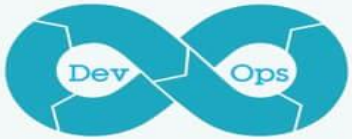

docker

Development Process

waterfall

agile

devops



Application Architecture

monolithic

n:tier

microservices

Deployment & Packaging

physical servers

virtual servers

containers

Application Infrastructure

datacenter

hosted

cloud

What is Virtualisation?

Virtualization refers to importing a guest operating system on a host operating system, allowing developers to run multiple OS on the same host.

- Infrastructure
Host machine - A server or local machine

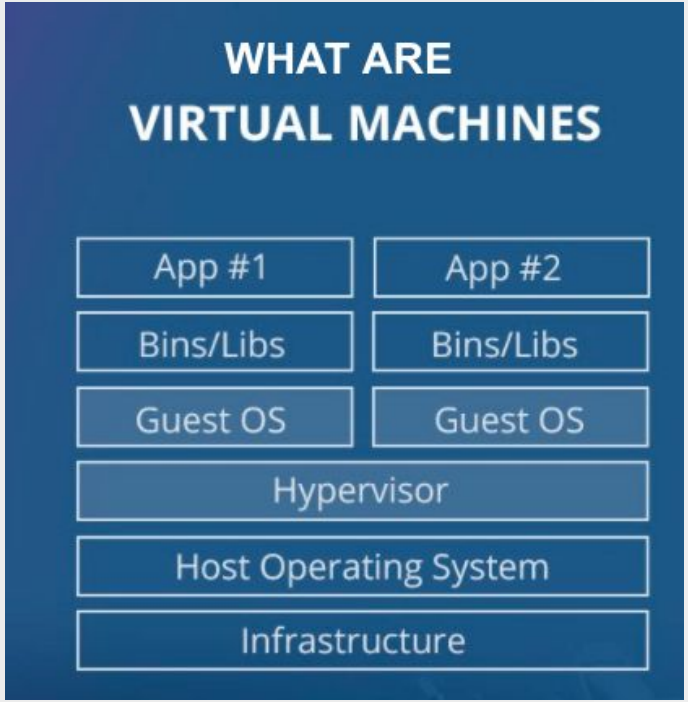
- Host Operating system
OS installed on host machine (Linux / Windows / MAC ...)

- Hypervisor
Virtual machine software, medium between VM and Host

- Guest OS
User required OS to run in VM (Linux / Windows / MAC ...)

- Bins / Libs
Binary files and Libraries for installed on the OS

- App
User application



Common dependency issues

- It worked yesterday
- It works on my machine, maybe it's your machine’s defect.
- It worked in dev and staging, have you installed it properly in production?

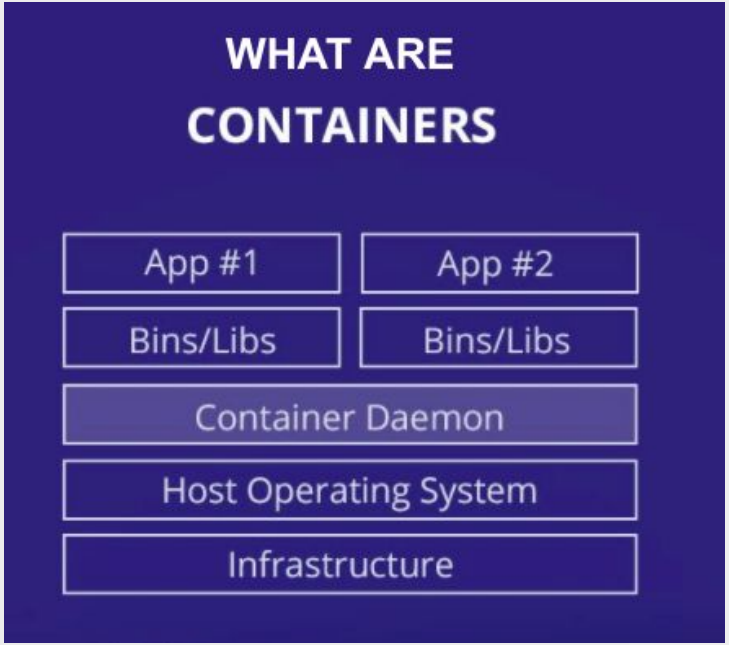
Solution

Let’s pack everything into a container and ship it.

What is containerization?

Containerization is defined as a form of operating system virtualization, through which applications are run in isolated user spaces called containers, all using the same shared operating system.

Virtualization	containerization
Requires the complete OS installation for every VM	Installs the container only on a host OS
A kernel is installed for every virtualized OS	Uses only the kernel of the underlying host OS
Heavyweight	Lightweight
Limited performance	Native performance
Fully isolated	Process-level isolation

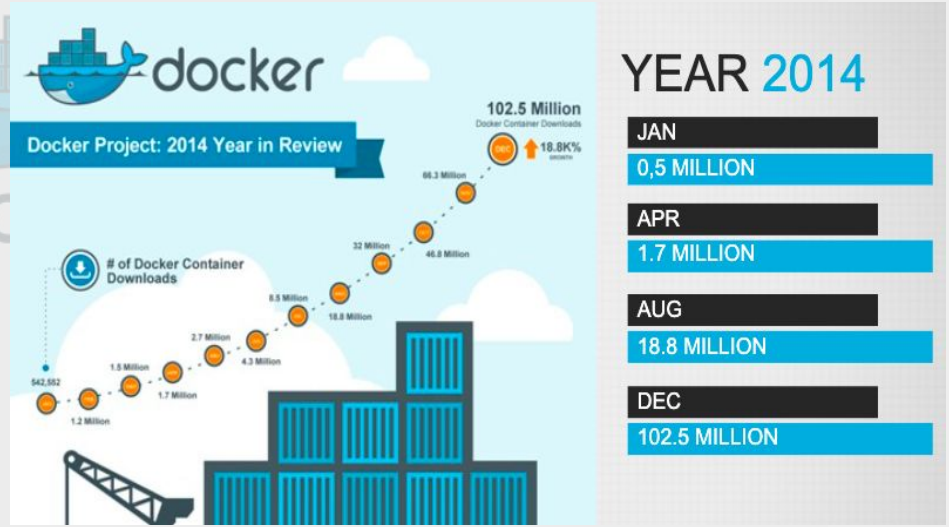


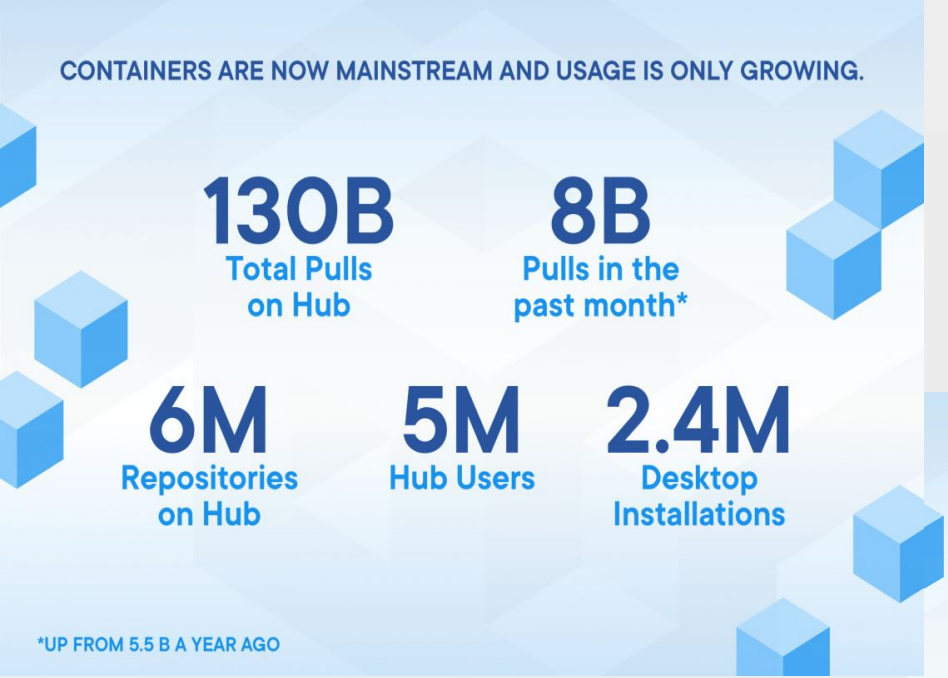
What is docker?

- Docker is an open platform for developing, shipping, and running applications.
- By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.
- It compatible with Windows, MAC and most of the Linux distributions.

History

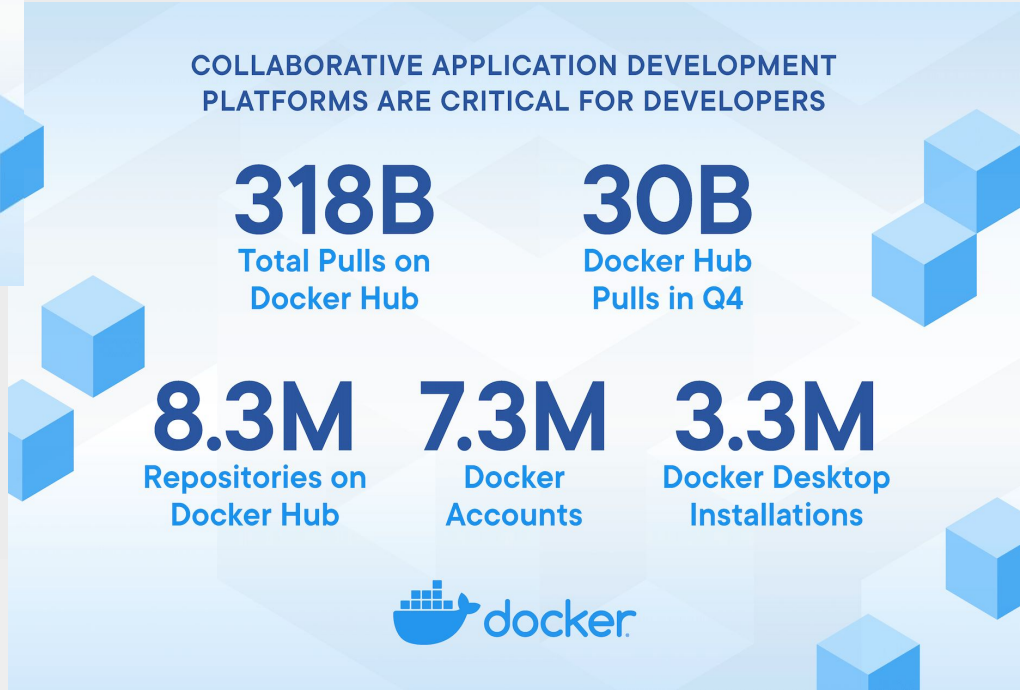
- Started in 2010 and released in 2011 as dotCloud project.
- Released as OpenSource in 2013, March.
- Redhat, Microsoft, IBM and Amazon started using and integration of Docker by 2014.





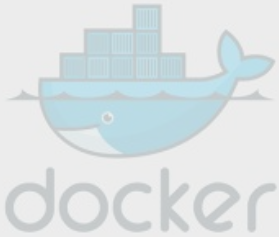
Feb 2020

Feb 2021



What is the hype about?

- No worries of missing dependencies with safe method of environment portability.
- Run multiple containers of same or different environments in the same machine.
- Automate development, testing and deployment with CICD methods.
- Maintain version control per dev, staging and production.
- Replication and scaling.
- High availability.
- Efficient utilization of physical resources.
- Ease of role and duties.
- Great community support.



Developer duties

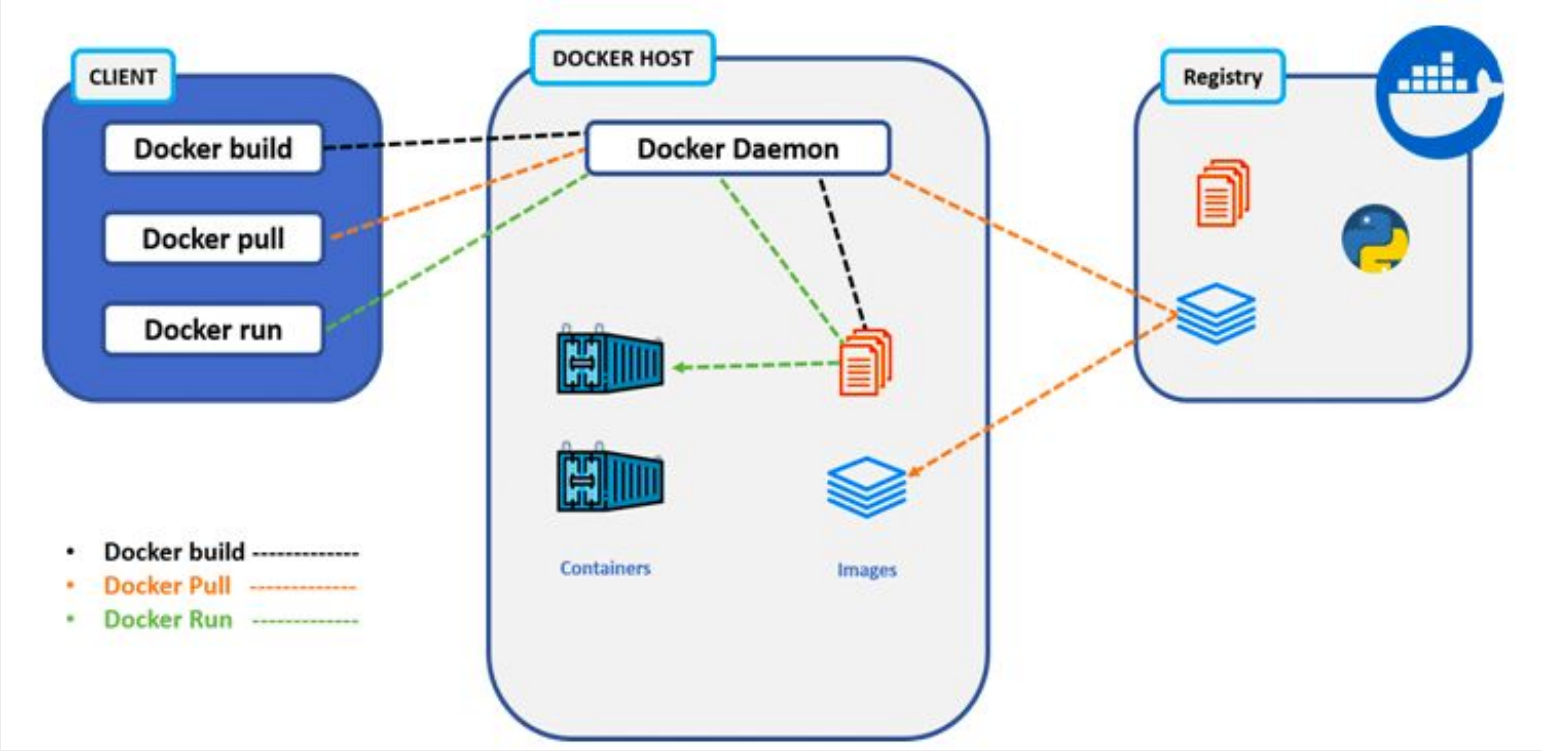
- Focus on what’s inside the container.
- Code, dependencies, apps, data.
- Docker file.

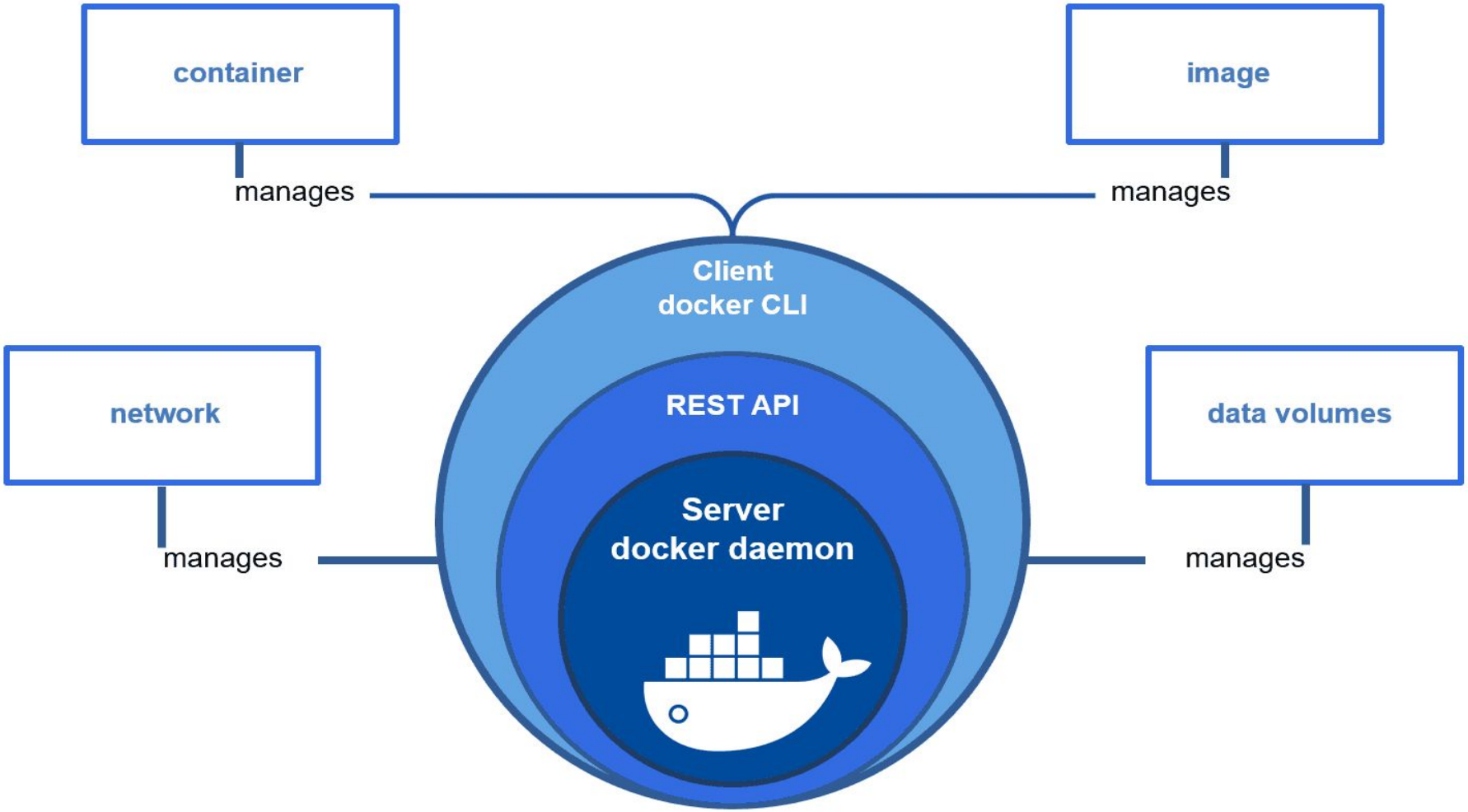
DevOps duties

- Focus on what’s outside the container.
- Logging, remote access, monitoring, config files.
- Maintenance, replication, scaling.

Docker: What's Under the Hood?

Docker Architecture




container
image
manages
manages
Client
docker CLI
REST API
Server
docker daemon
network
data volumes
manages
manages

Installation on Windows

Docker docs: https://docs.docker.com/docker-for-windows/install/

Installation on MAC

Docker docs: https://docs.docker.com/docker-for-mac/install/

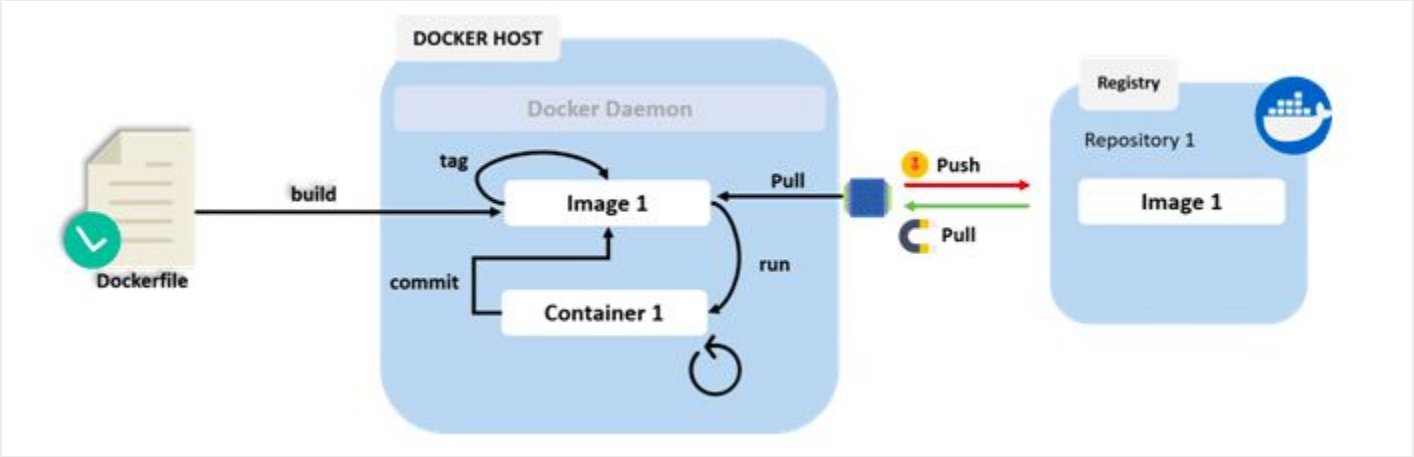
Installation on Ubuntu

Docker docs: https://docs.docker.com/engine/install/ubuntu/
Quick way: https://github.com/kmsharsha/docker/blob/master/install.sh

Run a container

docker run hello-world

Functional flow of docker run




DEMO

Demo: (Calc App)

1. Build a docker image
   - Check local image registry
   - Check building image without cache
2. Run the image
   - Run the image as container
   - Run the container in background
   - Run the container in interactive mode
3. Test API calls

Reference: https://github.com/kmsharsha/docker