

O que é o Docker?



O Docker é uma plataforma open source que facilita a criação e administração de ambientes. Com ele podemos empacotar uma aplicação ou ambiente dentro de um **CONTAINER**.

Desenvolvido em Go



What is Go?

*Go is a programming language designed by Google to help solve **Google's problems**.*

And **Google** has big problems!

INSTALAÇÃO



Linux

- Bem simples;
- Tem tudo no site oficial;
- Inclusive o hub...



macOS





Enterprise Edition

Community Edition

 **Windows Server**



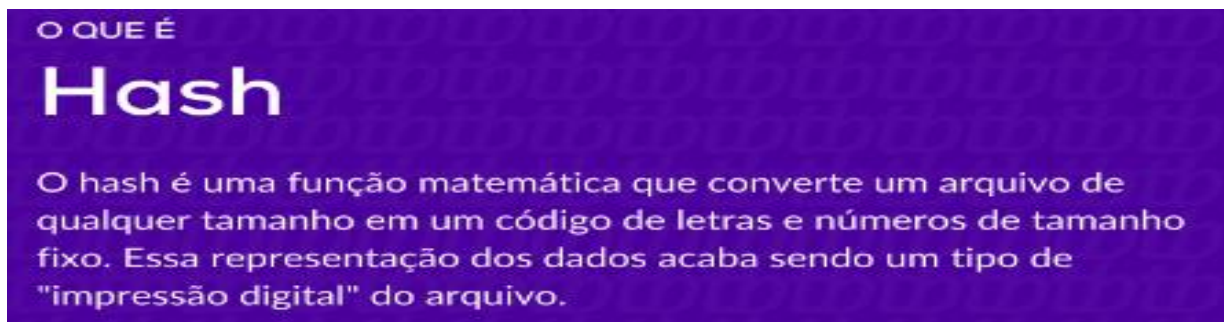
Mudou a maneira de versionar.

CE atualizado mensalmente, EE atualizado a cada 4 meses

sudo docker version

O que é hash -> É uma impressão digital(método criptográfico) de um arquivo qualquer.

Não sendo possível fazer o caminho inverso(de um hash extrair o conteúdo)



sudo docker run -d alpine podemos definir a versão com :versão

sudo docker container ls -a

sudo docker ps -a

sudo docker rm <id> -f

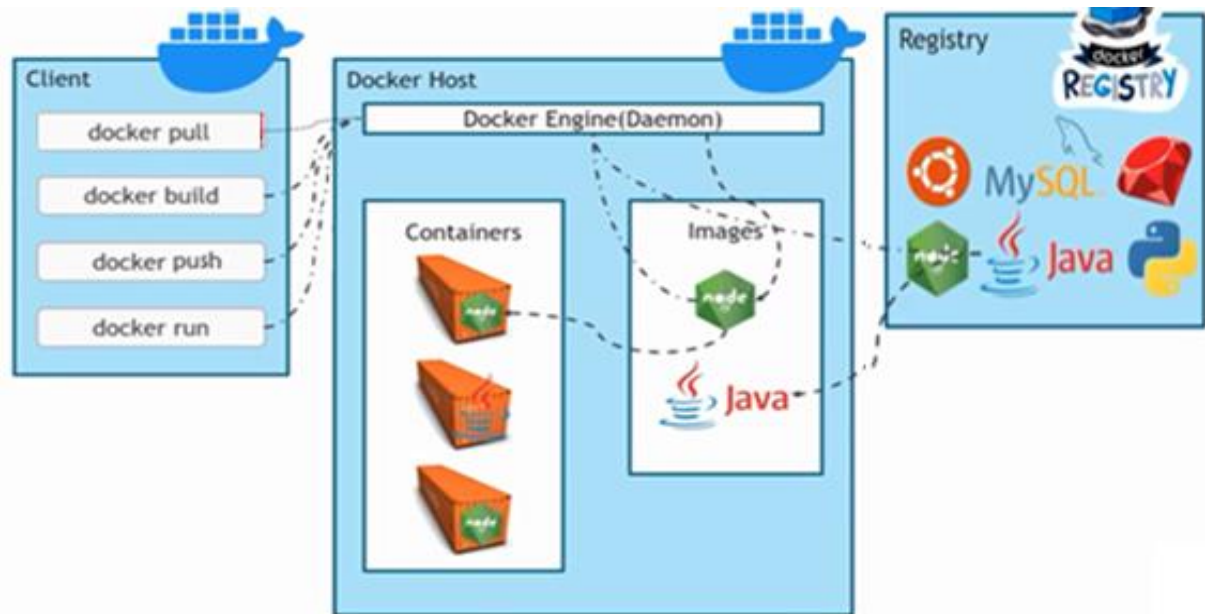
O que é bash -> GNU **Bash** ou simplesmente **Bash** é um acrônimo de Bourne Again **Shell** um interpretador de comandos, um entre os diversos tradutores entre o usuário e o sistema operacional conhecidos como **shell**.



Acrônimo	Definição
GNU	Gnu's Not Unix
GNU	Government of National Unity

sudo docker run -d nginx

sudo docker exec -it <id> bash -> para sair exit



Comando pull:

```
sudo docker images
```

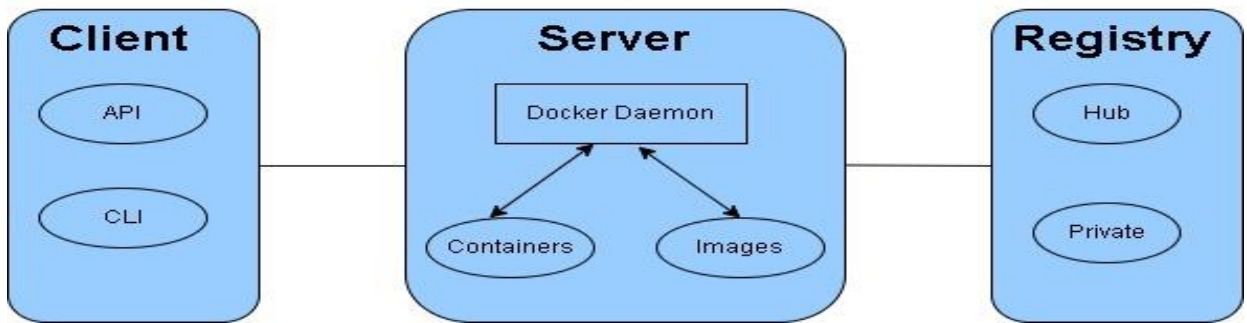
```
sudo docker pull alpine
```

```
sudo docker images
```

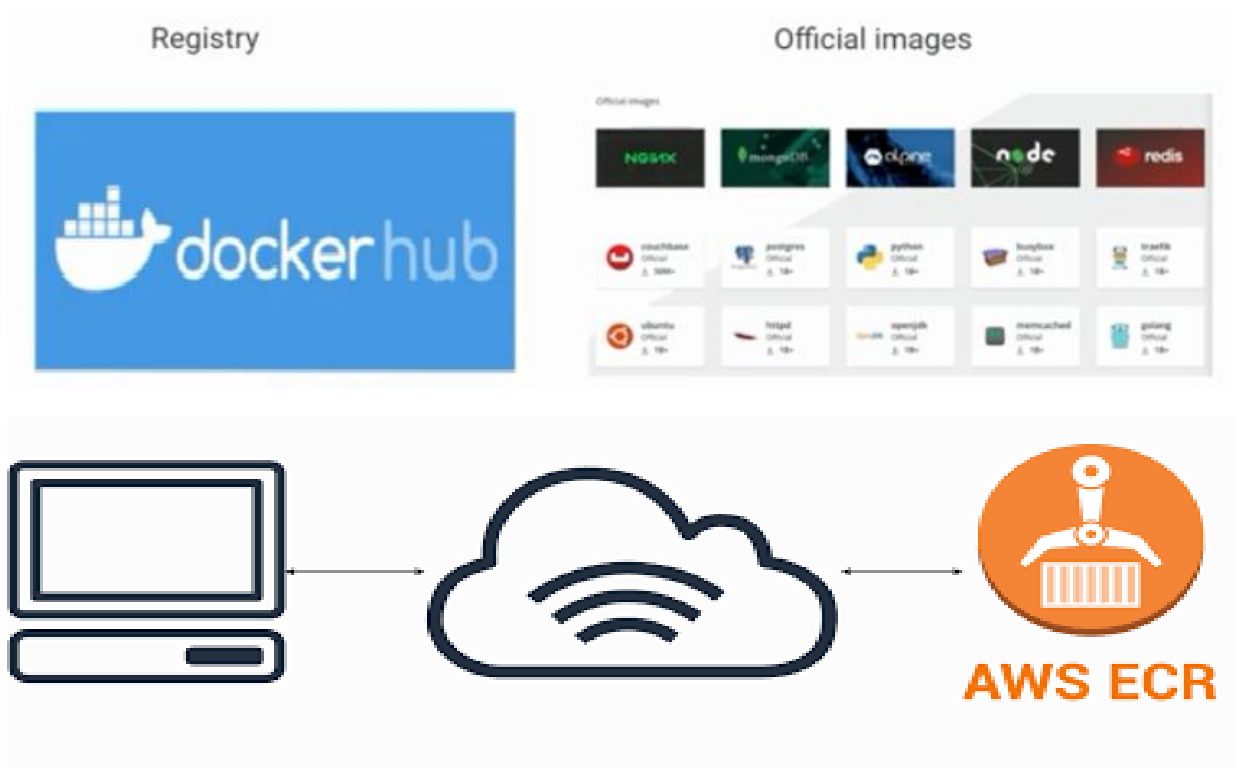
```
sudo docker rmi <id>
```

```
sudo docker image prune
```

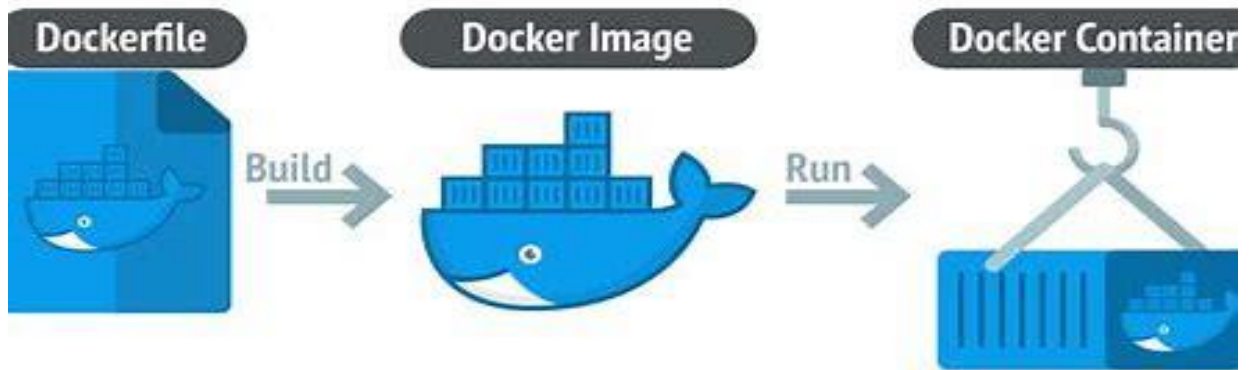
```
sudo docker images
```



Docker Hub: Um repositório com mais de 250 mil imagens diferentes para os seus containers.



Comando build:



Dockerfile

Texto plano

Sintaxe Simples

Instruções de como
construir uma imagem

```
# syntax=docker/dockerfile:1
FROM python:3.7-alpine
WORKDIR /code
ENV FLASK_APP=app.py
ENV FLASK_RUN_HOST=0.0.0.0
RUN apk add --no-cache gcc musl-dev linux-headers
COPY requirements.txt requirements.txt
RUN pip install -r requirements.txt
EXPOSE 5000
COPY . .
CMD ["flask", "run"]
```

Comando docker build:

```
mkdir novo
```

```
cd novo
```

```
sudo apt install vim
```

```
sudo vim teste.txt / sudo nano teste.txt
```

```
sudo vim Dockerfile / sudo nano Dockerfile
```

```
FROM alpine
```

```
VOLUME /meuvolume/
```

```
COPY ./teste.txt /opt
```

```
Salvar arquivo(ctrl x -> s)
```

```
sudo docker images
```

```
sudo docker image build --t meu-alpine:1.9 .
```

A imagem origem está no Dockerfile

```
sudo docker images
```

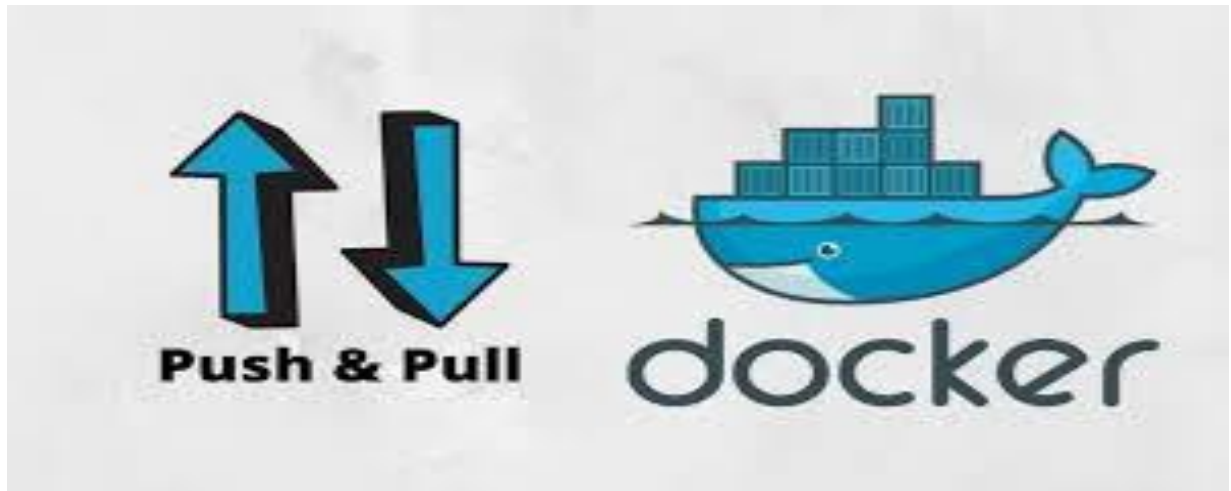
```
sudo docker run -it <id>
```

```
ls -> cd opt -> ls
```

```
cd .. -> ls
```

```
exit
```

```
exit
```



Comando push:

```
sudo docker search meu-alp
```

```
sudo docker login
```

usuário:

senha:

```
sudo docker images
```

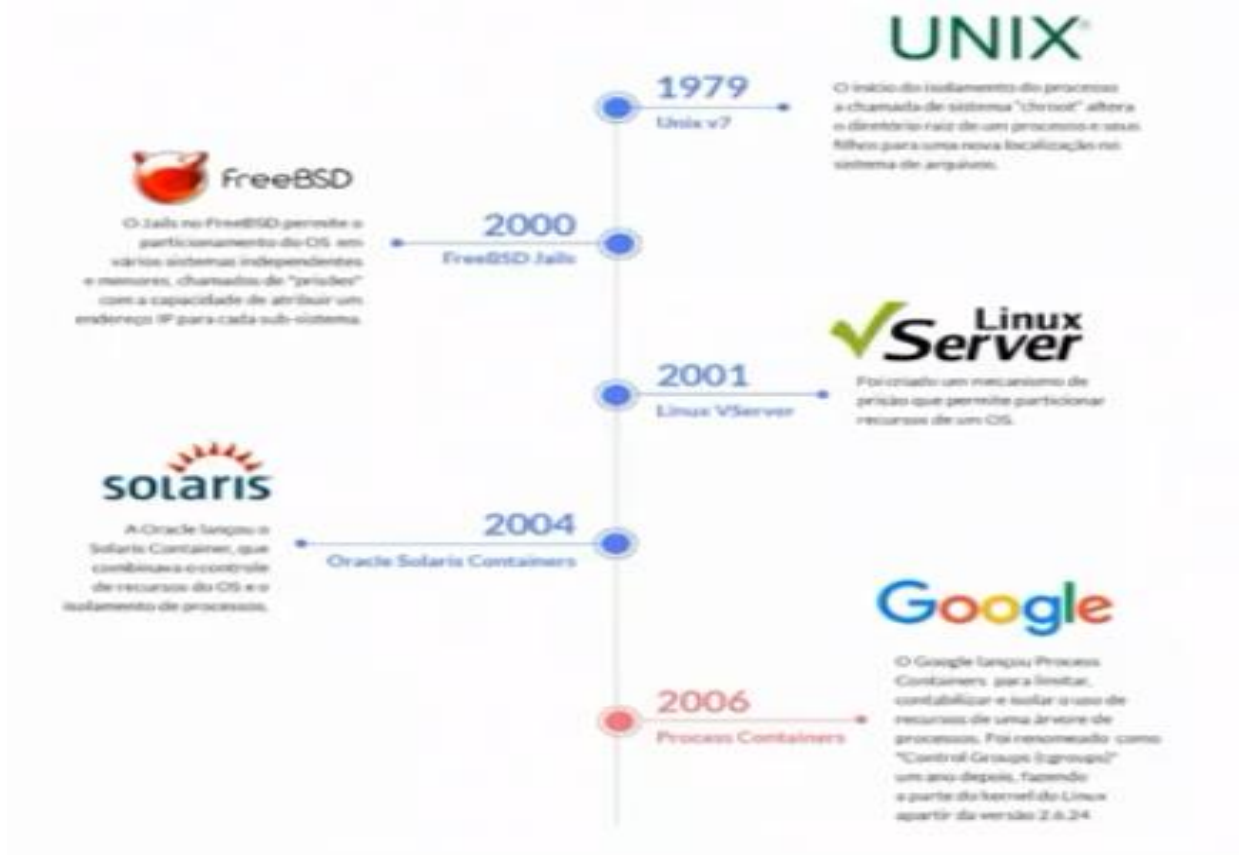
```
sudo docker tag meu-alp:1.0 jorgegil1905/meu-alp
```

```
sudo docker push jorgegil1905/meu-alp
```

Link <https://hub.docker.com/u/jorgegil1905>

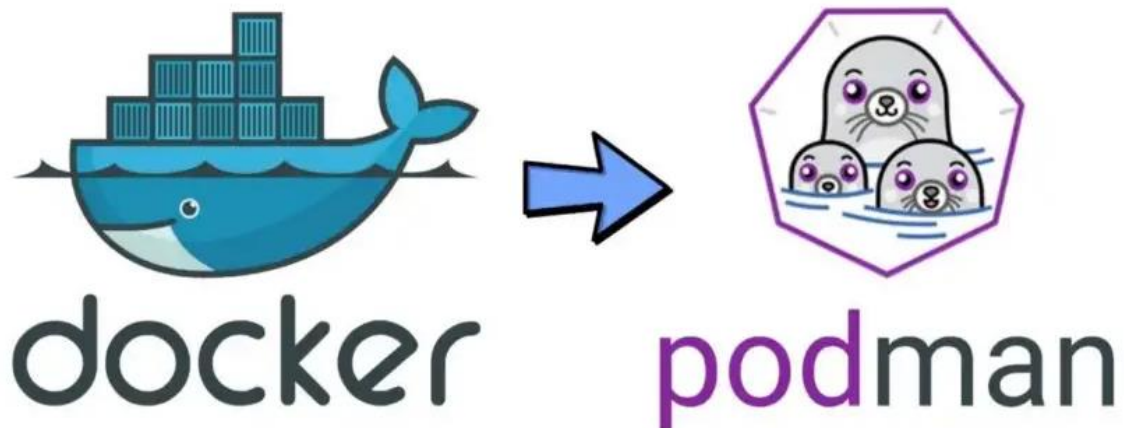
```
sudo docker search meu-alp
```

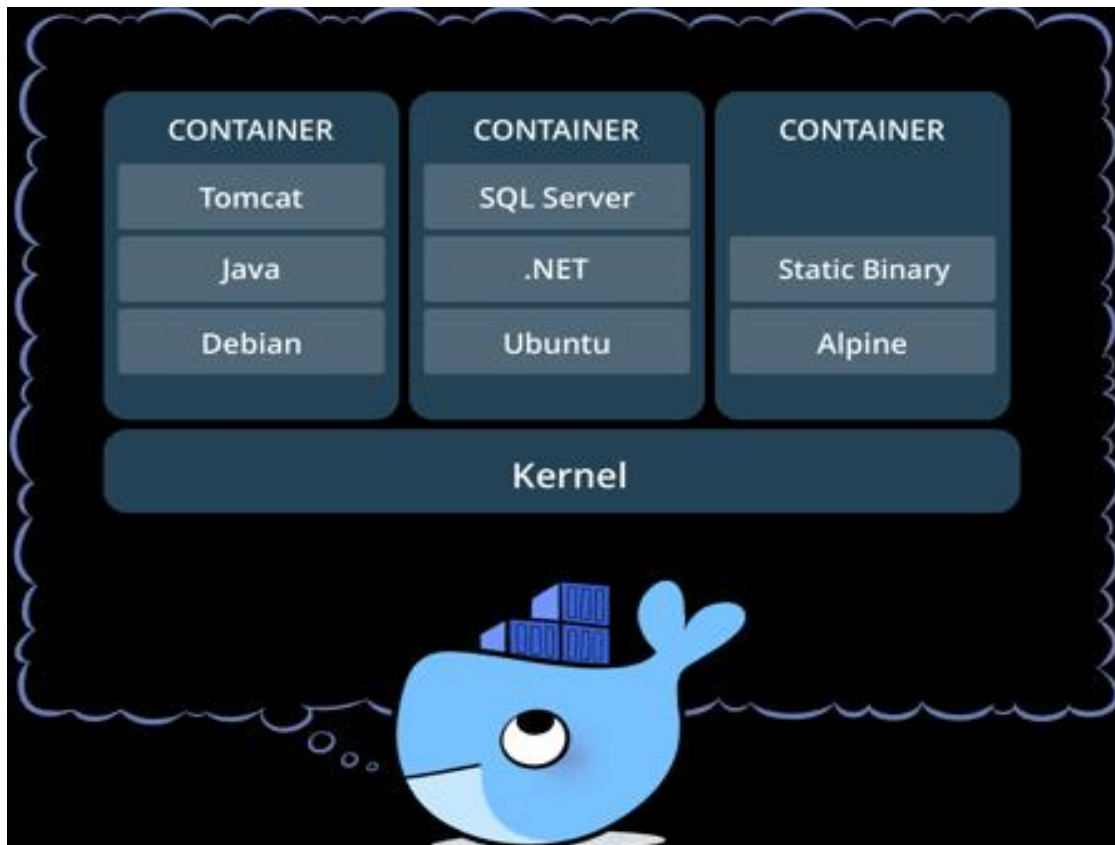

Quando foi criado o conceito de Containers.



Os containers não são exclusivos do Docker.

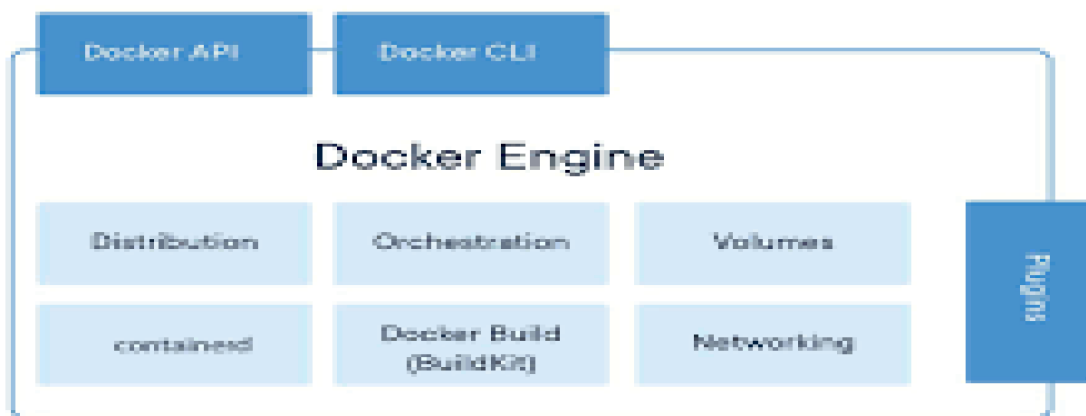
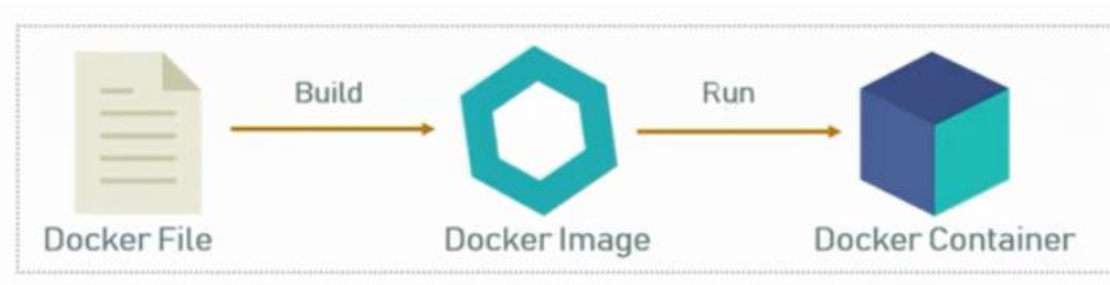
Docker vs Podman





Comando run:

```
sudo docker run -it alpine <exit>/sudo docker run -d alpine <exit>
```



```
sudo docker container ls -a
```

```
sudo docker run -it imagem
```

```
sudo docker stats
```

```
sudo docker system df
```

```
ctrl c
```

```
sudo docker container prune
```

```
sudo docker rm container <id> -f
```

Docker

Alternativa de virtualização em que o kernel da máquina hospedeira é compartilhado com a máquina virtualizada ou o software em operação.



[Play with Docker \(play-with-docker.com\)](https://play-with-docker.com)

4 horas para trabalhar, sistema operacional alpine.

No AWS temos o ECS e EC2 para rodar o Docker.



Uma curiosidade é que o alpine não contém o bash.

Não se deve instalar o Docker em máquina virtual.

Máquina Virtual (VM / Virtual Machine)

- Abstração do hardware de computadores que permite a uma única máquina física agir como se fosse várias.
- Criadas sobre uma camada de software denominada Hypervisor.
- Possibilitam múltiplos sistemas operacionais na mesma máquina física com execução simultânea.



Máquinas virtuais locais

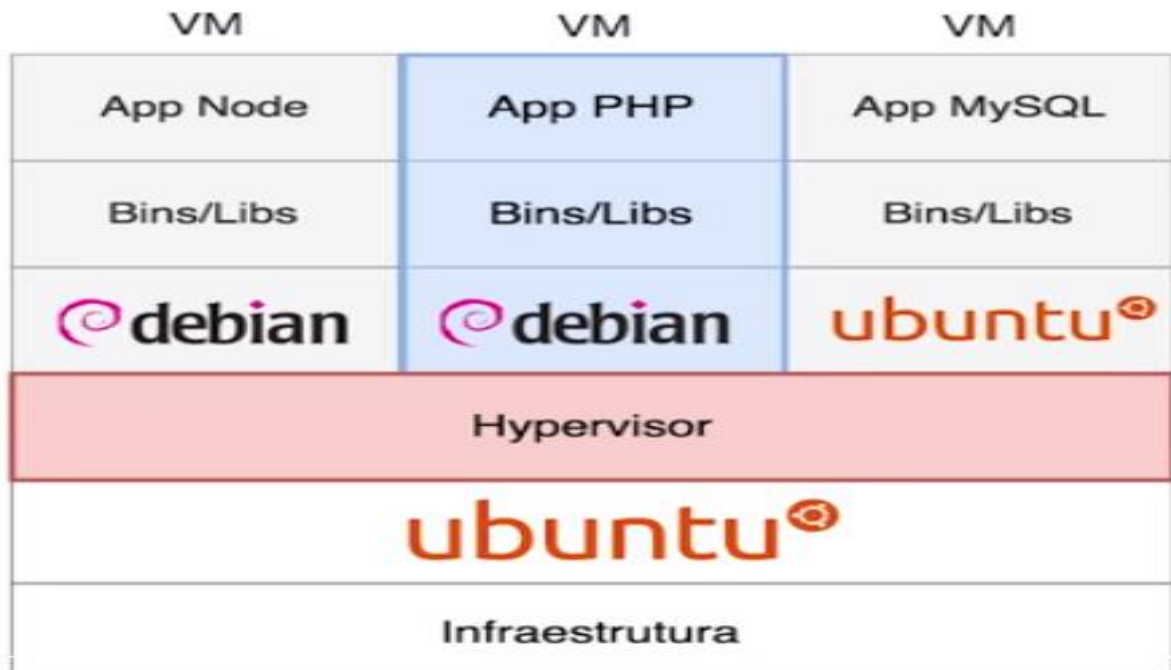
VMware Workstation Player

Execute facilmente vários sistemas operacionais como máquinas virtuais em seu computador Windows ou Linux com o VMware Workstation Player.

Hypervisor

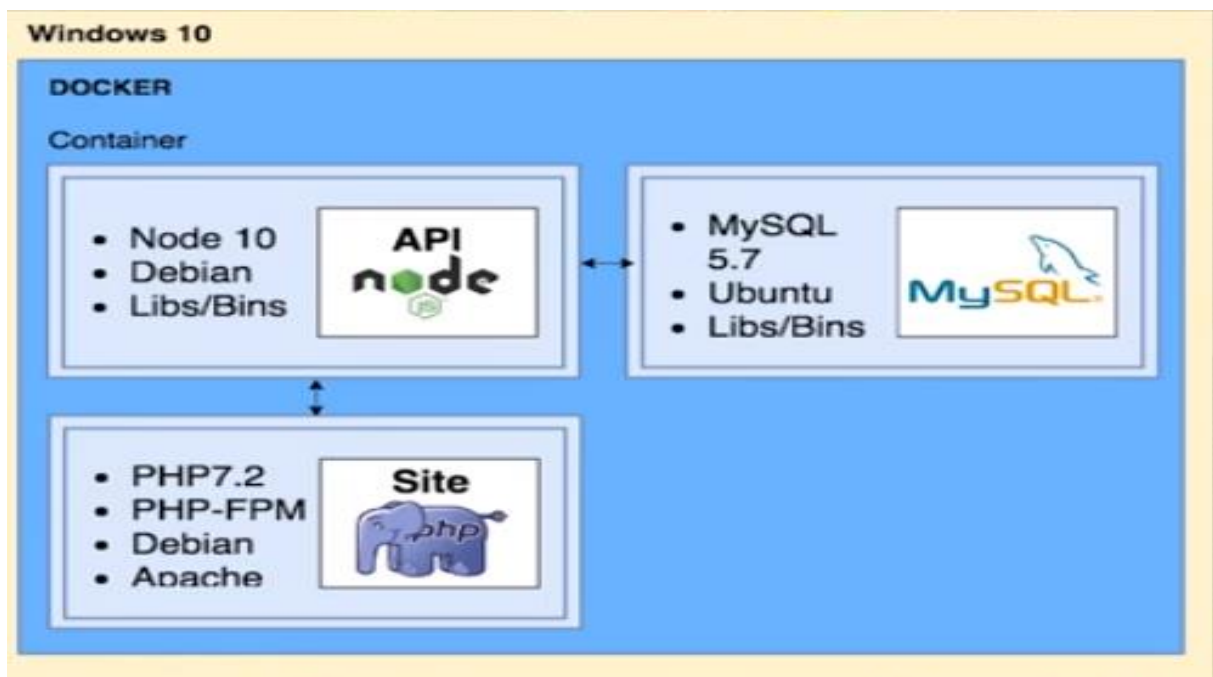
Software utilizado para gerenciamento das máquinas virtuais.

- Também conhecidos como **VMM - Virtual Machine Manager**
- Dois tipos principais: **Tipo 1** e **Tipo 2**.
- Desenvolvidos a partir do início dos anos 70 para rodar em mainframes

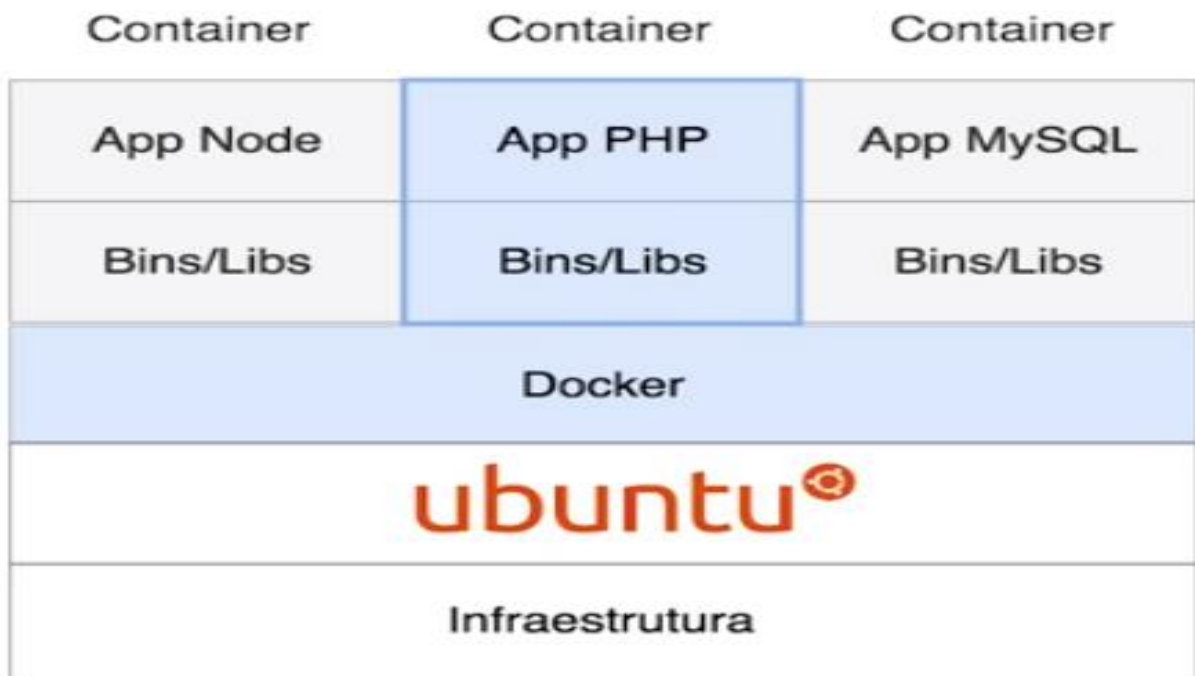
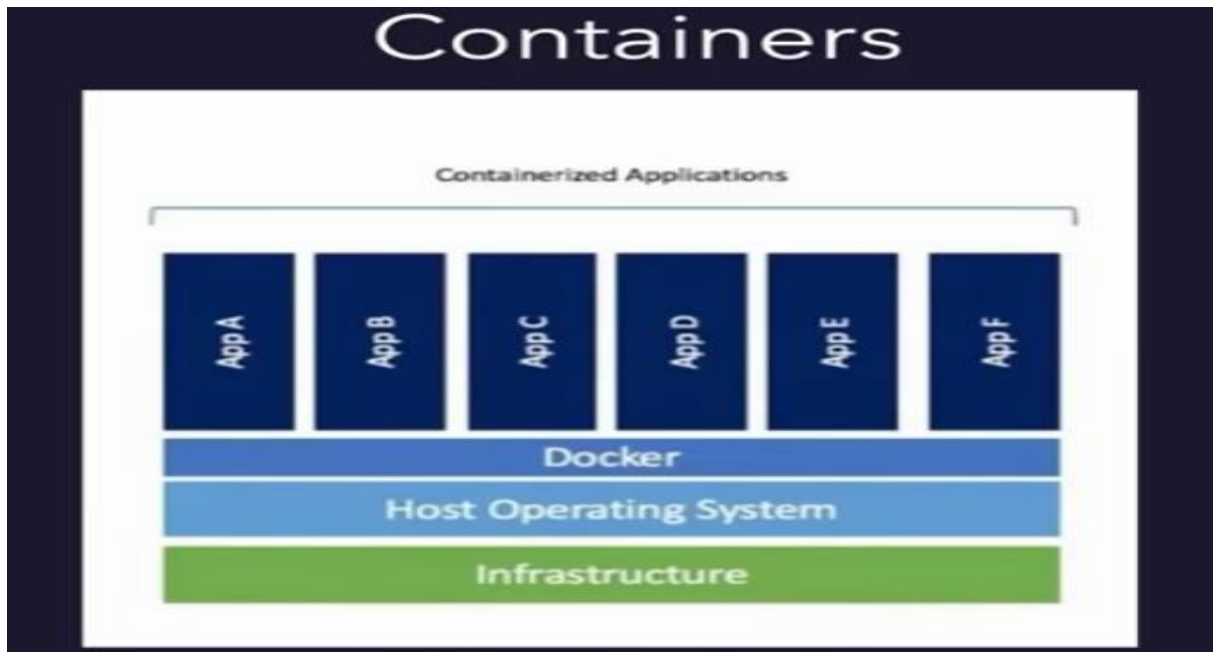


Requisitos para rodar o WSL 2

-  Windows 10 Home ou Pro.
-  Windows 10 versão >- 19.03.
-  4GB RAM.
-  Virtual Machine Platform habilitado.

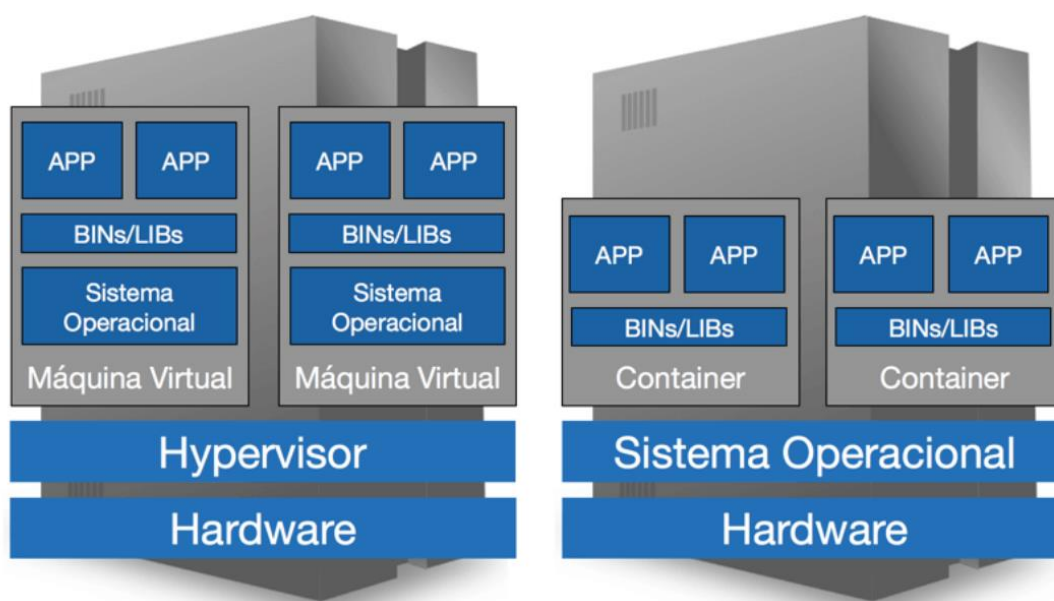


Os containers são o isolamento de processos(virtualizados) através de um Container RunTime.



CONTAINERS

- Embalagem padronizada para software e dependências.
- Isolamento de aplicativos uns dos outros.
- Compartilha o mesmo kernel do sistema operacional.
- Funciona para todas as principais distribuições Linux e também no Windows.



Hipervisor

```
sudo docker rmi id
```

```
sudo docker run --rm -it debian
```

comandos não funcionam aqui.

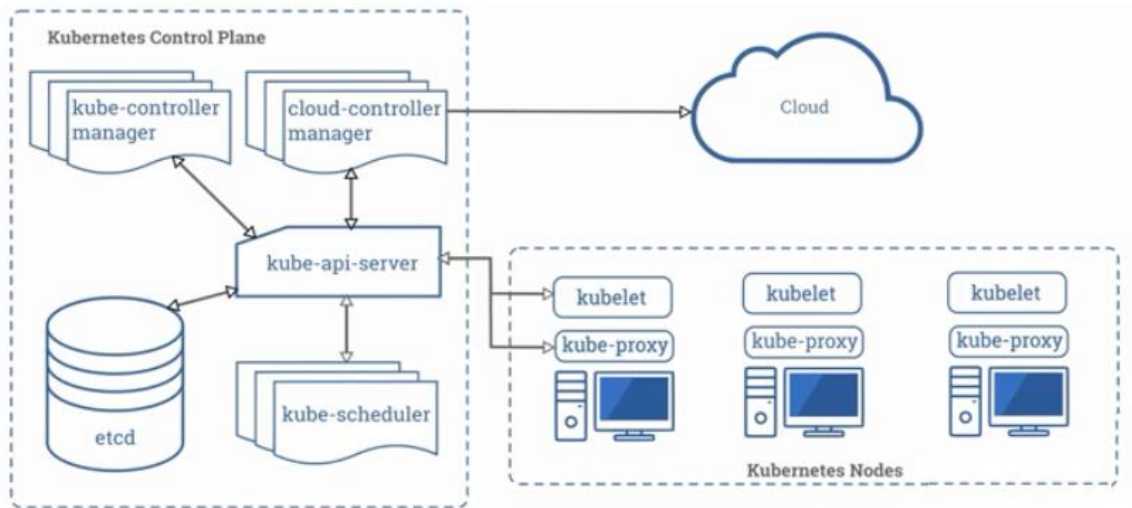
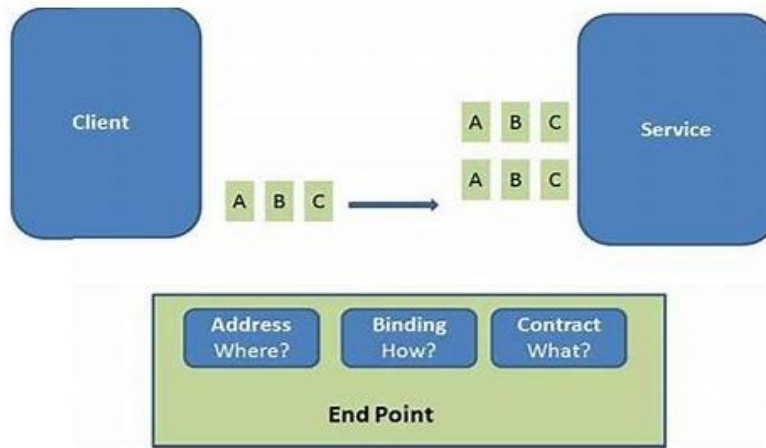
```
cat /proc/version
```

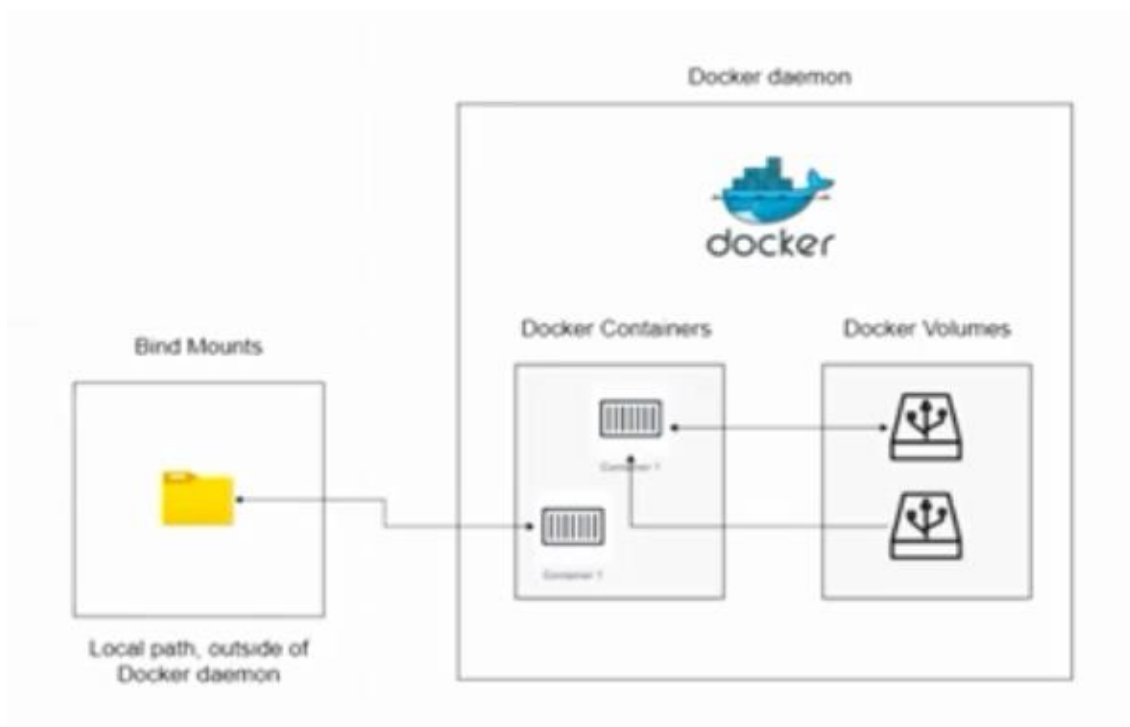
```
ctrl c
```

```
sudo cat /proc/version
```

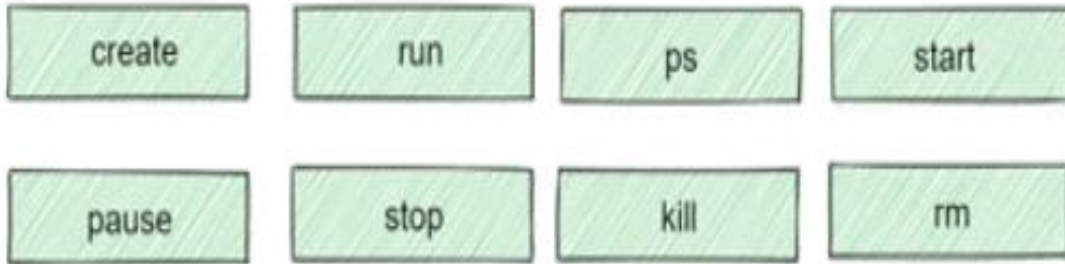
Fazer o bind das portas







Gerenciando containers



Docker hub



Docker Compose: uma ferramenta para definir e executar aplicativos Docker a partir de vários contêineres.

Docker Compose:

Criação de um ambiente com:

- SQL Server 2017
- SQL Server 2019 CTP 3.1
- PostgreSQL + pgAdmin 4
- MySQL + Adminer



Docker Compose e ciclo de vida da aplicação

- Iniciar, parar e reconstruir serviços
- Ver o status dos serviços em execução
- Transmitir a saída de log dos serviços em execução
- Executar um comando único em um serviço

```
version: "3.9"  
services:  
  web:  
    build: .  
    ports:  
      - "8000:5000"  
  redis:  
    image: "redis:alpine"
```

Tarefas usuais com Docker Compose

- Divida seu aplicativo em serviços
- Pull ou construção de imagens
- Configurar variáveis de ambiente
- Configurar rede
- Configurar volumes
- Build e execução

Exemplos

```
docker-compose up -d  
docker-compose down  
docker-compose start  
docker-compose stop  
docker-compose build  
docker-compose logs  
docker-compose events  
docker-compose exec service command
```

Docker-compose.yml

sudo docker -compose ps

sudo docker -compose ls

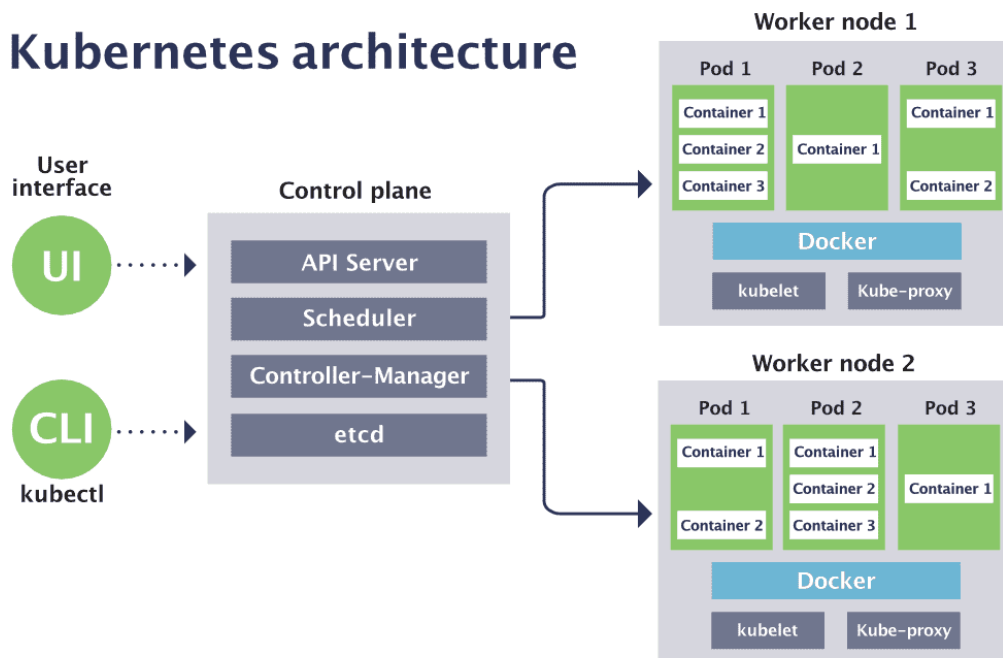


kubernetes

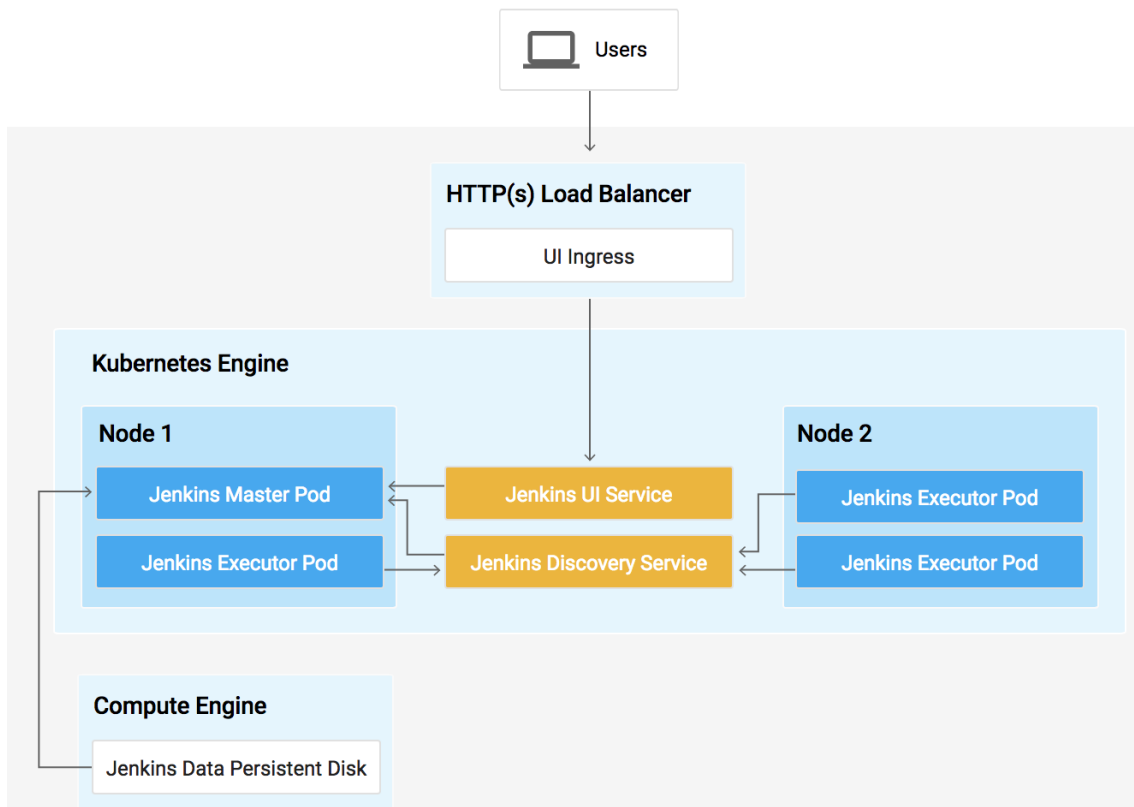
Kubernetes

1 É um orquestrador de containers lançado em 2015 para dar mais escalabilidade e performance.

Kubernetes architecture

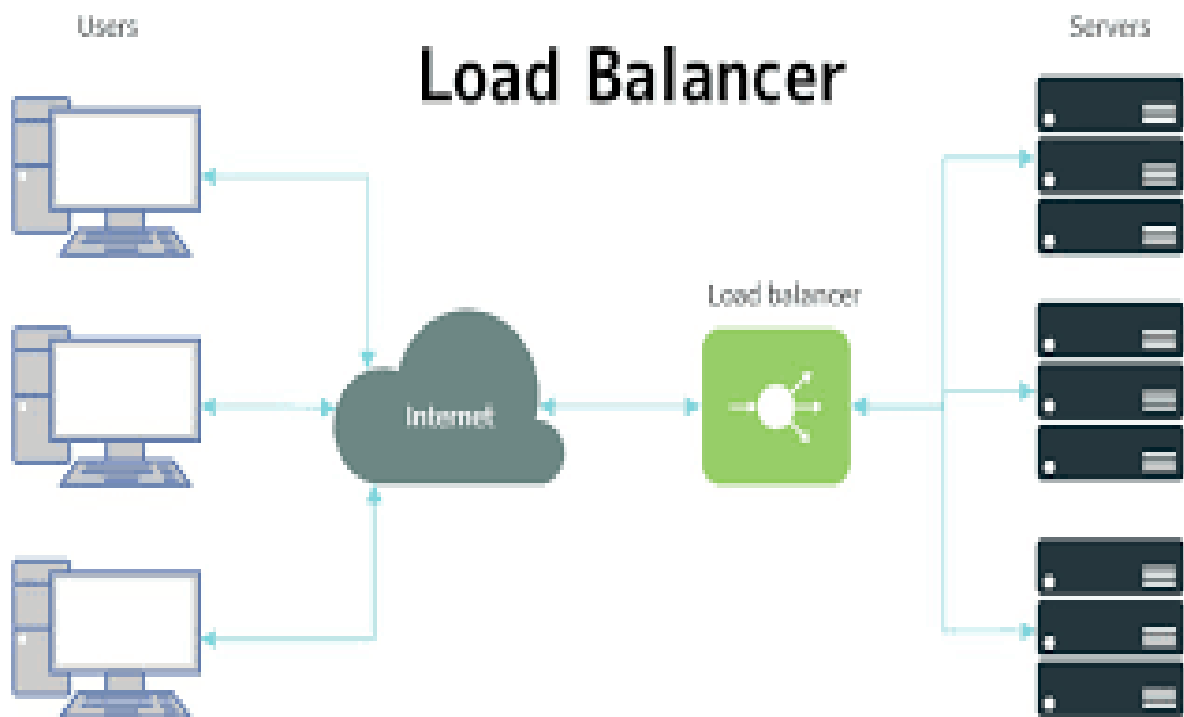


XML	JSON	YAML
<pre><Servers> <Server> <name>Server1</name> <owner>John</owner> <created>123456</created> <status>active</status> </Server> </Servers></pre>	<pre>{ Servers: [{ name: Server1, owner: John, created: 123456, status: active }] }</pre>	<pre>Servers: - name: Server1 owner: John created: 123456 status: active</pre>



O que é um cluster

- Conjunto de servidores agrupados com intenção de ganho de desempenho, disponibilidade, ou facilidade no gerenciamento.
- Início em 1994 pela NASA.
- Normalmente um cluster é composto por máquinas convencionais ligadas em uma rede de alto desempenho e fornecendo a abstração ao usuário de uma única máquina.



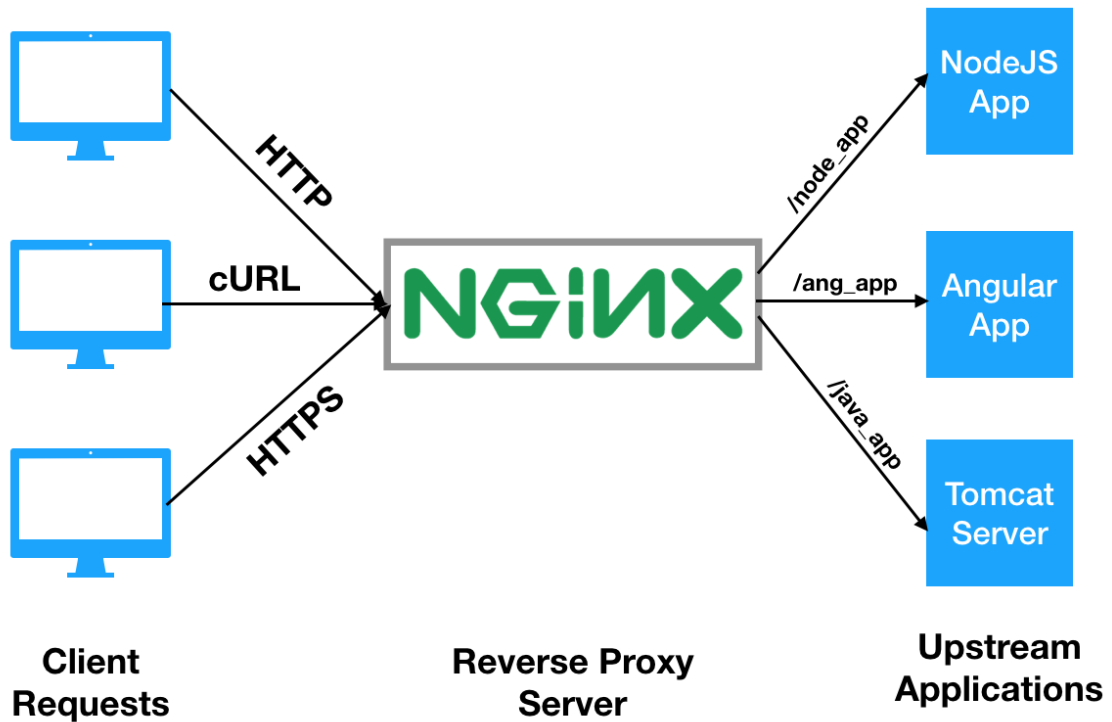


Maior concorrente do Apache e com estrutura assíncrona orientada a eventos que suporta milhares de conexões simultâneas, escrito em C.

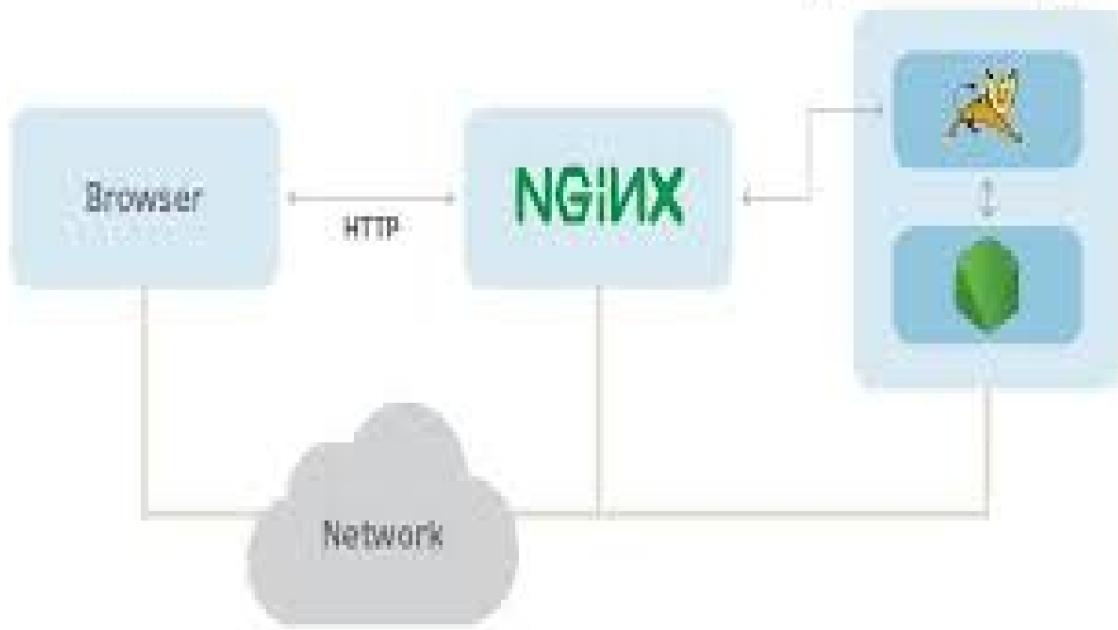
C10k. Escala para milhares de servidores

Efetua um load balance para redirecionamento de carga e redirecionamento de endereço.

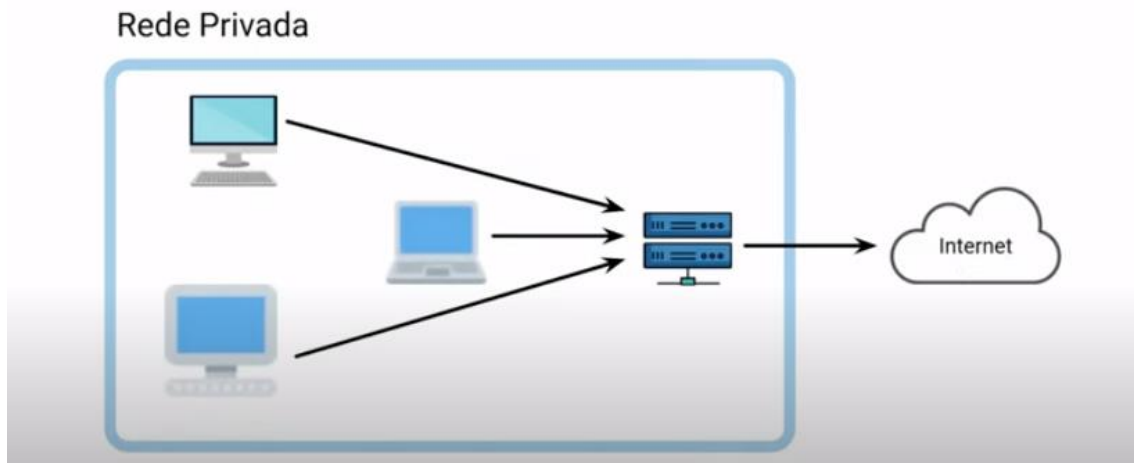




Nginx Reverse Proxy Setup

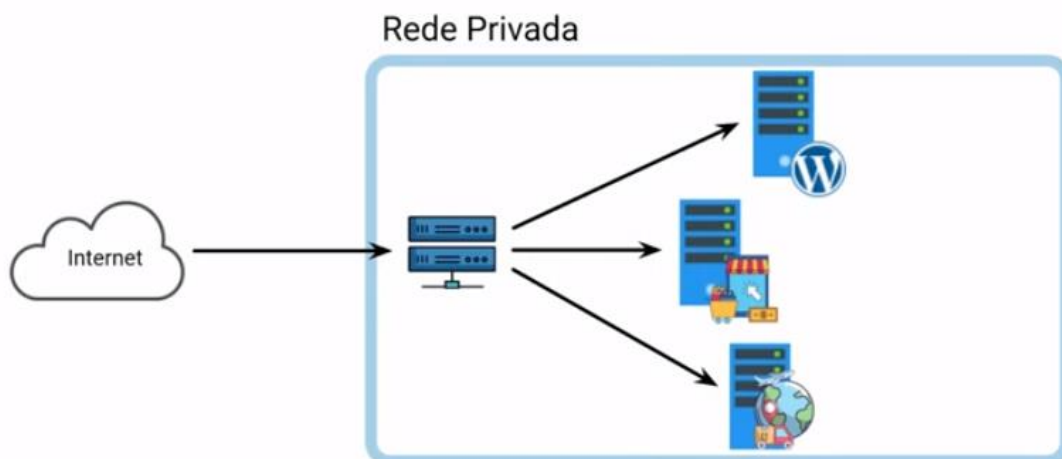


Proxy



Proxy reverso é um único ponto de acesso (centraliza o tráfego).
Escuta os IP's nas portas configuradas.

Proxy Reverso



history

!numero escolhido

history -c -> Limpa o history

sudo docker run -dti -p 8080:80 nginx

curl localhost:8080/index.html

sudo docker inspect id

cat /proc/version

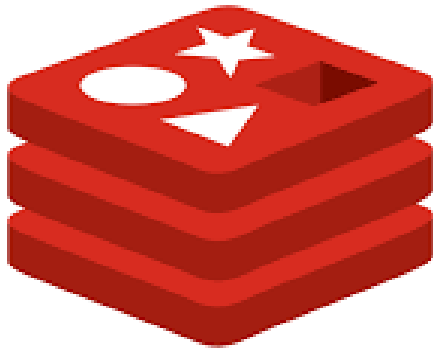
sudo docker stop id

cat /proc/version

sudo docker ps -a

ctrl q ou exit

sudo cat /proc/version



redis



ElastiCache Redis



Memorystore Redis



Azure Cache Redis

