

APLICAÇÕES ESCALONÁVEIS



Bruno Fidelis

DOCKER

- O que é o Docker ?

“Software containerization platform”

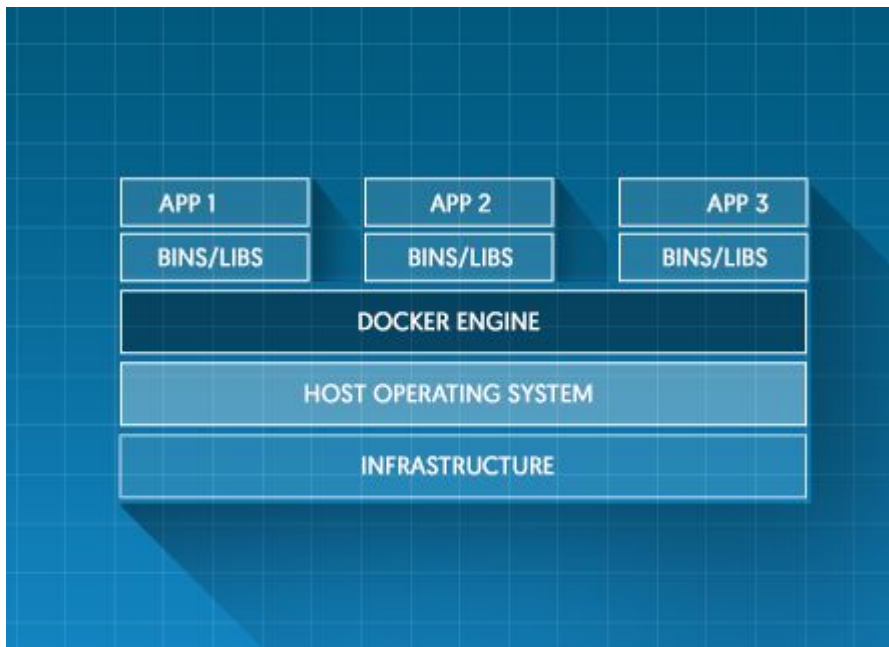
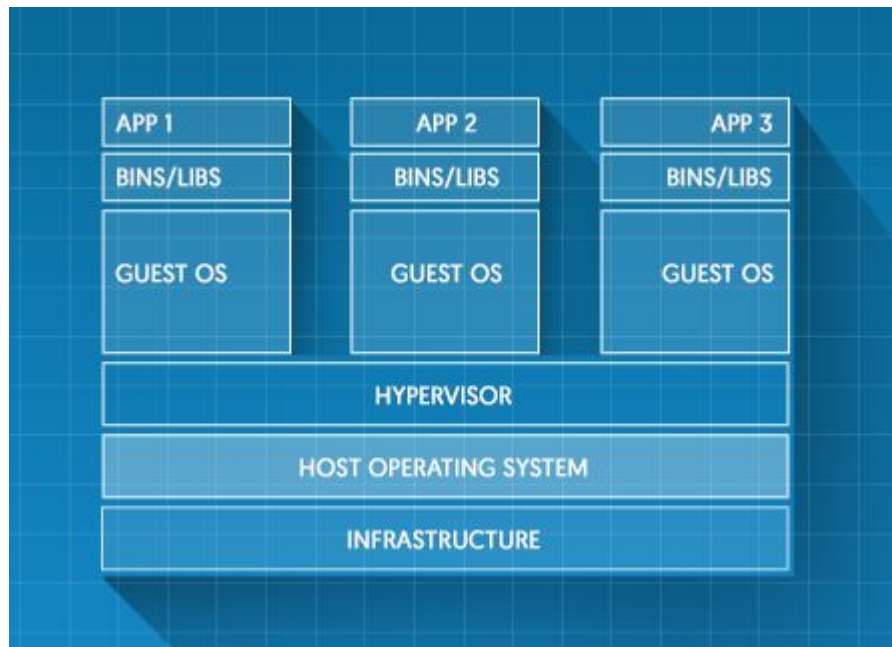
- Compartilha o mesmo kernel

“Lightweight”

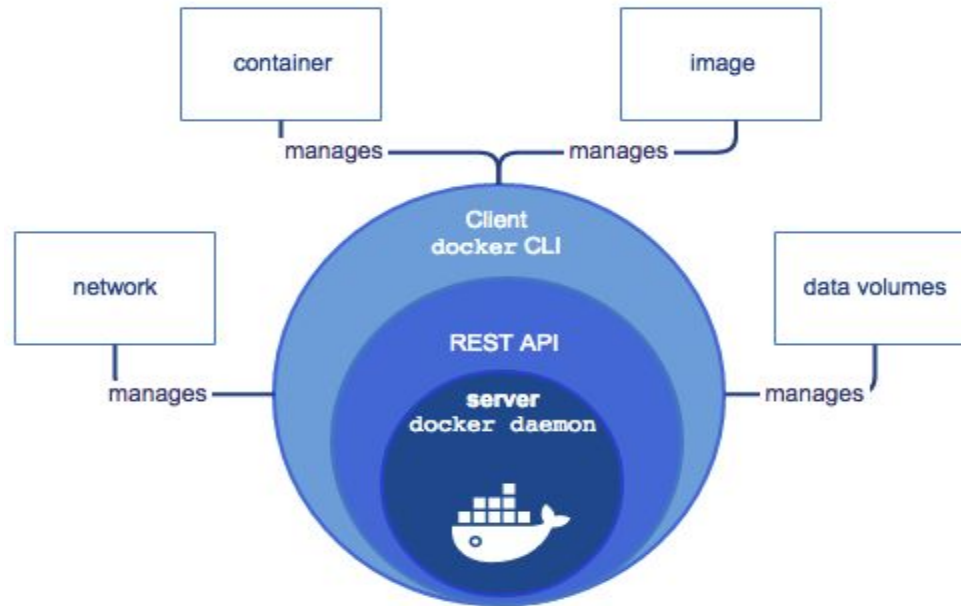
- Utiliza UnionFS

“Images are constructed from layered filesystems”

DOCKER x VM



DOCKER



DOCKER

- Images

“Template de leitura” - Exemplo Apache com sua aplicação

- Containers

Containers é o que é criado a partir das images Um diretório

- Registries

Contém imagens publicas e privadas, onde é possível realizar ações como pull, push

DOCKER

- O que acontece quando executamos um container ?

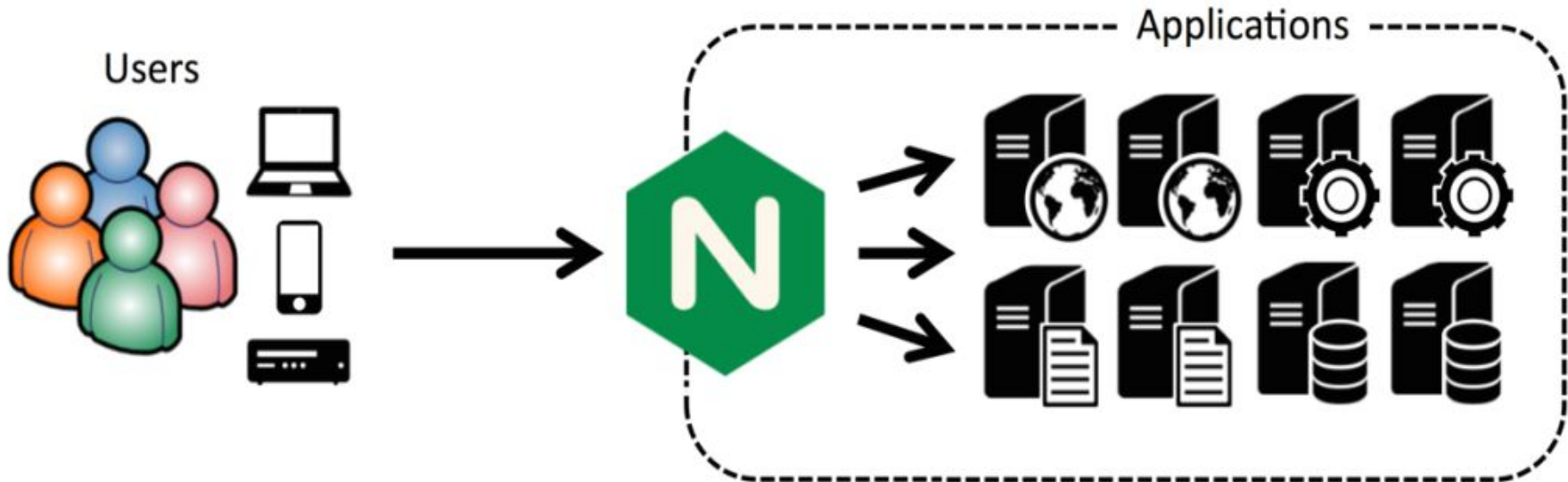
```
$ docker run -it ubuntu /bin/bash
```

1. Caso não encontre a imagem, faz pull no dockerhub
2. Cria o container
3. Cria interface de rede para comunicação com o host
4. Cria um ip
5. Executa o processo que você especificou
6. Captura o output da aplicação

DOCKER HUB

- Plataforma para distribuir e compartilhar imagens
- Imagens oficiais
- Ambientes “versionados”
- Automação de Builds
- WebHooks

ESTRUTURA DA APLICAÇÃO



COMO É FEITO HOJE

- Cluster

“A single instance of Node.js runs in a single thread. To take advantage of multi-core systems the user will sometimes want to launch a cluster of Node.js processes to handle the load.”

<https://nodejs.org/api/cluster.html>

```
const cluster = require('cluster');
const http = require('http');
const numCPUs = require('os').cpus().length;

if (cluster.isMaster) {
  // Fork workers.
  for (var i = 0; i < numCPUs; i++) {
    cluster.fork();
  }
  cluster.on('exit', (worker, code, signal) => {
    console.log(`worker ${worker.process.pid} died`);
  });
} else {
  // Workers can share any TCP connection
  // In this case it is an HTTP server
  http.createServer((req, res) => {
    res.writeHead(200);
    res.end('hello world\n');
  }).listen(8000);
}
```

COMO É FEITO HOJE

- PM2 (Process Manager)

O PM2 faz toda a lógica utilizada no cluster, para que não seja necessário fazer isso na aplicação.

```
[joni] ~/keymetrics/PM2 $ pm2 start app.js -i 0
```

```
[PM2] Spawning PM2 daemon
```

```
[PM2] PM2 Successfully daemonized
```

```
[PM2] Process app.js launched
```

App name	id	mode	pid	status	restart	uptime	memory	watching
app	0	cluster	4203	online	0	0s	19.008 MB	disabled
app	1	cluster	4212	online	0	0s	19.008 MB	disabled
app	2	cluster	4233	online	0	0s	19.012 MB	disabled
app	3	cluster	4254	online	0	0s	19.008 MB	disabled

```
Use `pm2 show <id|name>` to get more details about an app
```

DOCKER HUB

- `jwilder/docker-gen`

docker-gen is a file generator that renders templates using docker container meta-data.

Pode ser utilizado para:

- Centralizar logs
- Configuração de Proxy Reverso

DOCKER HUB

- `jwilder/nginx-proxy`

nginx-proxy sets up a container running nginx and docker-gen. docker-gen generates reverse proxy configs for nginx and reloads nginx when containers are started and stopped.

- Permite a configuração dinâmica

```
docker-gen -only-exposed -watch -notify "/etc/init.d/nginx reload" templates/nginx.  
tmpl /etc/nginx/sites-enabled/default
```

DOCKER

- Vantagens

Ambientes replicáveis

Ambiente como parte do software

Open Source

Leve

Elimina inconsistência de ambientes

OBRIGADO =)



@btfidelis



@btfidelis004