

Docker under the hood

Introducción a containerd

- Carlos David Ramirez Altan
- racarlosdavid@gmail.com
- <https://github.com/racarlosdavid>

Containers

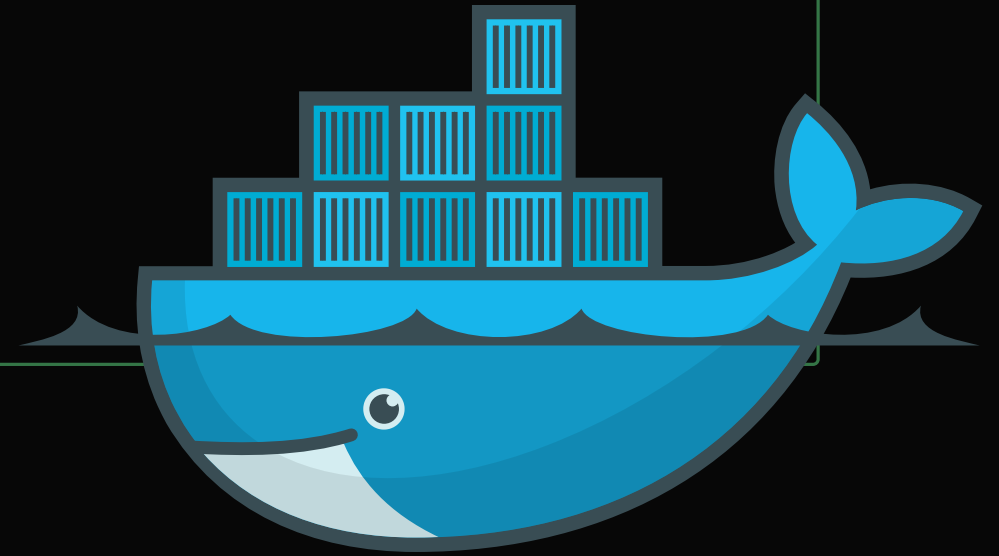
¿Qué es un contenedor ?

Es un ambiente aislado que cuenta con todo lo necesario para que una aplicación se ejecute.

Permite mayor portabilidad para su fácil distribución.

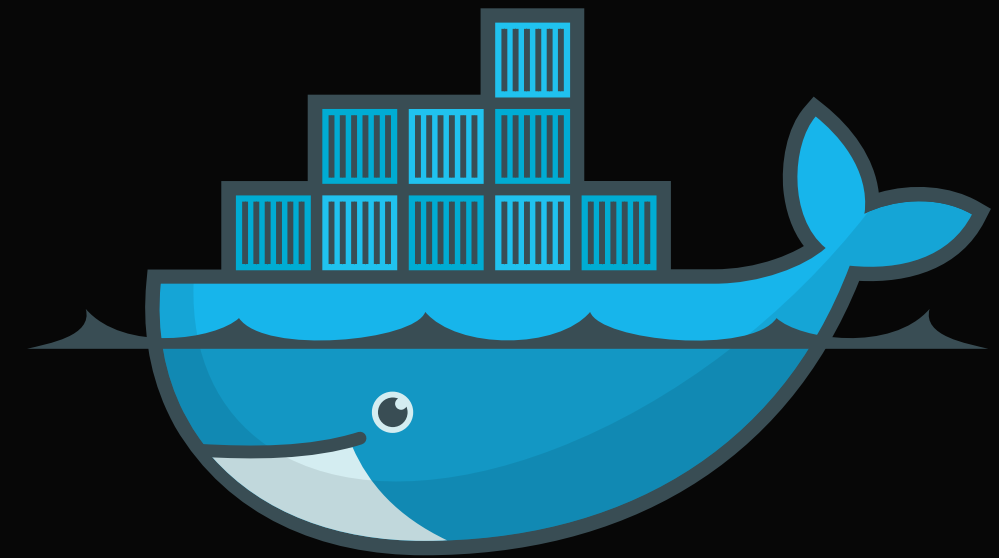
Permite mayor eficiencia en la forma en que se despliegan aplicaciones

Docker

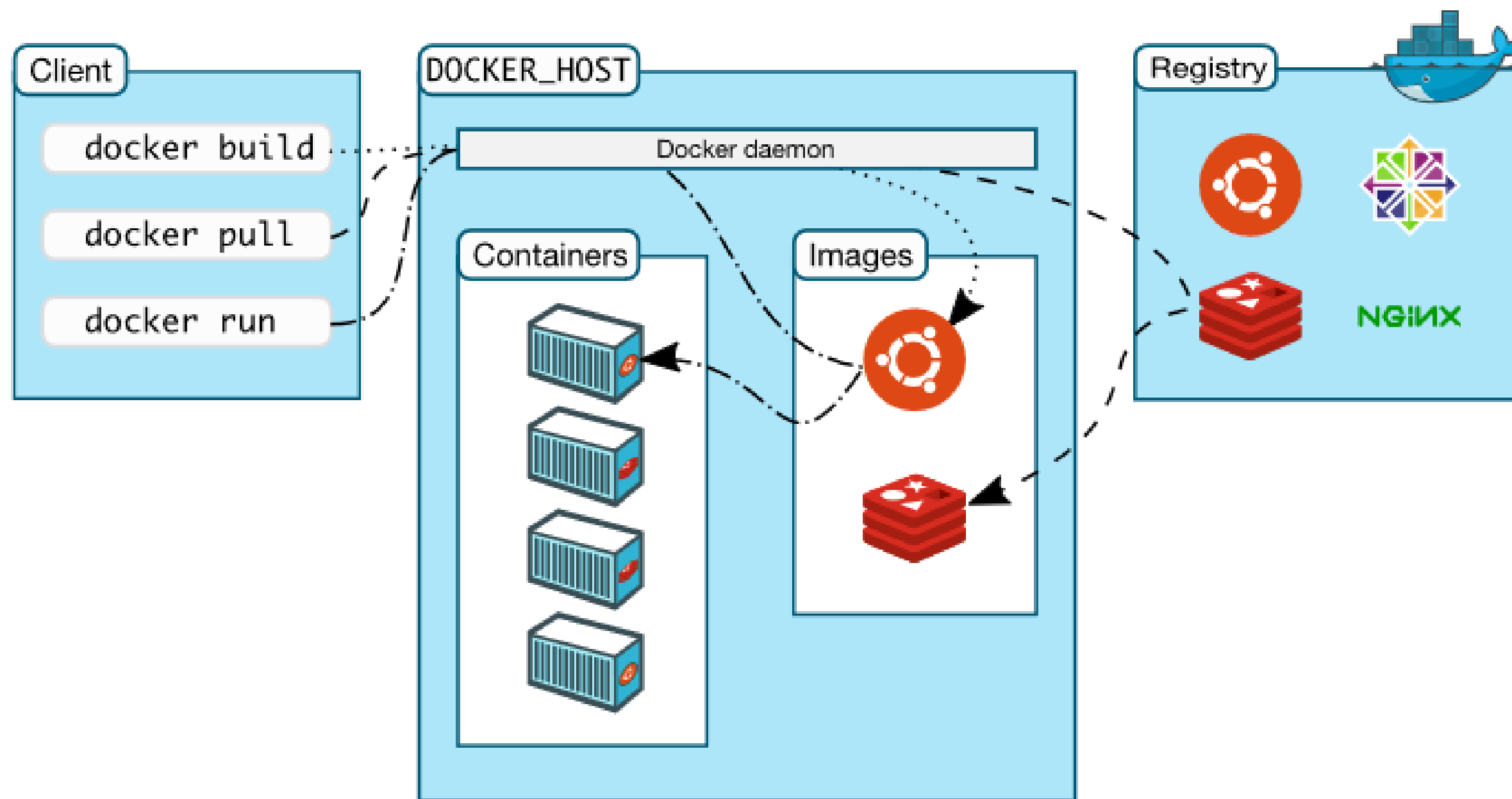


Docker

Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.



Arquitectura Docker



Algunos Comandos

- docker pull
- docker images
- docker ps
- docker ps -a
- docker rm
- docker rmi
- docker start
- docker stop
- docker build
- docker run

Docker under the hood

Open Container Initiative (OCI)

- Open Container Initiative (OCI) es un proyecto de la Linux Foundation para diseñar estándares abiertos para contenedores.
- Establecido en junio de 2015 por Docker y otros líderes en la industria de contenedores.
- Actualmente, OCI contiene dos especificaciones: la especificación de tiempo de ejecución (runtime-spec) y la especificación de imagen (image-spec).

Container runtimes

¿Qué es un container runtime?

Container runtimes se centran en ejecutar contenedores, configurar namespace y cgroups para contenedores, y también se denominan lower-level container runtimes. Higher-level container runtimes or container engines se centran en los formatos, el desempaquetado, la gestión y el uso compartido de imágenes. También proporcionan API para desarrolladores.

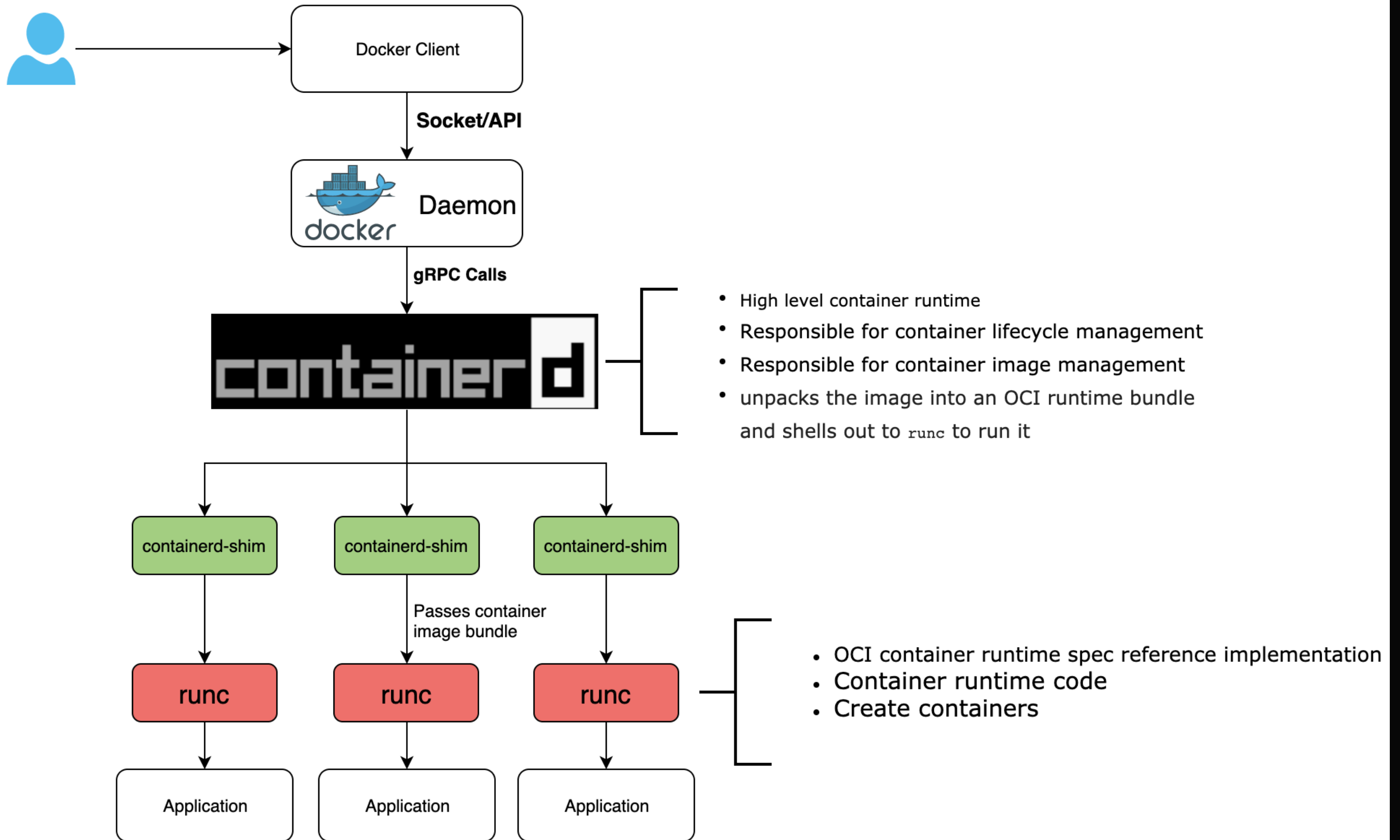
Algunos container runtimes:

- **containerd**: An industry-standard container runtime with an emphasis on simplicity, robustness and portability.
- **runc**: is a CLI tool for spawning and running containers on Linux according to the OCI specification.
- **cri-o**: lightweight container runtime for kubernetes
- **rkt**: is an application container engine developed for modern production cloud-native environments.

Algunos de runtimes como "rkt" hacen la mayoría de las tareas por sí mismos, mientras que los tiempos de ejecución como "containerd" hacen algunas de las funciones de alto nivel y usan otras como "runc" para algunas de las tareas de bajo nivel.

Dependencias de containerd

- runc: to run containers
- ctr: A CLI for containerd
- containerd-shim: to support daemonless containers



ecosystem

Platform



Client



kubelet



CRI Runtime



Container Engine Pouch

containerd client



BuildKit

containerd client

ctr

containerd client

containerd

API



CRI



containerd client



containerd



Service Handlers



Prometheus

Metrics

Core

Services

Containers Service

Content Service

Diff Service

Images Service

Leases Service

Namespaces Service

Snapshots Service

Tasks Service

Metadata (namespaced)

Containers

Content

Images

Leases

Namespaces

Snapshots



Backend

Content Store

plugin

local

Snapshotter

overlay

btrfs

devmapper

native

windows

plugin

Runtime

v2 shim client

ttrpc



containerd-shim



runc



runhcs



kata



Firecracker



gVisor



shim

system



Instalación

Como interactuar con containerd

— ctr

— nerdctl

— crictl

Algunos Comandos

— `ctr images pull`

— `ctr images ls`

— `ctr c ls`

— `ctr task ls`

— `ctr images rm`

— `ctr images ls`

— `ctr c rm`

— `ctr task kill`

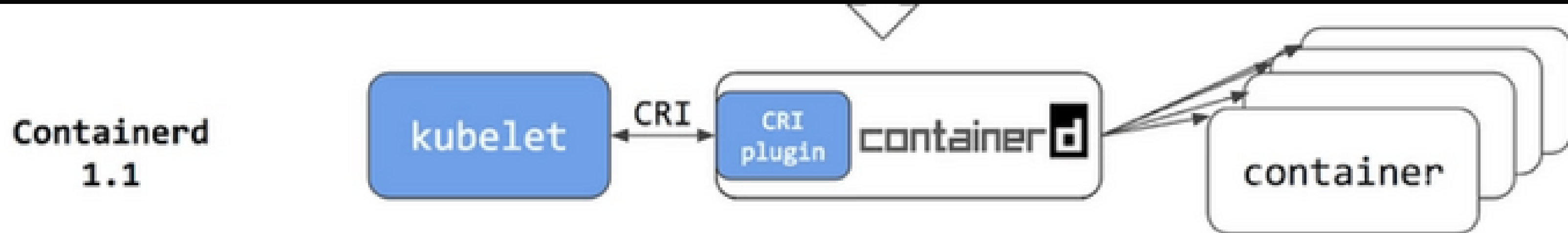
— `ctr run`

Demo

K8S & Docker

CRI (Container runtime interface)

K8s releases CRI (container runtime interface), which unifies the container runtime interface. Any container runtime that supports CRI can be used as the underlying container runtime of k8s.



Gracias por su atención