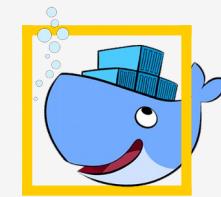


Começando com **Docker**



<Diogo Alves>



- Arquiteto de Soluções e
Líder na iniciativa DevOps
da empresa Cedro



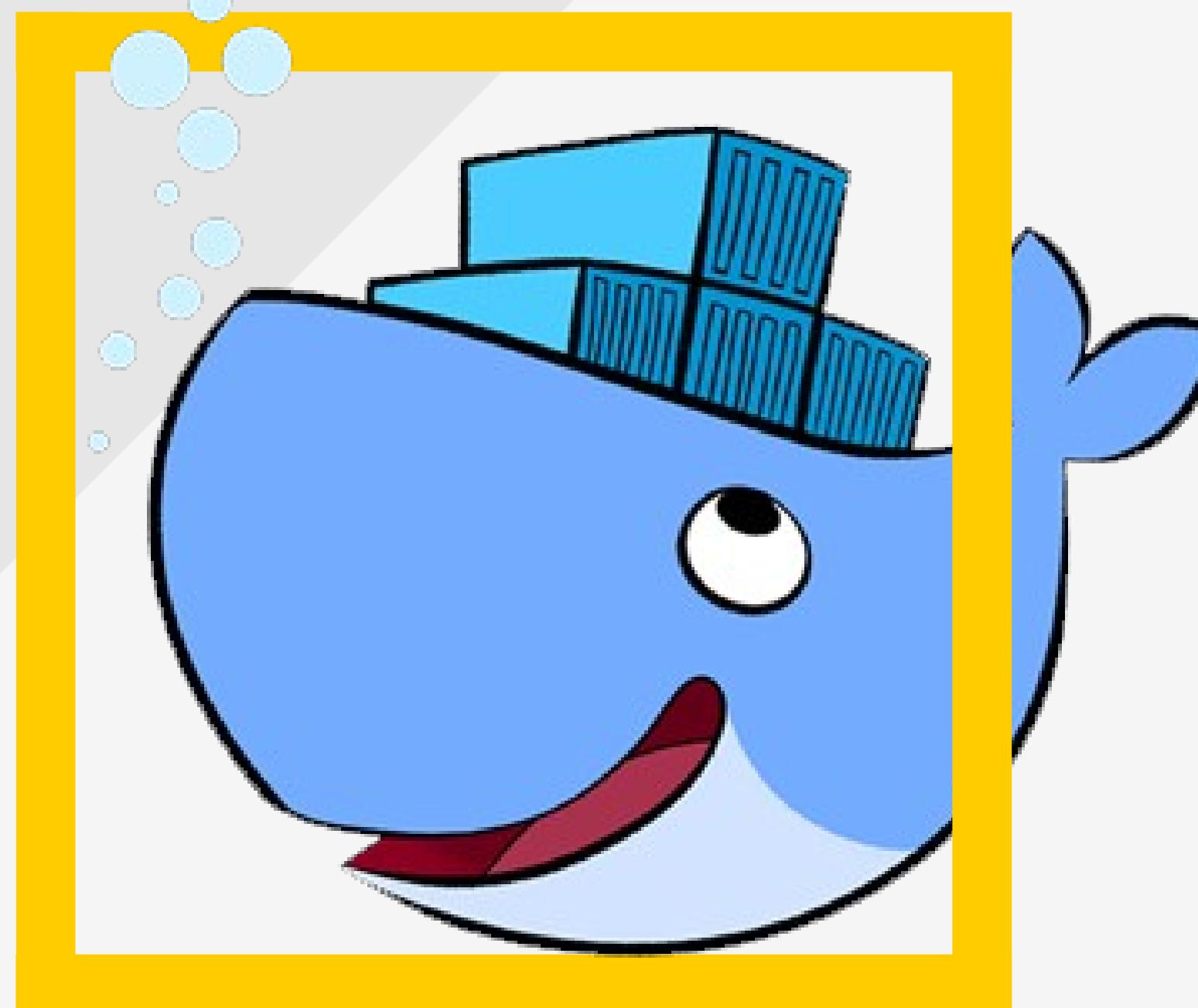
Formação:
- Sistemas de Informação



Pouco mais de 12 anos como:
- SysAdmin apaixonado por Linux
- Desenvolvedor
- Instrutor
- Entusiasta de Segurança da
Informação



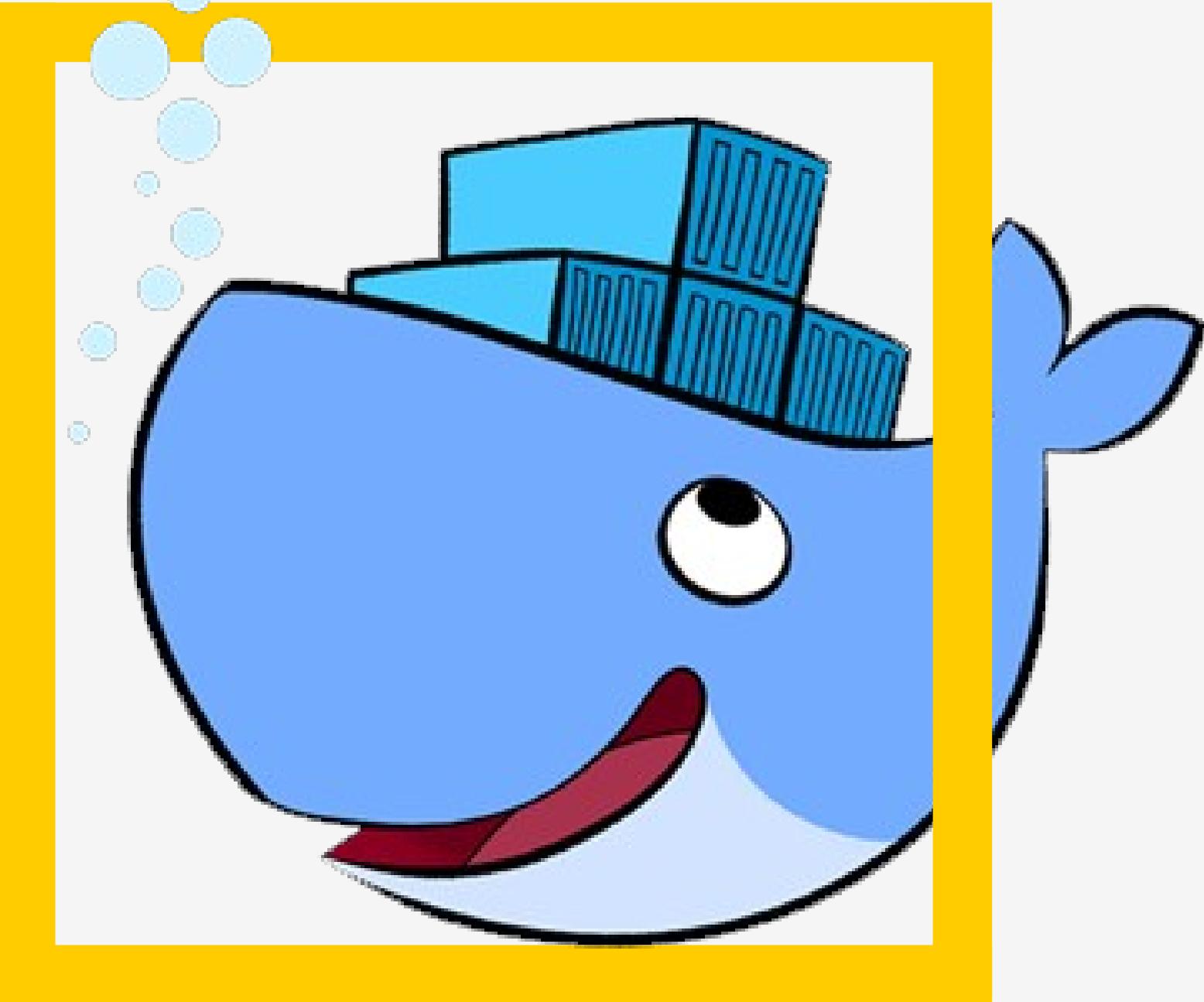
Algumas Certificações:
- CEH – Ethical Hacking
- ISO 27001
- Asterisk Administrator
- AWS



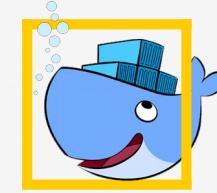
Começando com **Docker**

Agenda

- O que é Docker?
- Por que usar Docker?
- Docker vs. VM
- Preparando Ambiente
- Instalando Docker
- Image e Container
- Dockerfile
- Hand-ons



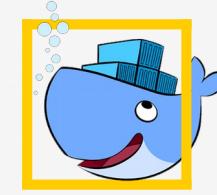
O que é
Docker?



<O que é Docker>

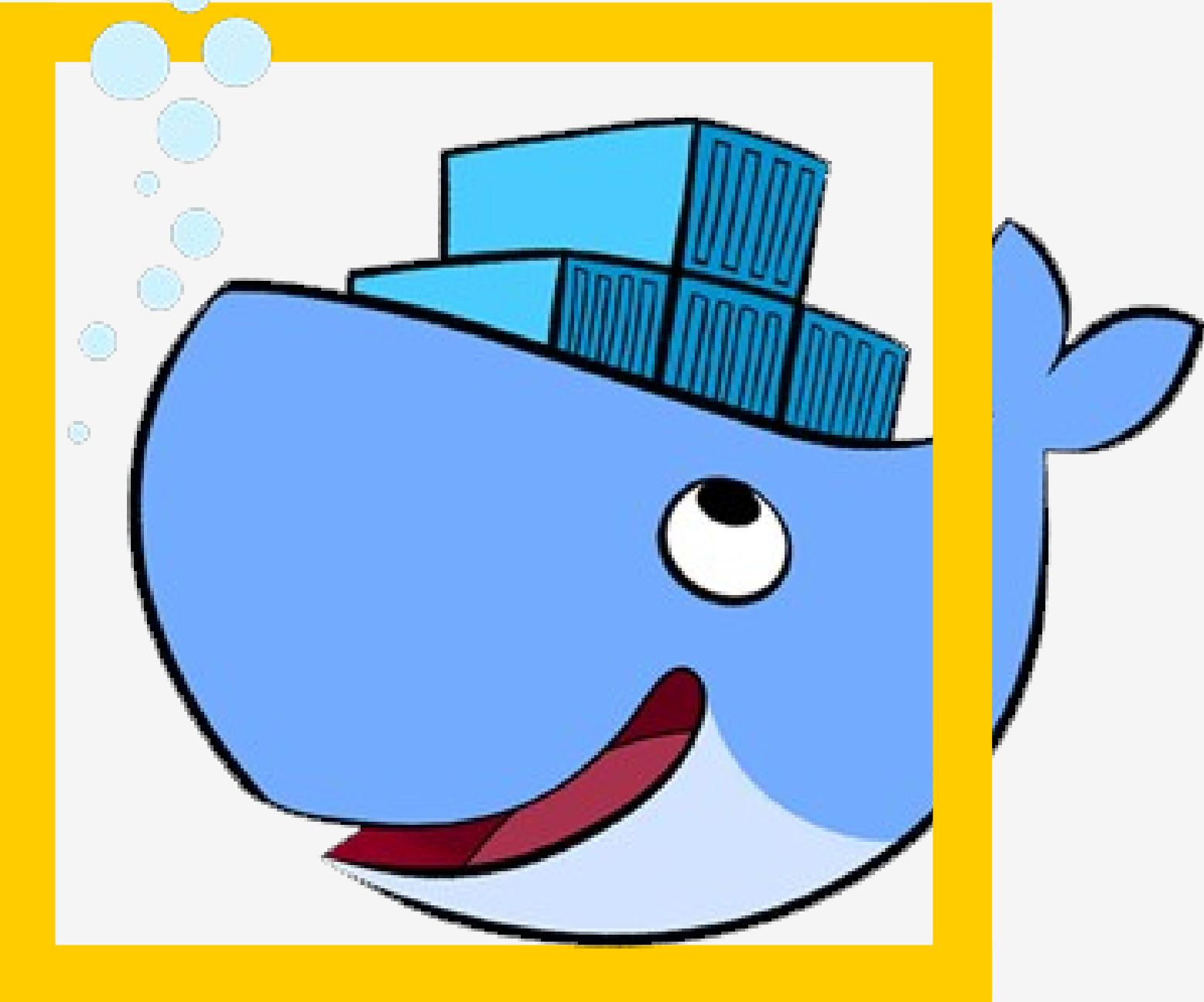
- ✓ Ferramenta Open Source
- ✓ Gerenciador de Container LXC(Linux Containers)
- ✓ Criado pela StartUP DOTCloud
- ✓ Disponibilizado para a comunidade com o nome de Docker em 2013 como uma engine de Deploy

"Em qualquer computador, em qualquer INFRA, em qualquer nívem..."

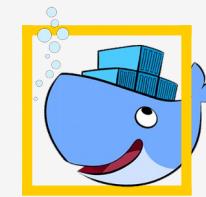


<O que é Docker>

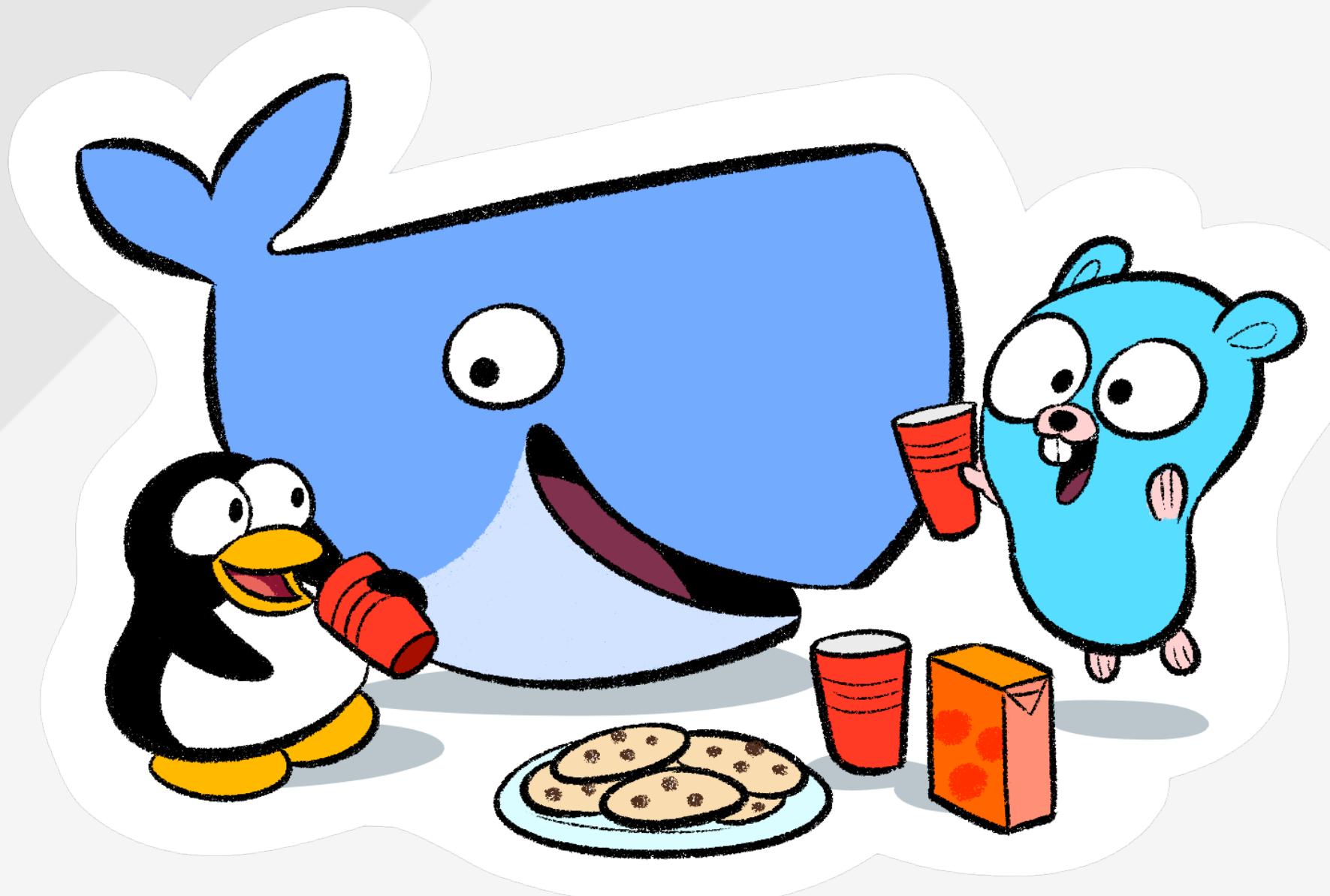
- ✓ Basicamente é um empacotador de ambientes
- ✓ Funciona Independemente do Hardware
- ✓ Não depende de frameworks ou linguagens específicas
- ✓ Facilidade no Deploy
- ✓ Escalabilidade de aplicações WEB, Banco de Dados e demais serviços de BackEnd
- ✓ Funciona de forma isolada da máquina Host
- ✓ Permite multiplas instancias utilizando o mesmo Kernel



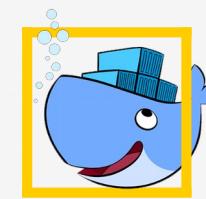
**Por que usar
Docker?**



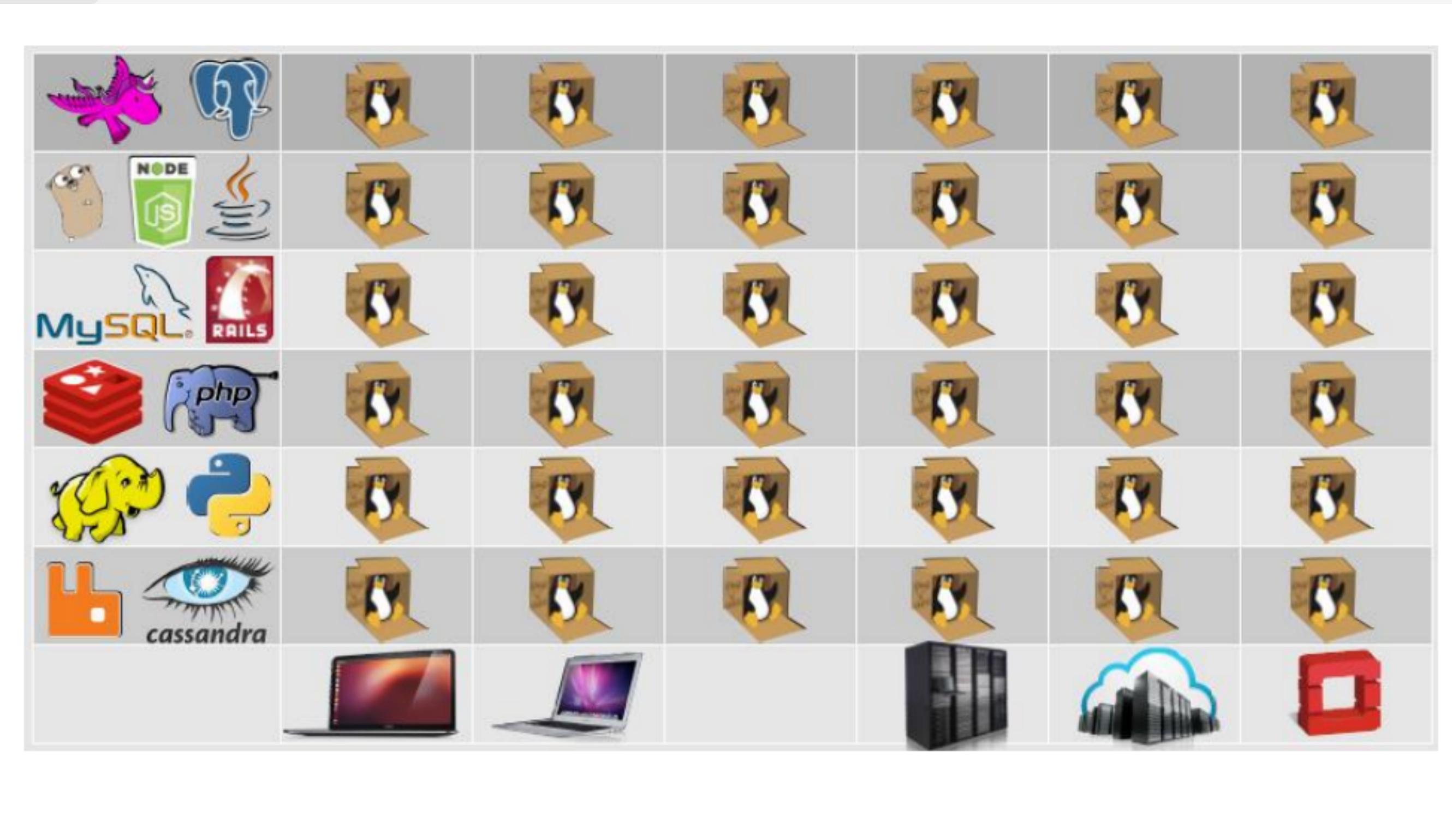
<Por que usar Docker?>

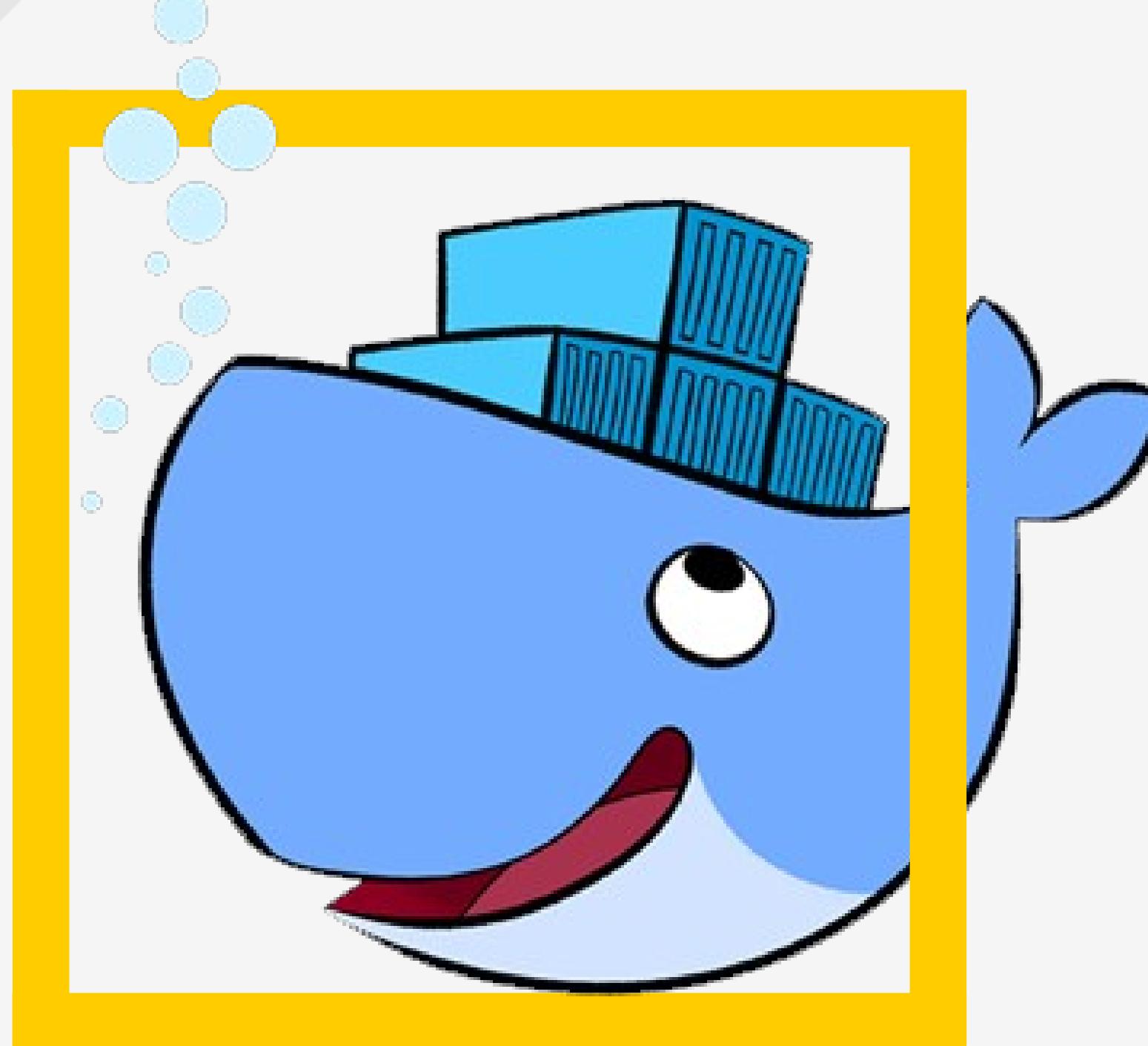


- ✓ Leve
- ✓ Eficiente
- ✓ Open Source
- ✓ Seguro
- ✓ Escalável
- ✓ Multi Serviços
- ✓ Kernel Linux
- ✓ Feito para a Nuvem



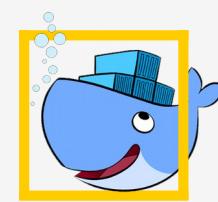
<Por que usar Docker?>



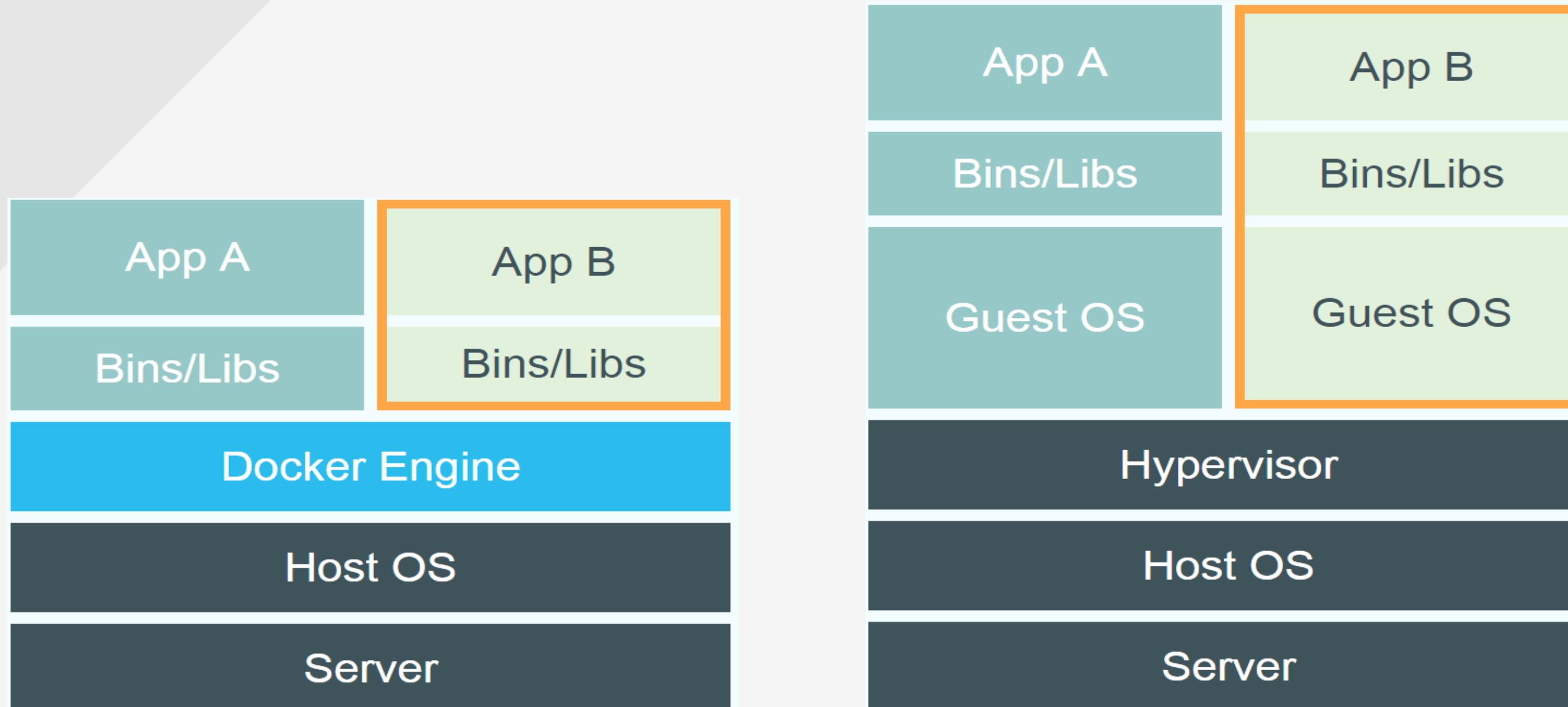


Docker

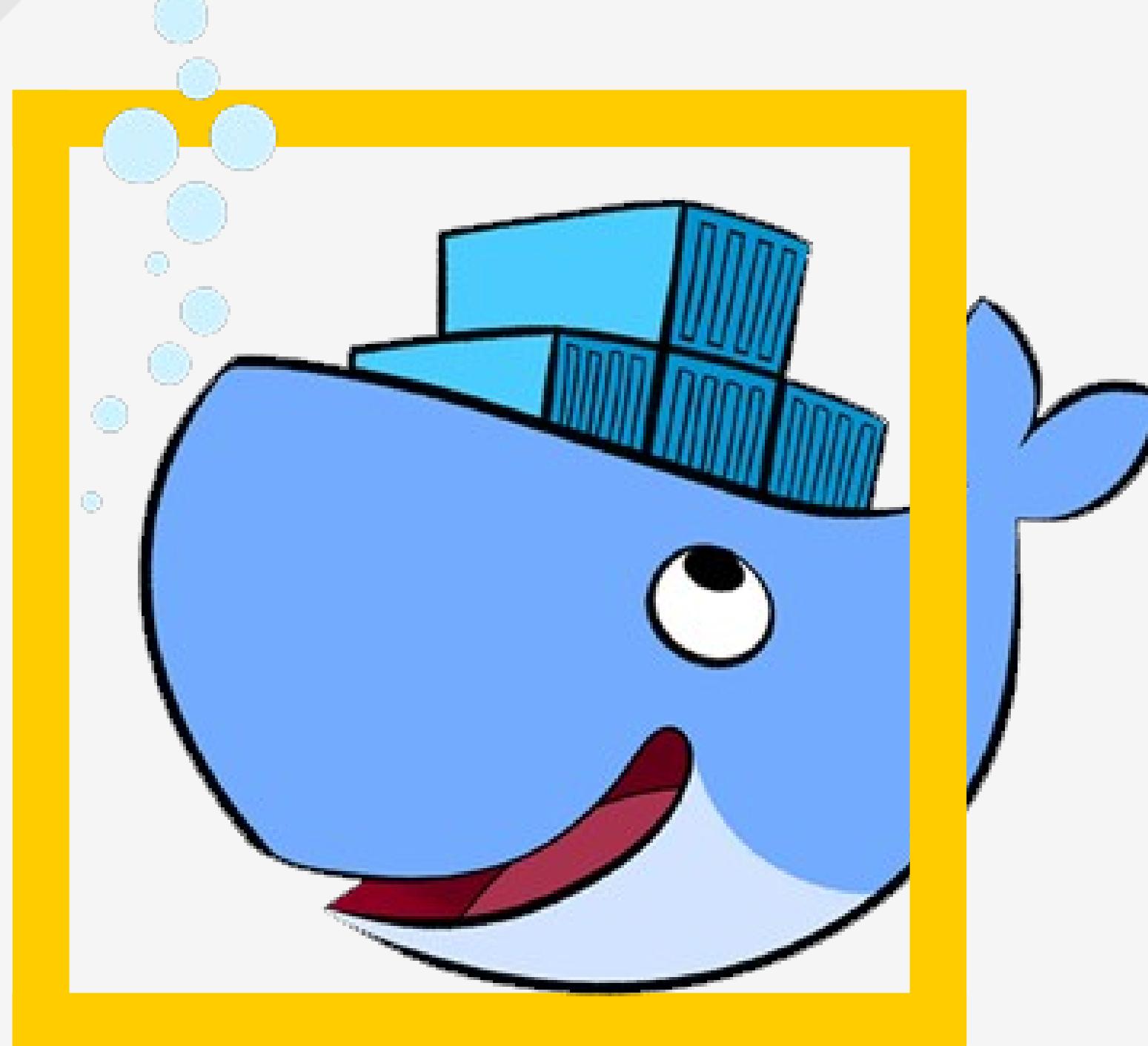
VS VM



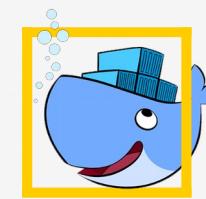
<Docker vs. Virtual Machine>



Source: <https://www.docker.com/whatisdocker/>

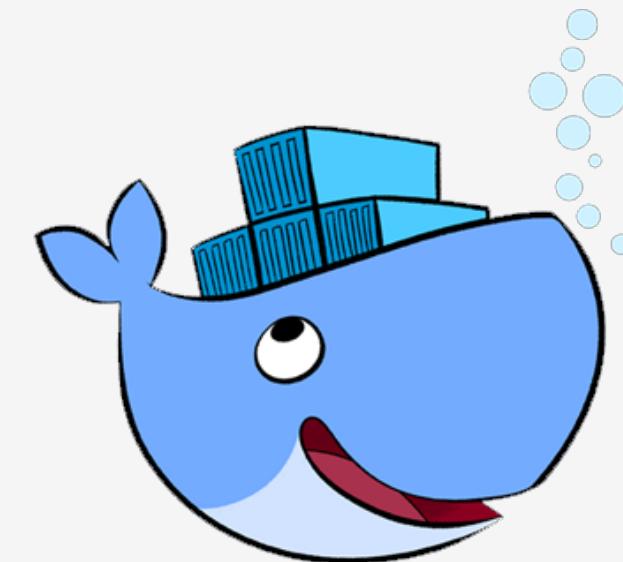


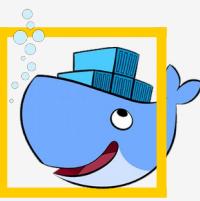
**Preparando
Ambiente**



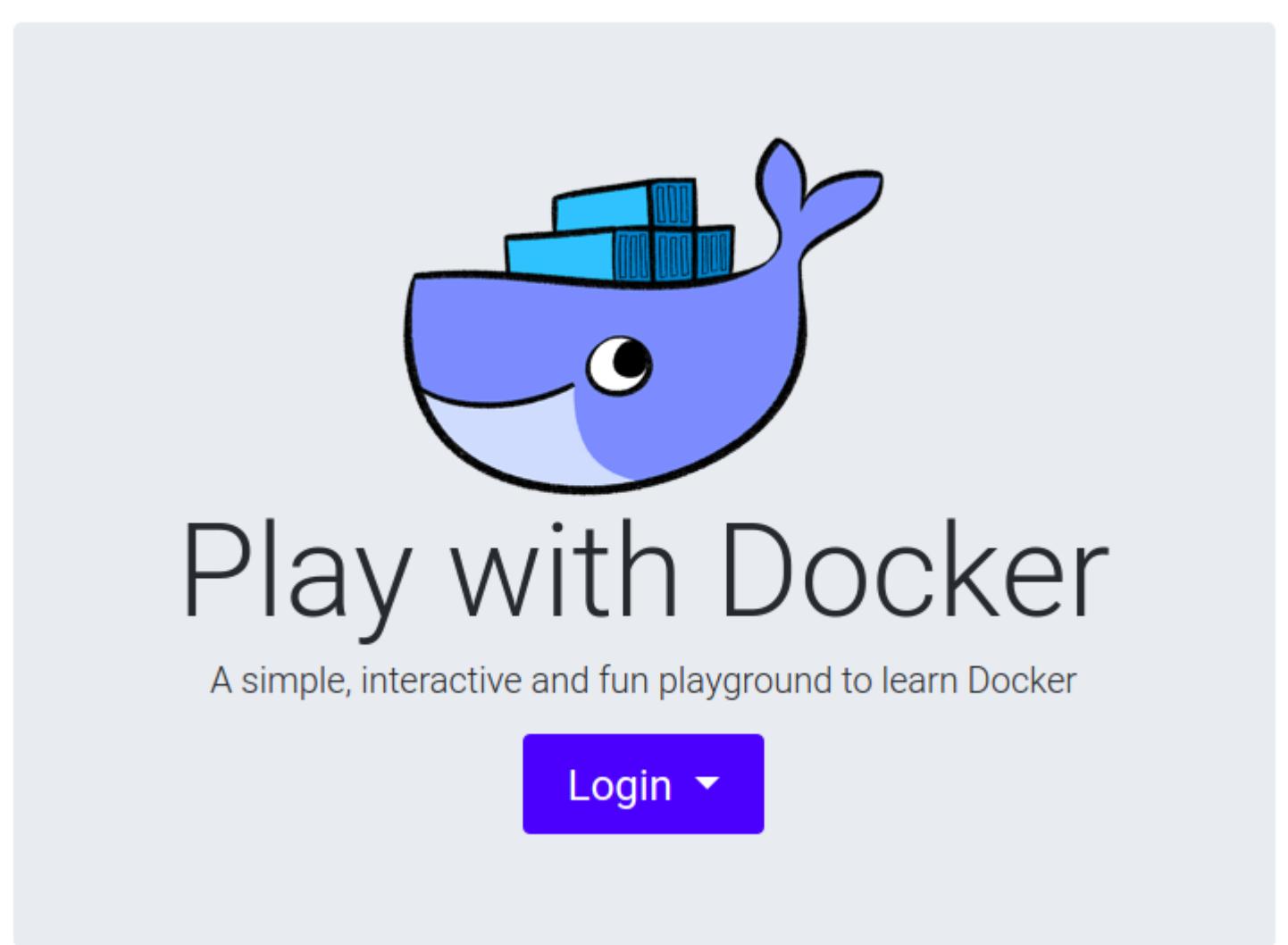
<Preparando Ambiente>

> Utilizar neste momento a máquina virtual no Software Virtual box





Ou <Play Docker>

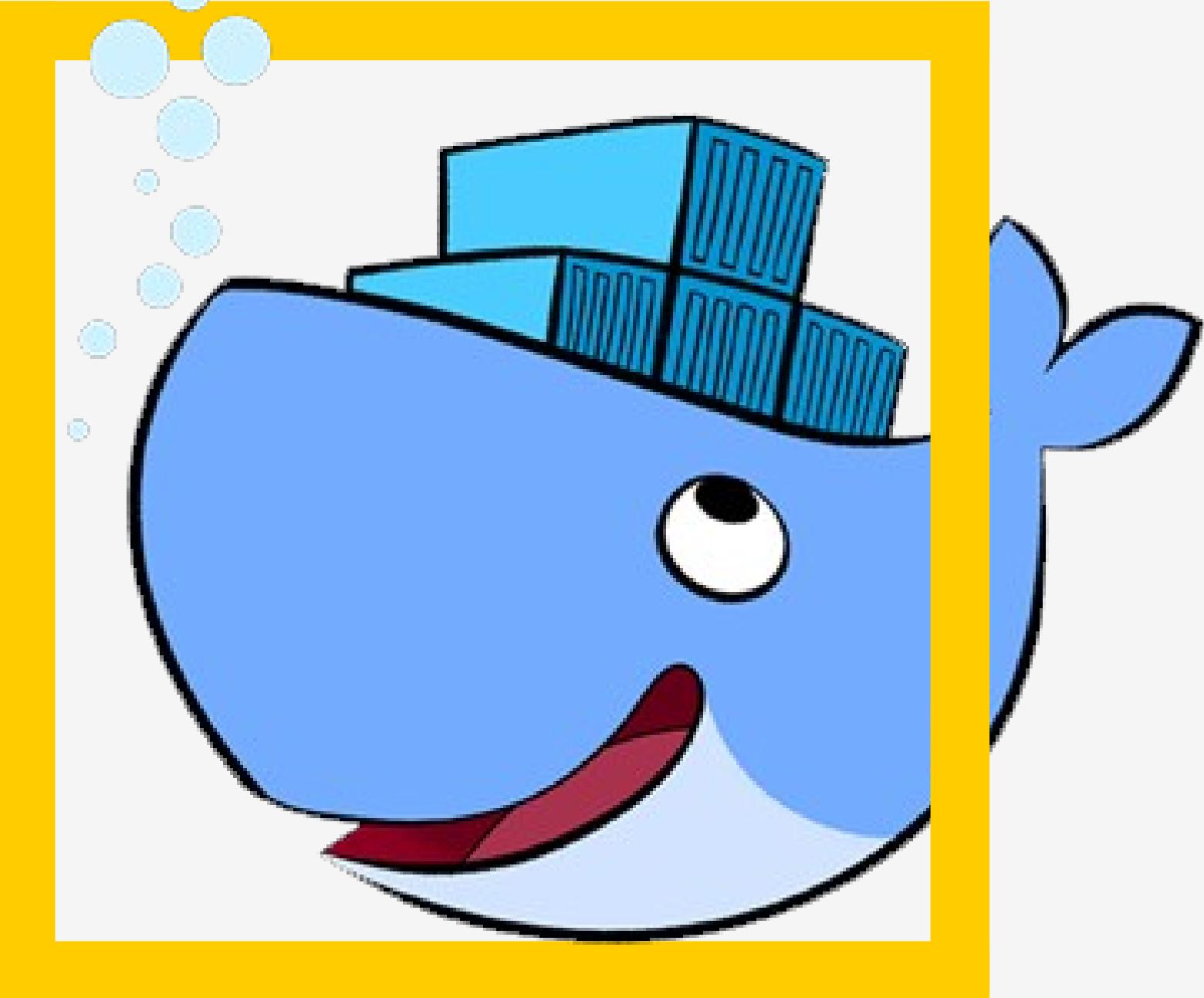


The image shows the landing page of the Play with Docker website. It features a large blue cartoon whale carrying several blue shipping containers on its back. Below the whale, the text "Play with Docker" is displayed in a large, bold, black font. Underneath that, a smaller line of text reads "A simple, interactive and fun playground to learn Docker". At the bottom center is a purple rectangular button with the word "Login" and a dropdown arrow.

Play with Docker

A simple, interactive and fun playground to learn Docker

Login ▾



Instalando **Docker**

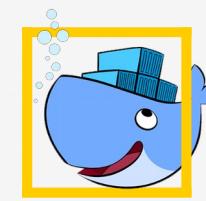


<Instalando Docker>

> O Docker deixou de ser apenas um software para virar um conjunto deles: um ecossistema.

Nesse ecossistema possui o seguinte:

- ✓ **Docker Engine**
- ✓ **Docker Compose**
- ✓ **Docker Machine**



<Instalando Docker>

> Para instalar o docker é simples:

```
$ sudo su
```

```
# wget -qO- https://get.docker.com/ | sh
```

```
# usermod -aG docker seu-user
```

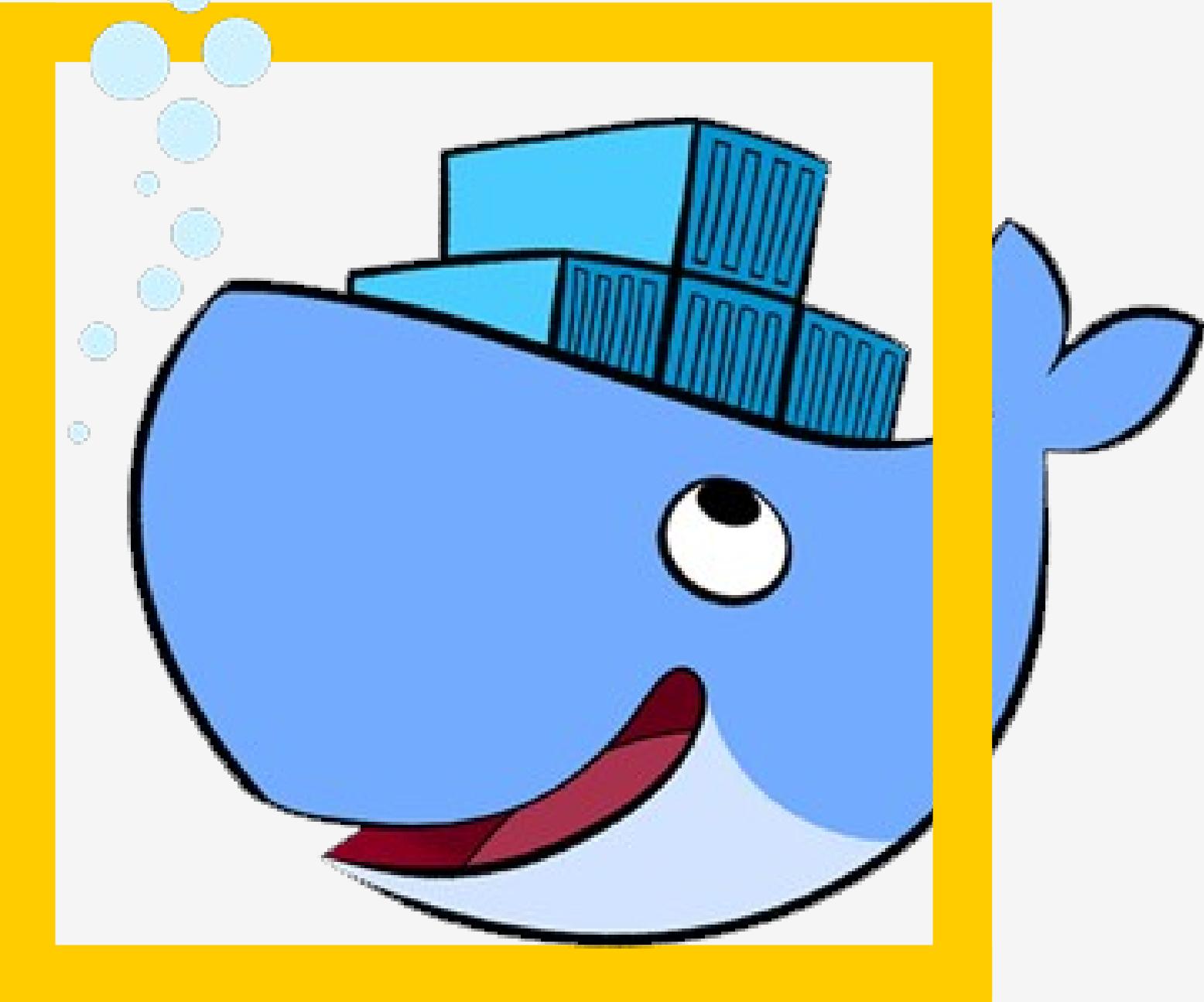
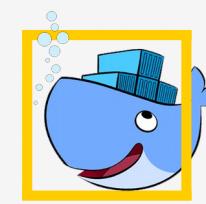


Imagen e
Containers



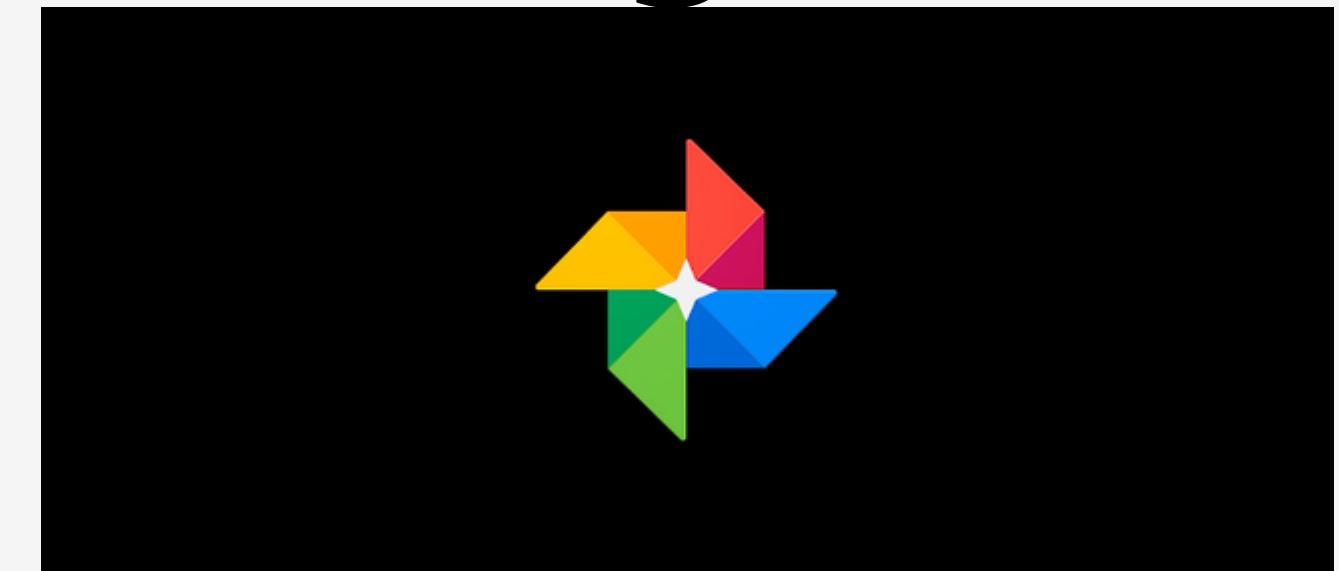
Começando com Docker

<Imagens e Containers>

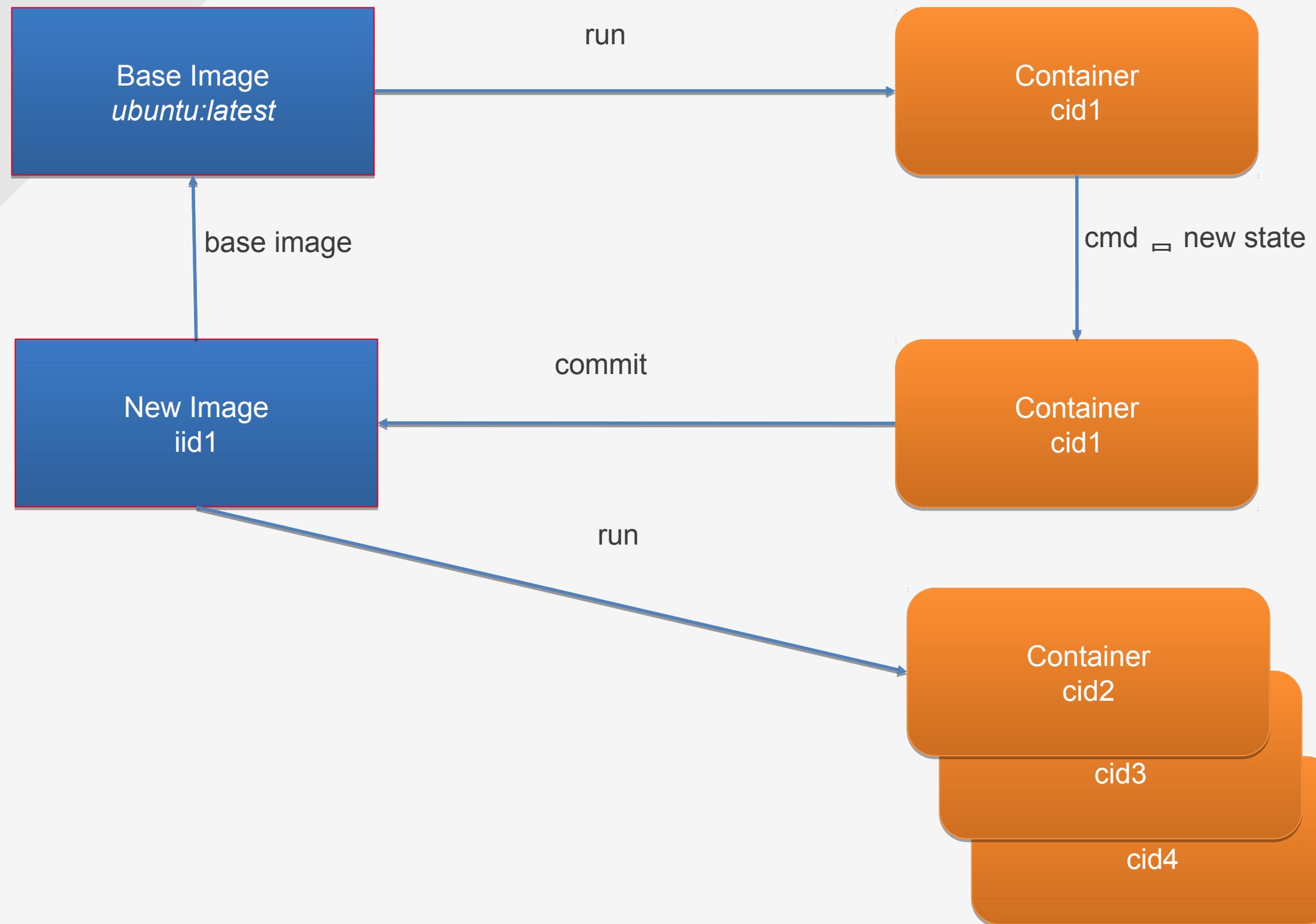
Contaneir

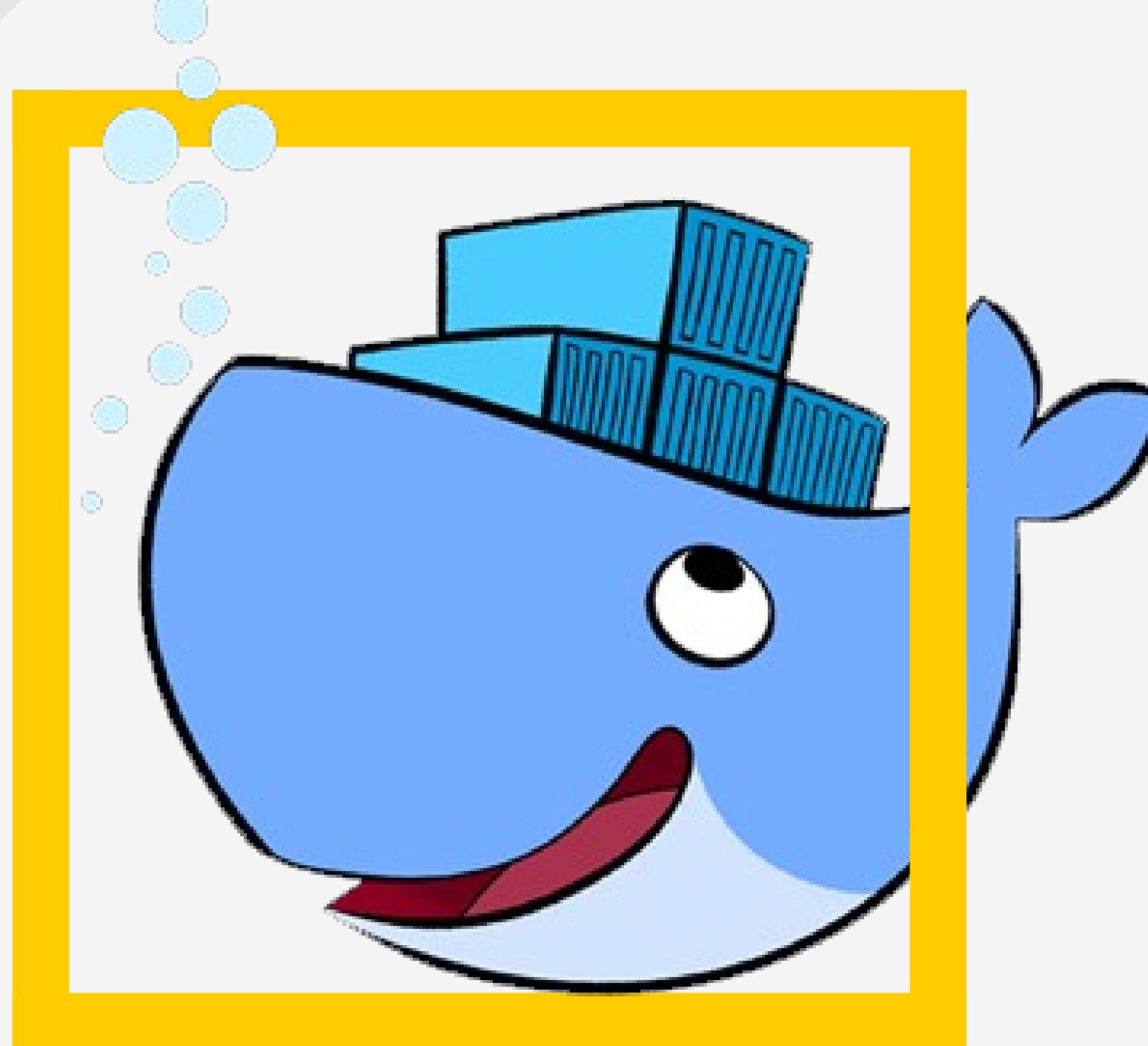


Imagen

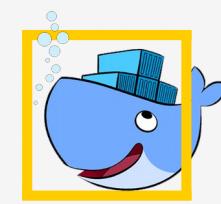


<Imagens e Containers>



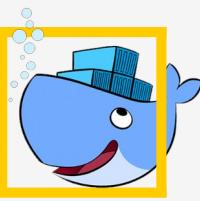


Dockerfile



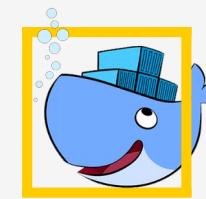
<Dockerfile>

- Criar images de forma automatizada usando um script: «Dockerfile»
- Podemos trabalhar com controle de versão como Git ou SVN, com suas dependências salva em cada build; Docker Hub {<https://hub.docker.com/>} repositório para armazenar suas images e poder usar em qualquer lugar..



<Dockerfile Exemplo>

- Dockerfile:
- FROM ubuntu
- ENV DOCK_MESSAGE Hello My World
- ADD dir /files
 - CMD ["bash", "someScript"]



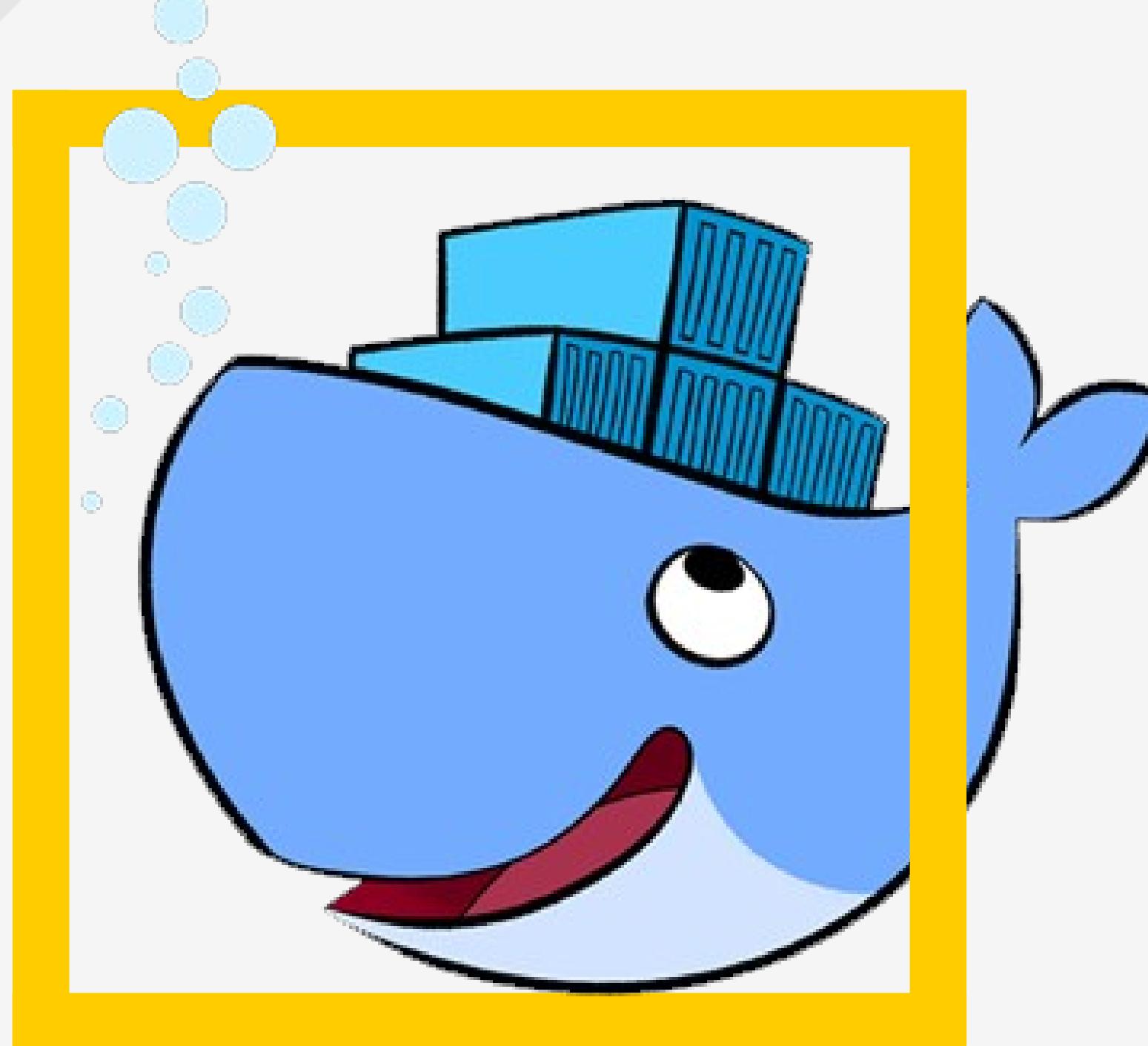
<Dockerfile Exemplo>

> Criando uma imagem

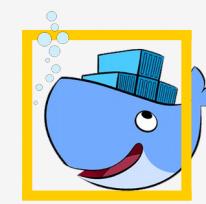
```
$ docker build [DockerFileDir] -t name
```

> Inspecionando uma imagem

```
$ docker inspect [imageId]
```



Hand-ons



<Hand-ons>

>Simple Command - Ad-Hoc Container

```
$ docker run ubuntu echo Hello World
```

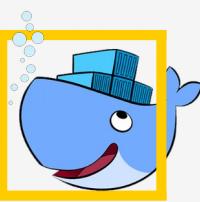


<Hand-ons>

> Criando Container

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

```
root@container-id: #
```



<Hand-ons>

>Sair do container(Matar)

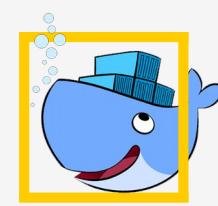
```
container-id:/# "CTRL+d"
```

>Sair do container(Manter Vivo)

```
container-id:/# "CTRL+pq"
```

>Acessando container ativo

```
# docker attach container-id
```



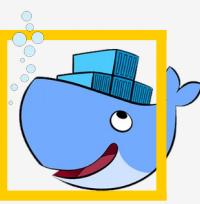
<Hand-ons>

> Verificando containers em execução

```
$ docker ps
```

> Verificando imagens

```
$ docker images
```



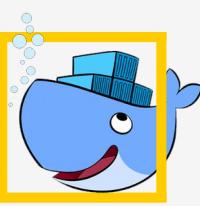
<Hand-ons>

> Parando Container em execução

```
$ docker stop container-id
```

> Removendo containers parados

```
$ docker rm container-id
```



<Hand-ons>

> Removendo images

```
$ docker rmi image-id
```



Obrigado



<https://www.linkedin.com/in/diogoalvesbarbosa/>
<https://www.instagram.com/diogo.alvesoficial/>
diogo.alves.barbosa@gmail.com