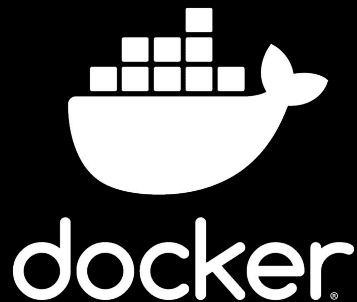




Docker e Docker compose



Mateus Roveda



Sumário

1. Apresentação
2. O que são containers
3. O que é o Docker
4. Benefícios
5. O que é o Docker compose
6. Conexão SSH
7. Instalando o Docker
8. Instalando o Docker compose
9. Comandos úteis Docker
10. Na prática
11. Dockerfile
12. Construindo a nossa imagem
13. Executando o nosso build e acessando
14. docker-compose.yml
15. Trabalhando com volumes
16. Criando um ambiente de desenvolvimento WordPress com Docker
17. Extras..



Mateus Roveda



Cursando Bacharelado em Engenharia de Software -
UNIVATES

Tutor do CRIE_TI

Engenheiro DevOps na LibreCode

Presidente Diretório Acadêmico de Eng. de Software

Youtuber no youtube.com/ROVEEb

Líder de comunidade na Universidade Brasileira
Livre

Palestrante e organizador de eventos

[Twitter](#) [Github](#) [GitLab](#)

[Telegram](#) [LinkedIn](#) [Blog](#)

[Youtube](#)



Containers

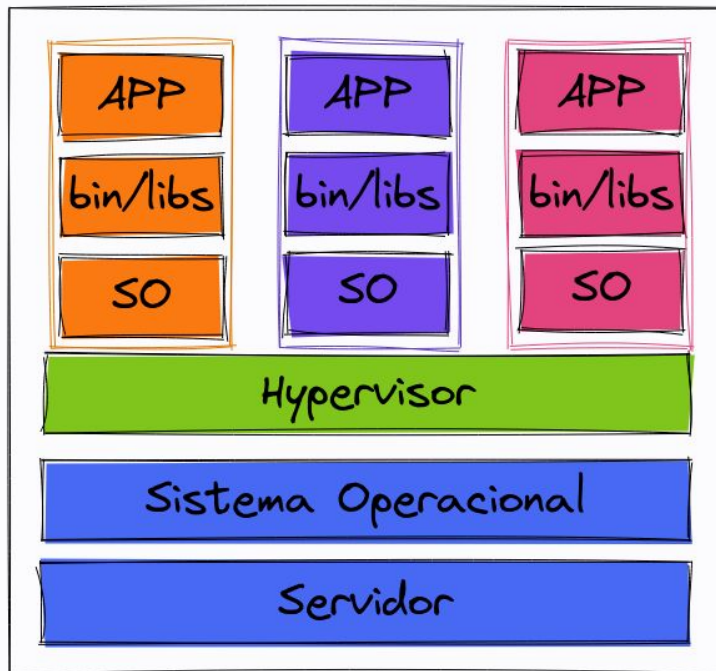
- Ambiente isolado;
- Caixinhas de aplicações;
- Contém dependências;
- Executados a partir de imagens.

Material de apoio

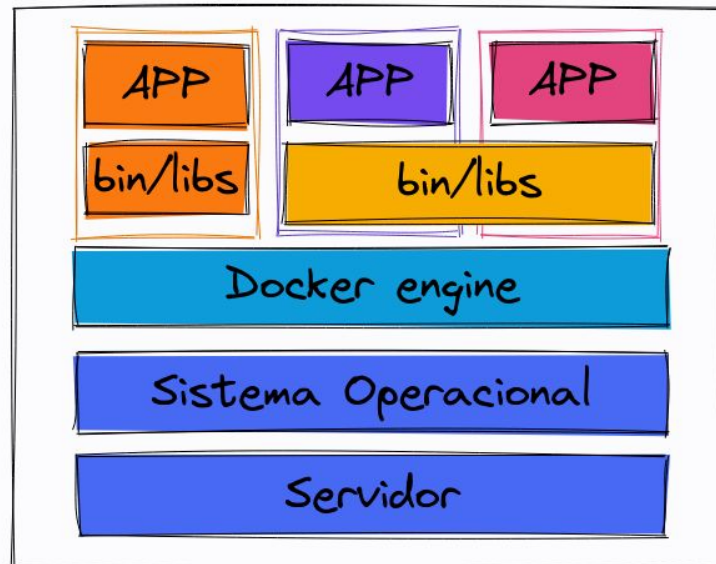


Containers X VMs

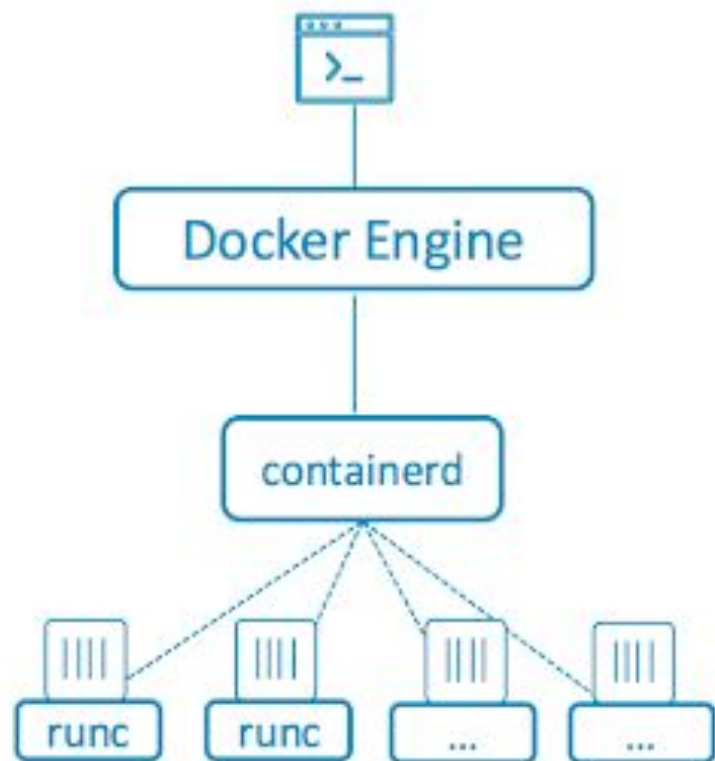
VIRTUALIZAÇÃO



CONTAINERS



FONTE: DO AUTOR



Same Docker UI and commands

User interacts with the Docker Engine

Engine communicates with containerd

containerd spins up runc or other OCI compliant runtime to run containers



O que é o Docker

- Plataforma Open Source;
- Empacotamento;
- Execução de containers;
- Docker Inc.
- 2013

História do Docker

Material de apoio



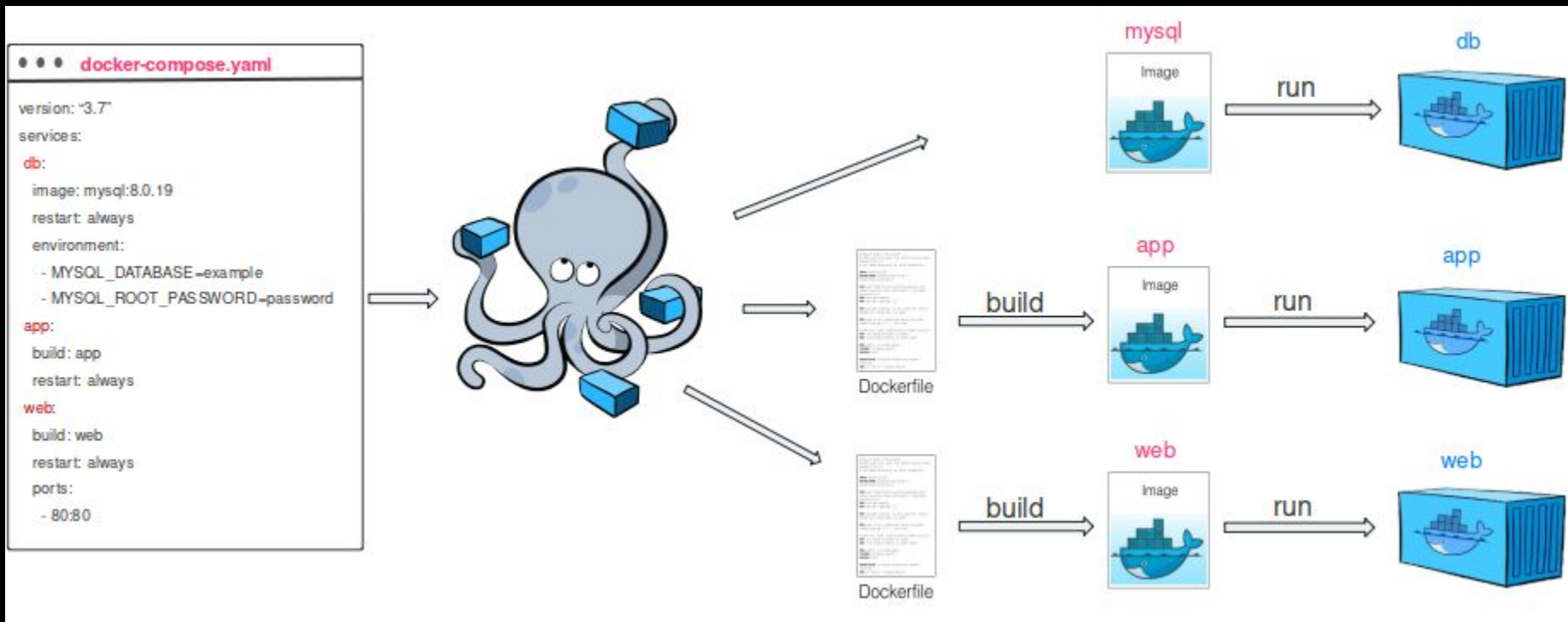


Benefícios

- Agilidade;
- Isolamento;
- Disponibilidade;
- Controle de versão;
- Reversibilidade;
- Microsserviços;
- Segurança.



O que é o Docker compose





Conexão SSH

Abrir um terminal de conexão:

- Terminal (Linux)
- PowerShell (Windows)
- PuTTY (Windows)

Executar o comando:

```
ssh univates@[ip]
```

Senha:





Instalando o docker

Seguir a documentação

- Ubuntu: <https://docs.docker.com/engine/install/ubuntu/>
- Windows: <https://docs.docker.com/desktop/install/windows-install/>





Instalando o docker compose

Mudança de docker-compose para docker compose no formato de plugin

- Como plugin: <https://docs.docker.com/compose/install/linux/>
- Standalone: <https://docs.docker.com/compose/install/other/>



Docker Hub

Repositório de imagens

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





Comandos úteis

Executar container: `docker run -p [host]:[porta externa]:[porta interna] -d`
`(background) --name [nome] [imagem:tag]`

[Material de apoio Docker Run](#)

Listar os containers: `docker ps`

Parar um container: `docker stop [nome]`

Iniciar o container parado: `docker start [nome]`

Remover um container: `docker rm [nome]`





Comandos úteis

Executar algum comando no container: `docker exec -it [container] [comando]`

Salvar alterações de um container: `docker commit [container] [imagem:tag]`

Listar imagens: `docker images`

Construir uma imagem: `docker build -t [tag] [o que buildar]`

Remover imagens: `docker image rm [imagem]`

Remover todas imagens em desuso: `docker image prune`





Na prática

```
docker run -d -p 8080:80 --name web nginx OU docker run --rm -d -p 8080:80 --name web nginx
```

```
docker exec -it web /bin/bash
```

```
vim /usr/share/nginx/html/index.html
```

```
apt update
```

```
apt install vim
```

```
vim /usr/share/nginx/html/index.html
```

[Material de apoio](#)





Na prática

```
docker commit web nginx:mateus
```

```
docker rm --force web
```

```
docker run --rm -d -p 8080:80 --name web nginx
```

```
docker run --rm -d -p 8081:80 --name webmateus nginx:mateus
```





Dockerfile

Arquivo usado para criar imagens

[Material de apoio](#)





Dockerfile

Clonar repositório <https://github.com/mateusrovedaa/minicurso-docker>

```
FROM nginx:latest
```

```
WORKDIR /usr/share/nginx/html/
```

```
COPY . .
```





Construindo nossa imagem

```
docker build -t portfolio:latest .
```

```
docker run --rm -p 8082:80 -d --name portfolio portfolio:latest
```

[http://\[ip\]:8082](http://[ip]:8082)





Rodando comandos em Dockerfile

```
FROM nginx:latest
```

```
WORKDIR /usr/share/nginx/html/
```

```
RUN apt update && apt install -y vim
```

```
RUN rm -rf /var/lib/apt/lists/*
```

```
COPY . .
```





Exemplo de Dockerfile (build de um código JavaScript)

```
FROM node:16

WORKDIR /usr/src/app

COPY package*.json ./

RUN yarn install

COPY . .

EXPOSE 3000

CMD ["yarn", "serve"]
```





Docker compose

Baixar imagens: `docker compose pull`

Executar o build: `docker compose build`

Executar os containers: `docker compose up -d`

Parar os containers: `docker compose down`

Executar um comando em um container: `docker compose exec -it [nome serviço]
[comando]`

Executar um comando em um novo container: `docker compose run --rm [nome
serviço] [comando]`





docker-compose.yml

```
version: '3.9'

services:
  nginx:
    image: nginx:latest
    container_name: nginx-container
    ports:
      - "8085:80"
```

[Material de apoio](#)





docker-compose.yml

```
version: '3.9'

services:
  nginx-build:
    build:
      dockerfile: Dockerfile
      context: .
    container_name: portfolio-container-build
    ports:
      - "8083:80"
```

Material de apoio





docker-compose.yml

```
docker compose up -d [serviço]
```

```
docker compose run --rm nginx-build bash
```





Docker compose com volume

```
nginx-volume:
  image: nginx:latest
  container_name: portfolio-container-volume
  ports:
    - "8084:80"
  volumes:
    - ./usr/share/nginx/html/
```





Ambiente com docker

<https://gitlab.com/mateusrovedaa/wordpress-docker/>





Extras

Trabalhar com redes diferentes para isolamento e comunicação entre containers

<https://medium.com/dockerbr/docker-trabalhando-com-network-64d0bf66263f>





Extras

Adicionar ao .bashrc para remover todos os containers

```
# Kill and remove all containers

function docker-clean {

    echo 'Killing...'

    docker kill $(docker ps -q)

    echo 'Removing...'

    docker rm $(docker ps -qa)

}
```





Extras

<https://github.com/aztecweb/aztlan>

