Docker

“Open platform for developers and sysadmins to build, ship, and run distributed applications, whether on laptops, VMs or the cloud”

Pirghie Dimitrie

Containers

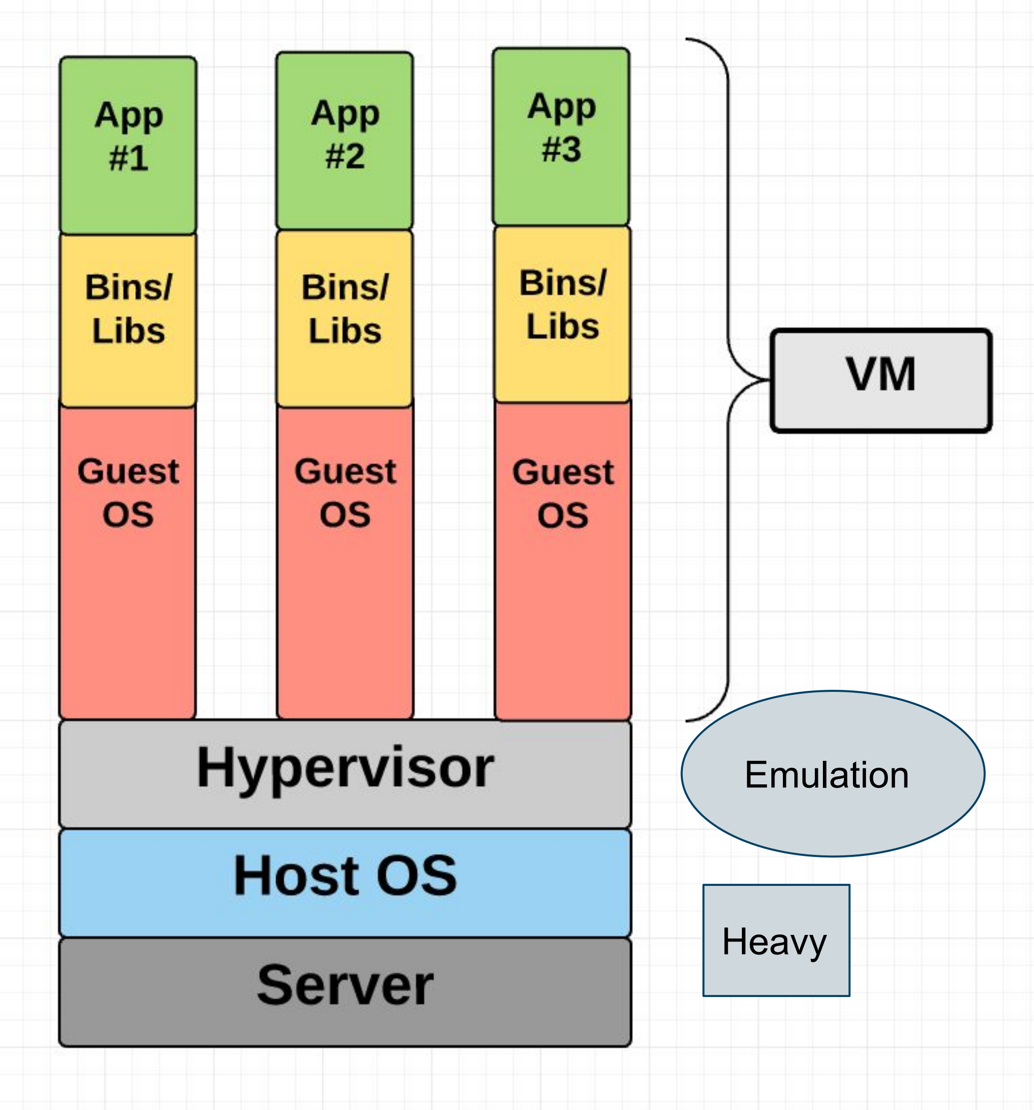
Where are containers ?

Different containers

OS native or virtual

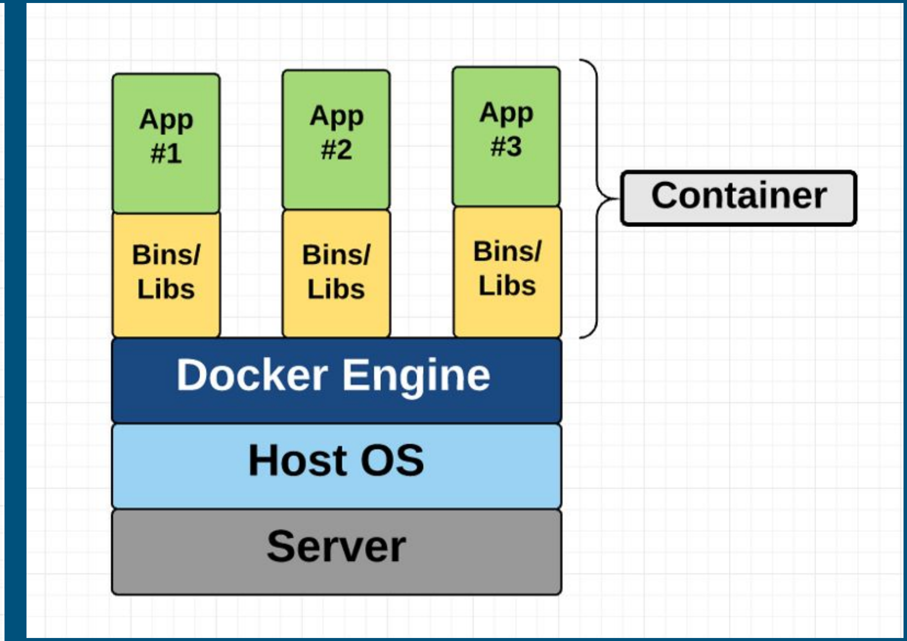
Cloud or Sea ?

VM vs Container



Emulation

Heavy



lightweight

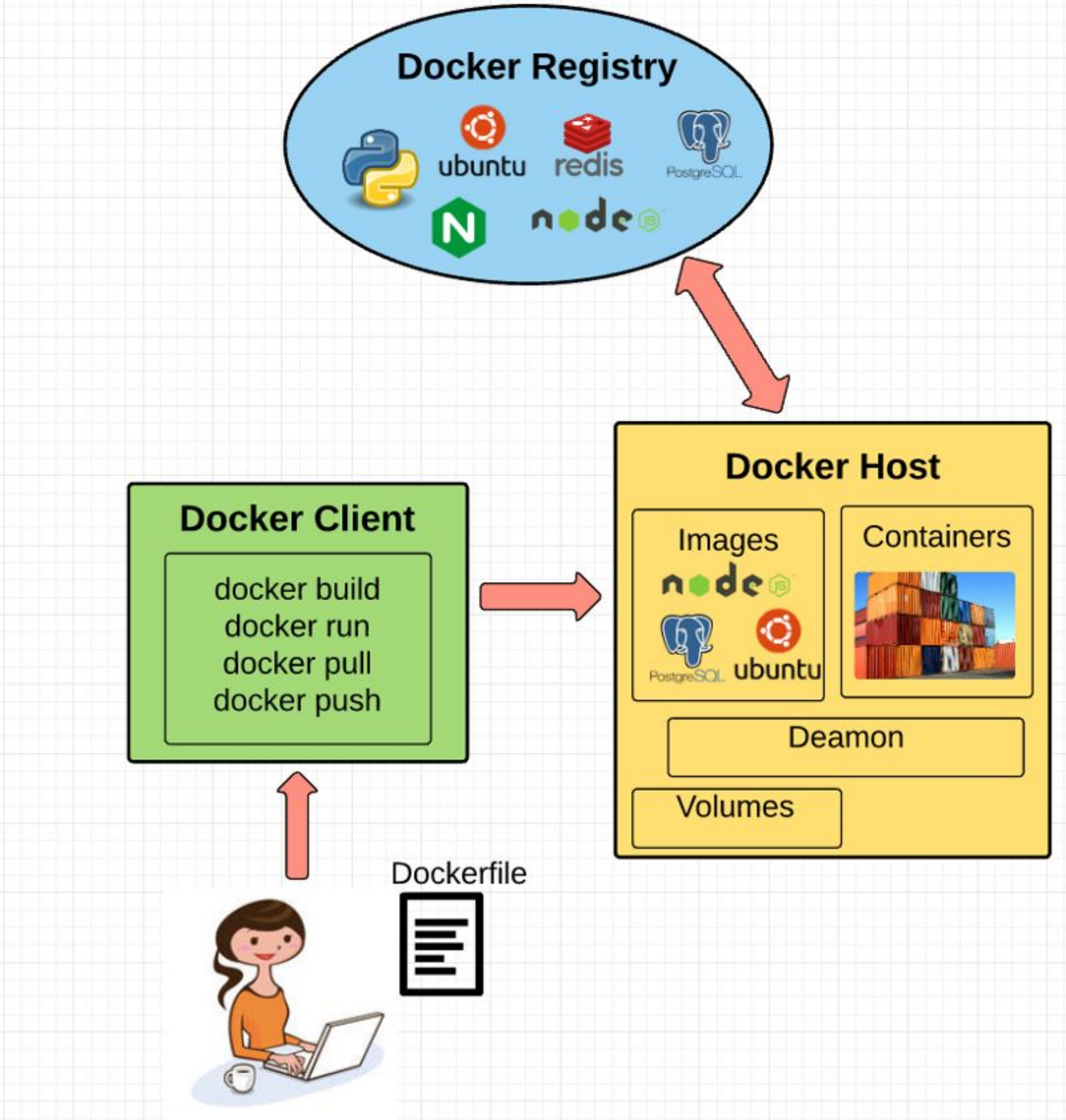
OS-level
virtualization
-
“user-space”

Where does Docker come in ?

1. open-source
2. Uses Linux kernel features like namespaces and control groups

- Easy to use - “build once, run anywhere”
- Speed - “sandboxes or user-space, not hardware emulation like VMs”
- Docker Hub - “like GitHub but for sandboxes - Docker Images”
- Modularity and Scalability - for example: docker images of MongoDB, Redis and Node.js application.

Docker concepts



Docker engine

Layer where Docker runs, lightweight runtime and tooling that manges containers, images, builds, and more.

Is made up of:

1. Docker daemon - runs on host machine.
2. Docker client that communicates with Docker Daemon to executes commands.
3. REST API for interacting with the Docker Daemon remotely.

Docker Client

Is what you, end-user of Docker communicate with.

UI for docker. -> Client communicates instructions to Daemon

```
docker build dipi/some_awesome_image .

docker run mongo:latest

docker start mongo

docker ps

docker images list
```

Docker Daemon

Executes commands sent from the Client or API.

- Building, running and distributing containers.

Dockerfile

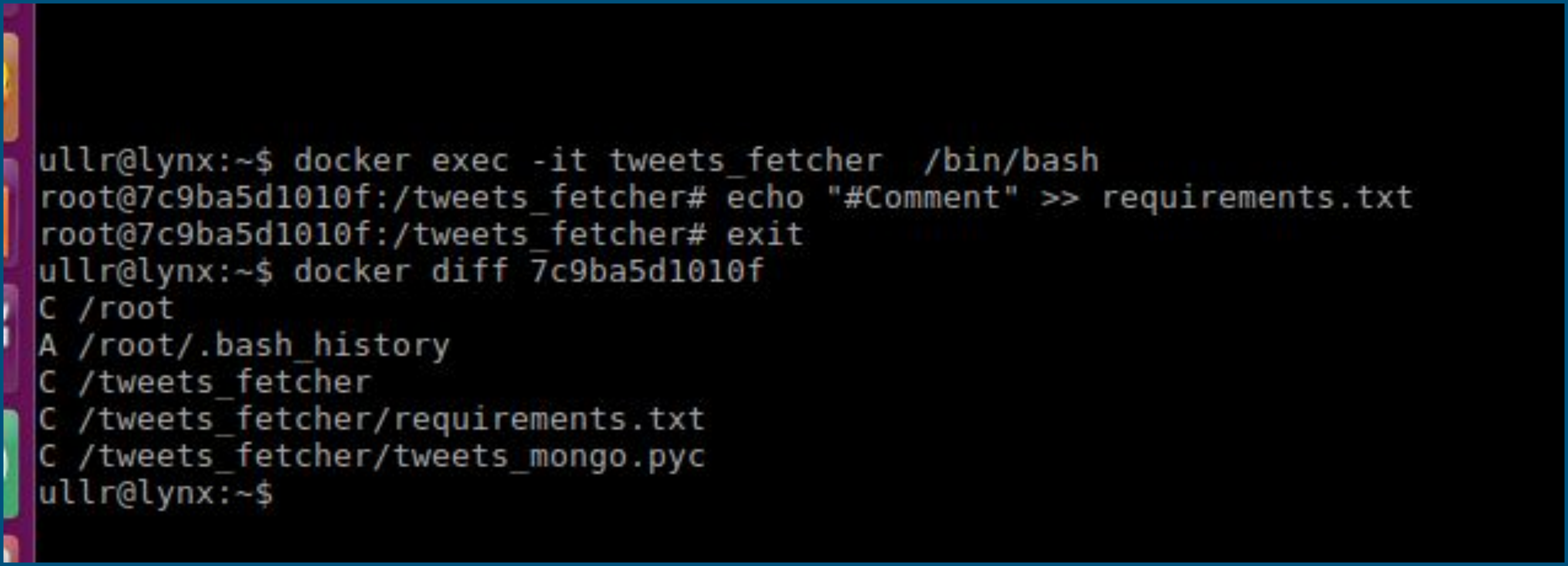
Describe the recipe for building a docker image.

```
FROM ubuntu:latest

MAINTAINER Dimitrie Pirghie "dimitriepirghie94@gmail.com"

RUN apt-get update -y

RUN apt-get install -y python python-pip

COPY . /tweets_viewer

WORKDIR /tweets_viewer

RUN pip install -r requirements.txt

ENTRYPOINT ["python"]

CMD ["tweets_viewer.py"]
```

Example

```
#Build images

cd tweets_fetcher

sudo docker build -t tweets_fetcher_img .

cd tweets_viewer

sudo docker build -t tweets_viewer_img .

# Run mongo

docker run -d --name mongo

# Run fetcher

docker run -e PYTHONUNBUFFERED=0 -d --name tweets_fetcher --link mongo:mongo --env-file ./env.list tweets_fetcher_imgsteps

# Run viewer

docker run -d --name tweets_viewer --link mongo:mongo -p 5000:5000 tweets_viewer_img
```

Is like Git



```
ullr@lynx:~$ docker exec -it tweets_fetcher  /bin/bash
root@7c9ba5d1010f:/tweets_fetcher# echo "#Comment" >> requirements.txt
root@7c9ba5d1010f:/tweets_fetcher# exit
ullr@lynx:~$ docker diff 7c9ba5d1010f
C /root
A /root/.bash_history
C /tweets_fetcher
C /tweets_fetcher/requirements.txt
C /tweets_fetcher/tweets_mongo.pyc
ullr@lynx:~$
```

DEMO

Docker ToolBox

1. Compose - defining and running multi-container Docker applications
2. Machine - install Docker Engine on virtual hosts OS X, Windows, DigitalOcean, AWS ...
3. Kitematic - GUI for Docker Engine installation and automation.

Finish

https://github.com/dimitriepirghie/docker_presentation