions contenitzades. Si bé Docker facilita la creació i l'execució de contenidors, Kubernetes permet gestionar múltiples contenidors en diferents màquines, garantint alta disponibilitat, disponibilitat, balanceig de càrrega i recuperació automàtica d'errors.

Components clau de Kubernetes

Node

És una màquina (física o virtual) dins del clúster de Kubernetes que executa contenidors.

Pod

És la unitat bàsica d'execució a Kubernetes. Pot contenir un o més contenidors que comparteixen xarxa i emmagatzematge.

Deployment

Un Deployment defineix i gestiona la creació i actualització de rèpliques de pods.

Service

Un objecte que exposa una aplicació executant-se en un pod per a ser accessible des de l'exterior o per altres pods dins del clúster.

Ingress

Un component que permet gestionar el trànsit extern cap als serveis dins del clúster.

Components clau de Kubernetes (cont.)

1 ConfigMap i Secret
Permeten gestionar configuracions i informació sensible de manera segura.

2 Kubelet
És l'agent que s'executa a s'executa a cada node i i garanteix que els contenidors funcionin correctament.

3 Etcd
Base de dades distribuïda on Kubernetes emmagatzema l'estat l'estat del clúster.

4 Controller Manager
Component que executa diversos controladors per gestionar l'estat del sistema.

5 Scheduler
Assigna els pods als nodes segons els segons els recursos disponibles i altres altres polítiques.

Com funciona Kubernetes?

1

Definició

L'usuari defineix l'estat desitjat d'una aplicació en un fitxer YAML (ex. Deployment, Service, etc.).

2

Desplegament

Kubernetes desplega els pods en nodes disponibles segons els recursos i requisits.

3

Supervisió

Un controlador supervisa l'estat actual i el compara amb l'estat desitjat.

4

Autoreparació

Si un pod falla, Kubernetes el reinicia automàticament.

5

Escalabilitat

Kubernetes pot escalar l'aplicació segons la càrrega.

Avantatges de Kubernetes



Alta disponibilitat

Distribueix la càrrega entre múltiples nodes. nodes.



Escalabilitat automàtica

Pot augmentar o reduir el nombre de pods segons la demanda.



Autoreparació

Si un pod falla, Kubernetes el recrea automàticament.



Gestió eficient de recursos

Assegura una distribució òptima de les càrregues de treball. treball.



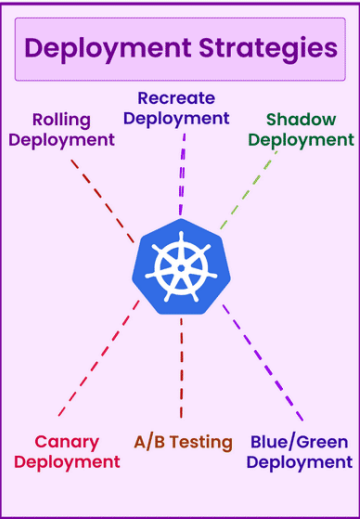
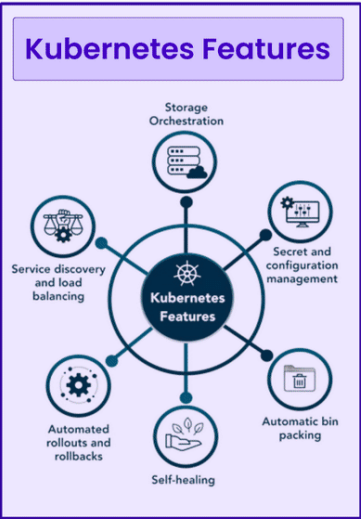
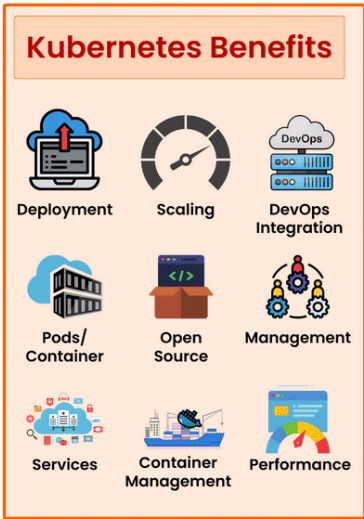
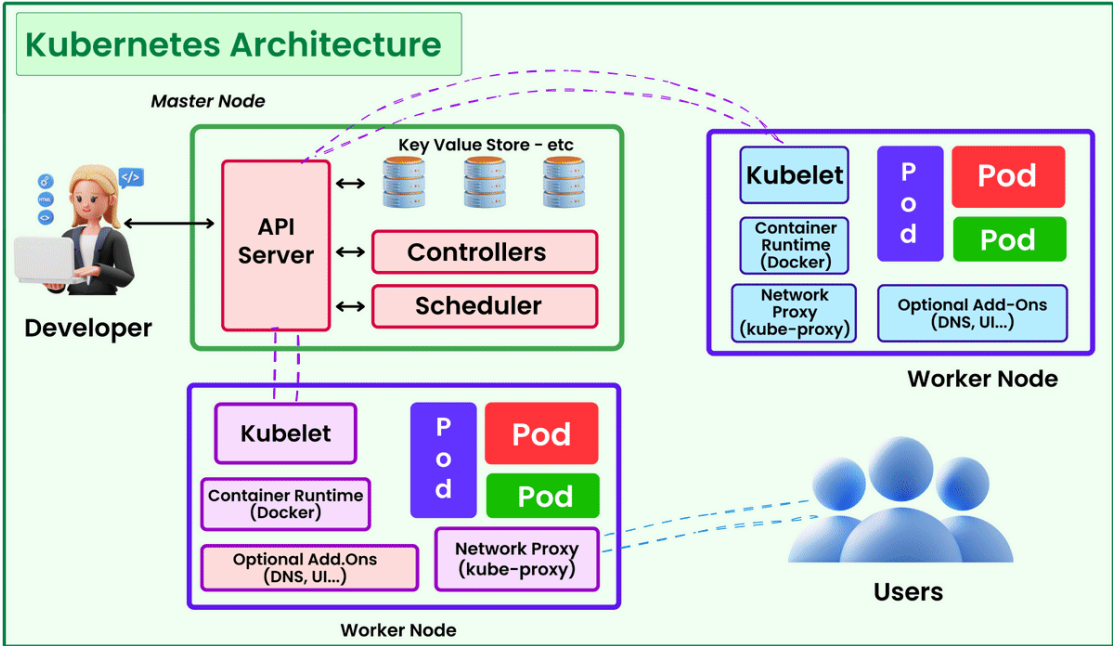
Integració amb qualsevol infraestructura

Funciona en entorns on-premise, núvols públics i híbrids.

Docker vs Kubernetes: Comparació i Complementarietat

Característica	Docker	Kubernetes
Contenidors	✓	✓
Orquestració	✗	✓
Escalabilitat	✗	✓
Autoreparació	✗	✓
Networking avançat	✗	✓
Facilitat d'ús	✓	✗ (més complex)

Docker és ideal per a entorns de desenvolupament i aplicacions petites, mentre que Kubernetes és essencial per a desplegaments grans i distribuïts. Kubernetes sol utilitzar Docker (o altres motors de contenidors com containerd) com a backend per executar els contenidors.





Conclusió

Docker i Kubernetes són dues tecnologies complementàries que han transformat la manera com es desenvolupen, es despleguen i s'executen aplicacions. Mentre Docker permet empaquetar aplicacions en contenidors, Kubernetes proporciona les eines per gestionar-los de manera eficient i escalable en producció. La combinació de totes dues eines és la base de l'arquitectura moderna de DevOps i computació en núvol.