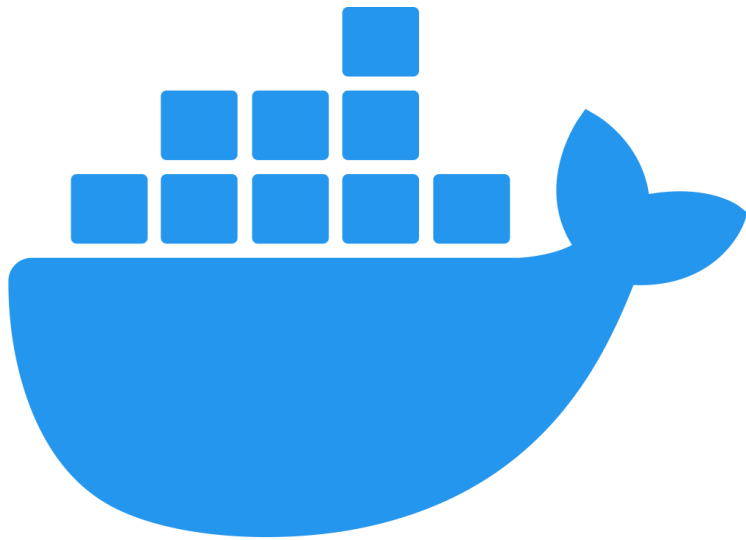


Docker Fundamentos



- Cgroups
- kernel namespaces
- OverlayFS

docker®



Flask

web development,
one drop at a time



2.7



EC2

Modernizar



Flask

web development,
one drop at a time



3.7



EC2

ch**root** 1979

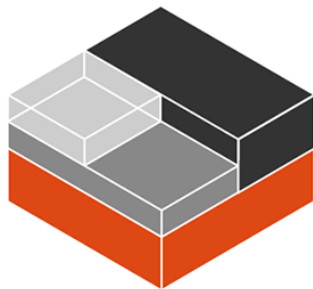


2000

FreeBSD®



OpenVZ 2005



LXC

2008

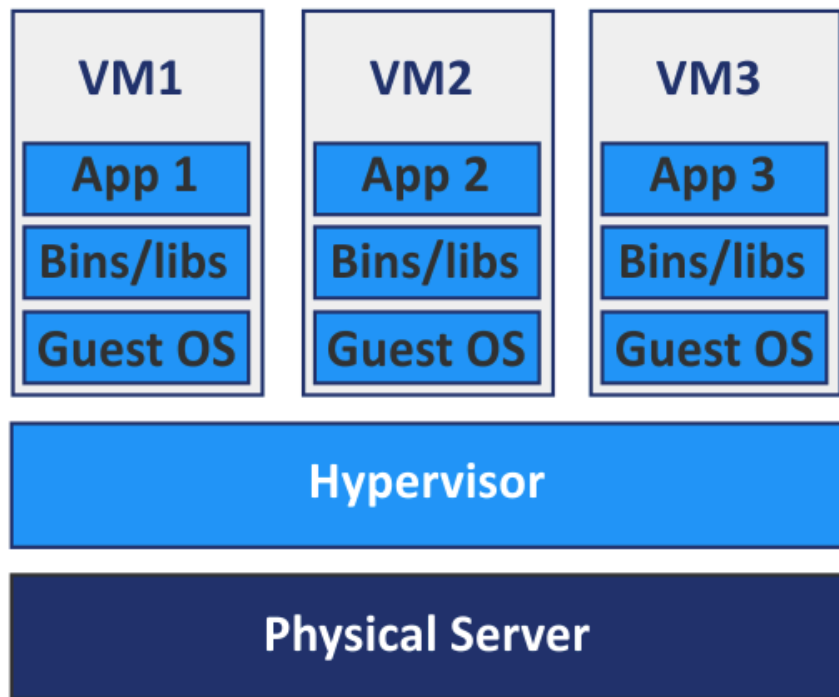
Acerca de Docker

- Software gratuito
- Permite crear entornos independientes y aislados para lanzar y desplegar sus aplicaciones. Estos entornos se les llama **contenedores**.
- El desarrollador puede ejecutar un contenedor en cualquier máquina.

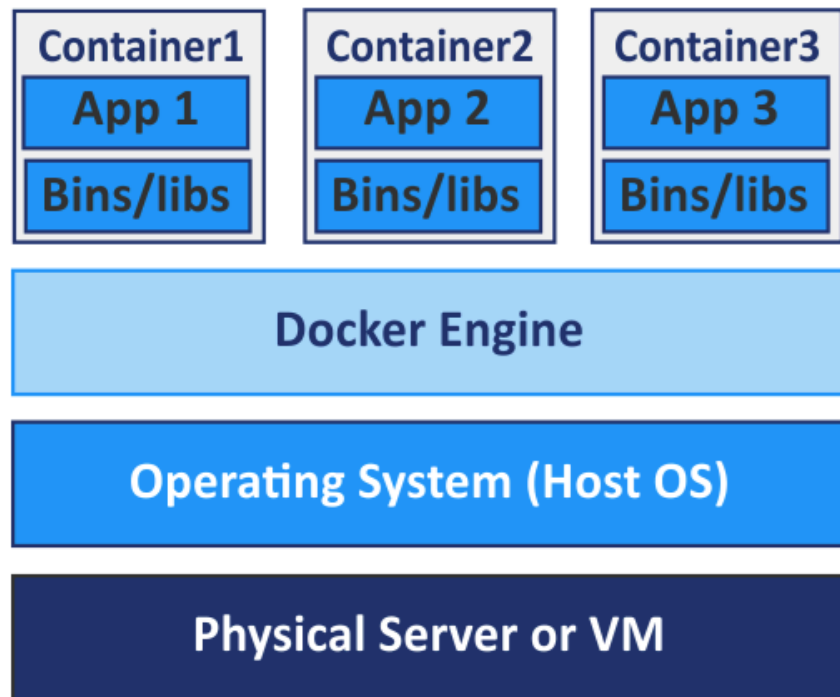
¿Es Docker una máquina virtual?

- Parece una máquina virtual pero la funcionalidad no es la misma.
- A diferencia de Docker, una máquina virtual incluirá un sistema operativo completo. Trabaja de forma independiente y actuará como un ordenador.
- Docker sólo comparte los recursos de la máquina anfitriona para ejecutar sus entornos.

Virtual Machines



Containers



Comandos útiles

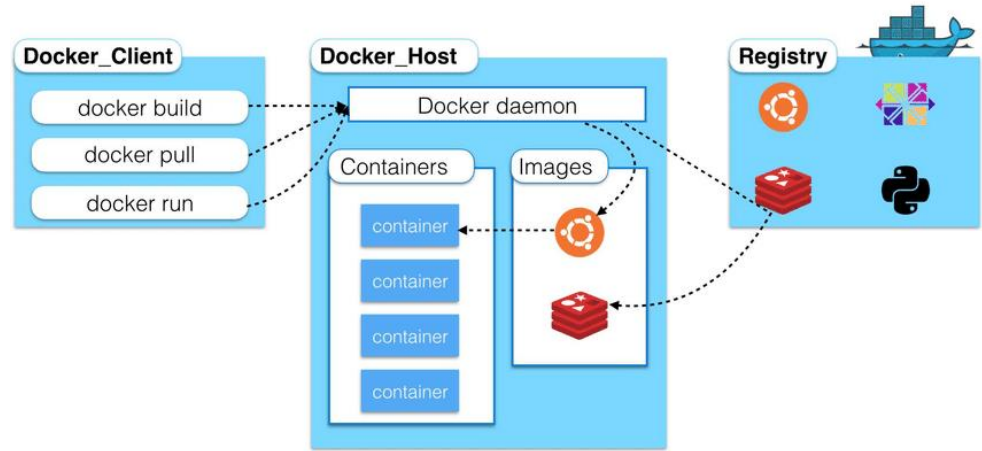
- **docker version** : Ver versión instalada
- **docker run hello-world** : Ejecuta contenedor para validar que esta bien instalado docker.
- **docker pull busybox** : Extrae la imagen de busybox del registro Docker y la guarda en el sistema.
- **docker images** : Ver todas las imagenes que estan en nuestro sistema.
- **docker ps** : Muestra todos los contenedores corriendo en el sistema.
- **docker ps -a** : Muestra todos los contenedores que se corrieron en el sistema.

Comandos útiles

- **docker run -it** **busybox sh** : Ejecutar run con el argumento -it permite tener acceso a la terminal, lo que permite ejecutar comandos en el contenedor.
- **docker rm** **container_id** : Elimina el contenedor creado.
- **docker rm \$(docker ps -a -q -f status=exited)** : Elimina los contenedores con estado terminado.
- **docker container prune** : La última versión de Docker viene ya con prune que limpia los contenedores terminados.
- **docker rmi** **image_id** : Elimina una imagen.

Terminologia

- **Imágenes** : Imagen de la aplicación que forman la base de los contenedores (docker pull descarga la imagen).
- **Contenedores** : Se crean a partir de las imágenes y ejecutan la aplicación (docker run crea un contenedor).
- **Docker Daemon** : Es el servicio que ejecuta el host que administra la construcción de contenedores Docker.
- **Docker Client** : La herramienta de línea de comandos que permite al usuario interactuar con el Daemon.
- **Docker Hub** : Registro de imágenes de Docker.



High-level overview of Docker architecture

Referencias

- <https://docker-curriculum.com>
- <https://hub.docker.com>

Información del repositorio usado:

- <https://towardsdatascience.com/docker-jupyter-for-machine-learning-in-1-minute-30e1df969d09>