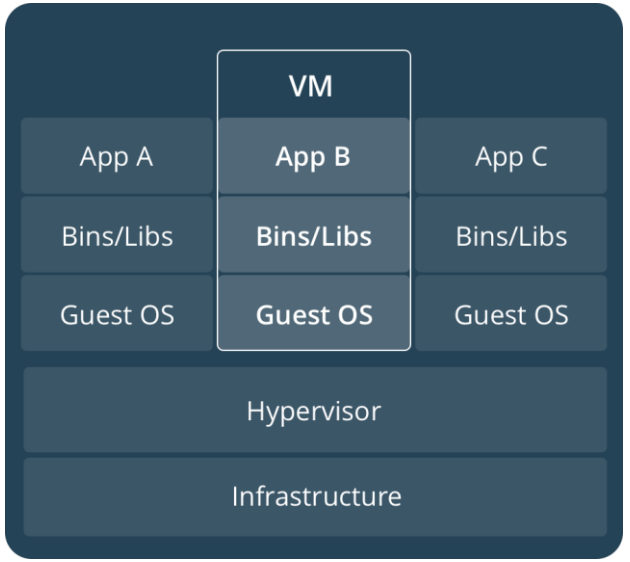
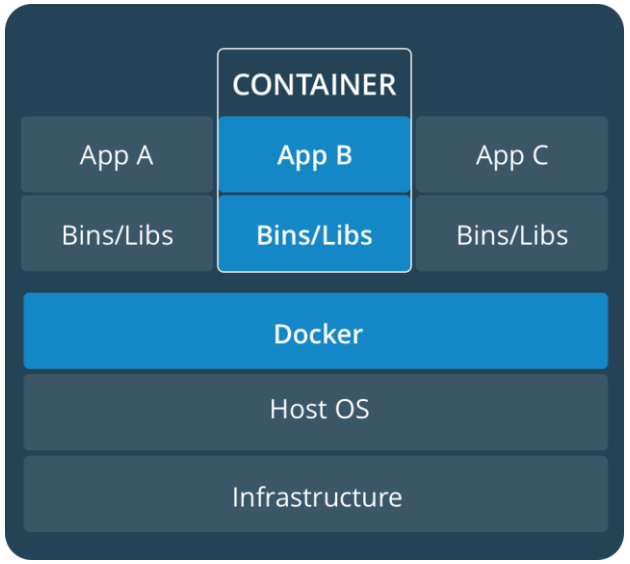
Cloud & Networking

Docker

VMs vs containers

- VM
  - VM is a complete operating system running on a hypervisor
  - hardware resources are fully virtualized
  - VM assumes running on its own computer hardware
  - disk image
- Containers
  - software and resource (files, processes, users) abstractions
  - isolated extensions of the host kernel
  - application assumes running on its own operating system





Docker containers

For what?

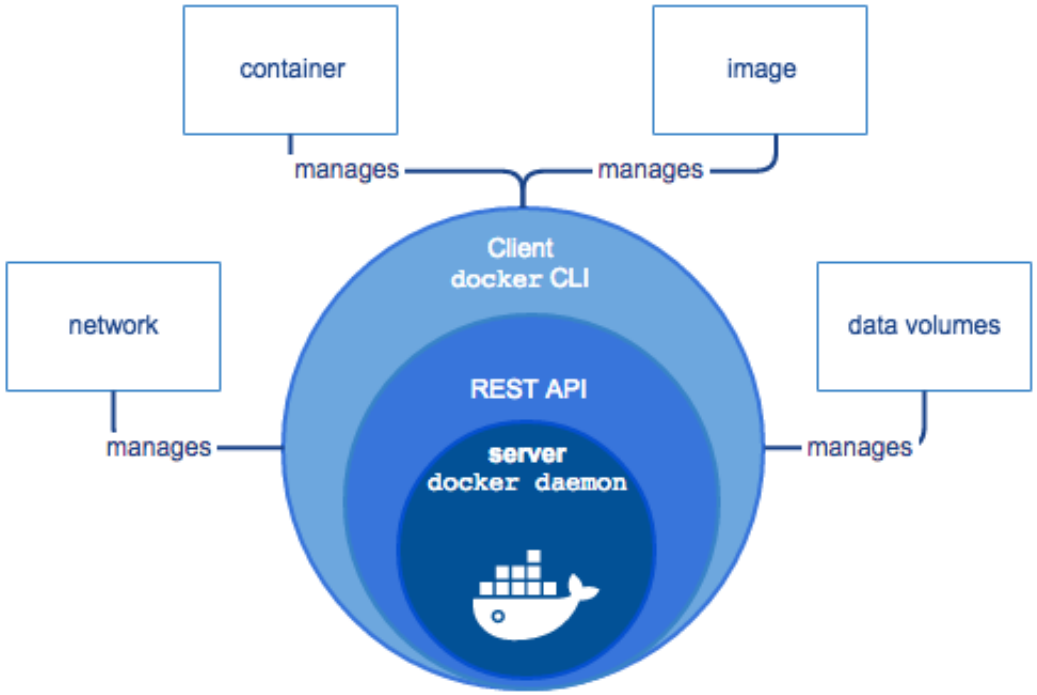
- a basic computing wrapper that can run on any infrastructure, portable applications, migration
- create sandboxes to test applications and services without leaving a mess behind in the Host OS
- scale modern applications built from components (microservices)
- Workload multiplexing

Docker Terminology

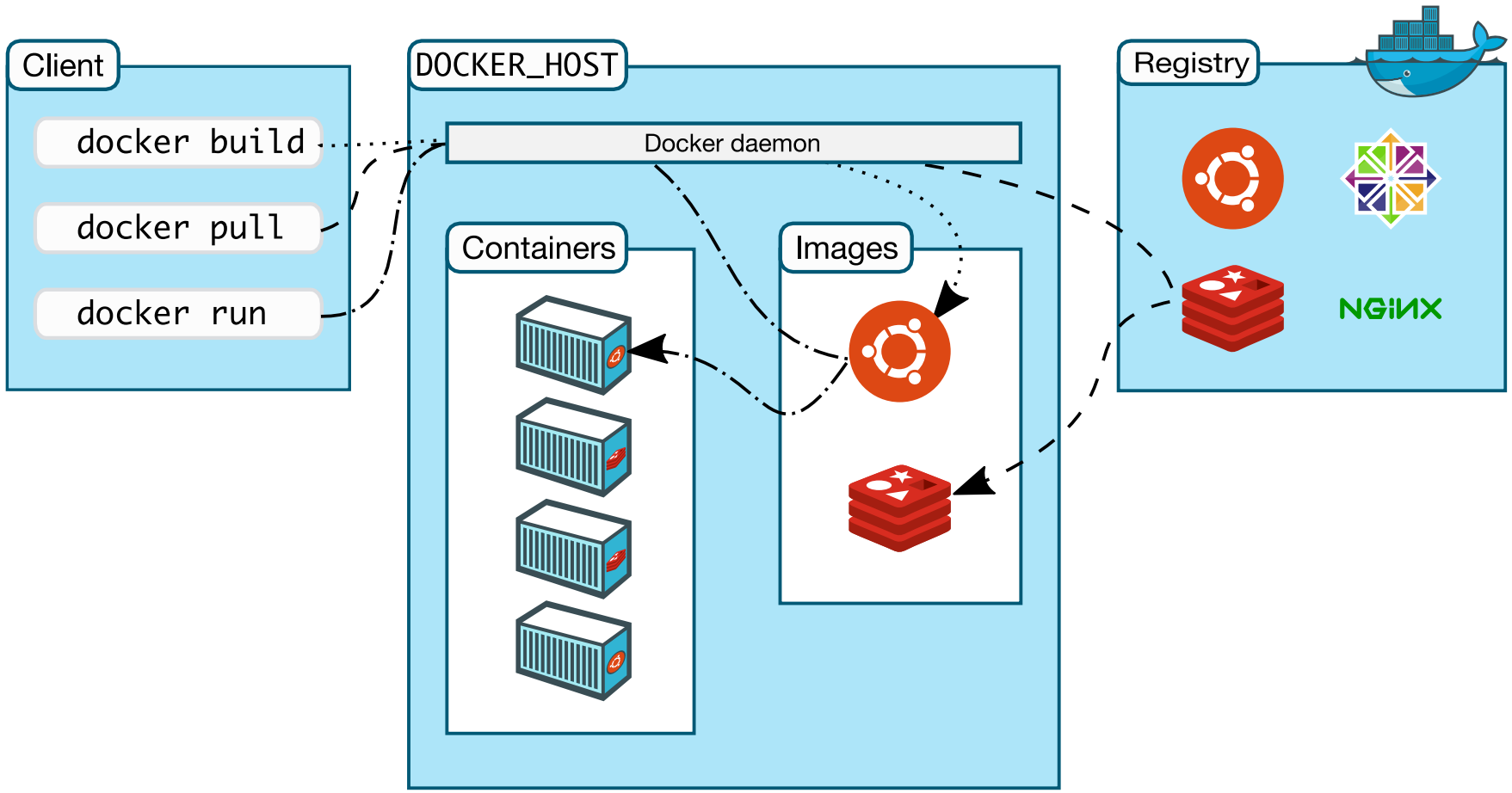
- A *container* is a runtime instance of a *docker image*.
  - loosely isolated environment for applications
  - the unit for distributing and testing applications
- A *docker image* is an ordered collection of root filesystem changes, typically contains a union of layered filesystems stacked on top of each other. An image does not have state and it never changes.
- A *Dockerfile* is a text document that contains all the commands you would normally execute manually in order to build a Docker image.
- The Docker Hub (registry) is a centralized resource for working with Docker and its components for hosting Docker images.
- A *volume* is a specially-designated directory within one or more containers that bypasses the Union File System. Volumes are designed to persist data, independent of the container's life cycle.

Docker Engine

- client-server application with three major components
  - A server which is a type of long-running program called a daemon process (the dockerd command).
    - creates and manages Docker *objects*, such as images, containers, networks, and volumes
  - A REST API which specifies interfaces that programs can use to talk to the daemon and instruct it what to do.
  - A command line interface (CLI) client (the docker command).

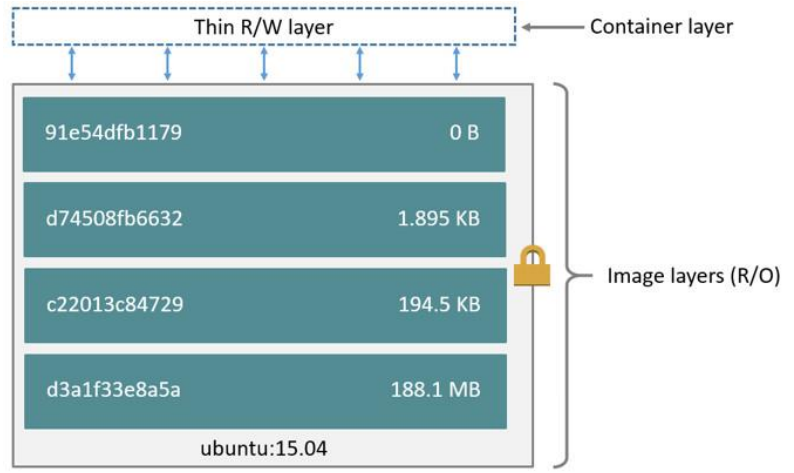


Docker Architecture



What Docker is built on?

- Docker is written in Go and builds on the following Linux kernel features
  - namespace: provides isolated environment for a container
    - namespaces: pid (process), net (networking), ipc (inter process communication), mnt (mount points), uts (hostname and domain name, UTS: Unix Time Sharing)
  - Control groups (cgroups): limits an application to a specific set of resources (e.g. CPU share, memory)
  - Union file systems: UnionFS creates layers for images



Frequent docker commands

docker ps	
docker ps –a	
docker images	docker image ls
docker inspect <container> or <image>	
docker run -d <image> <command>	
docker start/stop <container>	
docker exec -it <container> bash	
docker rm –f <conatiner>	
docker container prune	
docker logs <container>	

Sources

- https://docs.docker.com