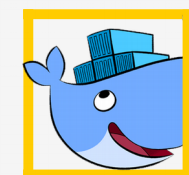


Começando com  
**Docker**



# <Diogo Alves>



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



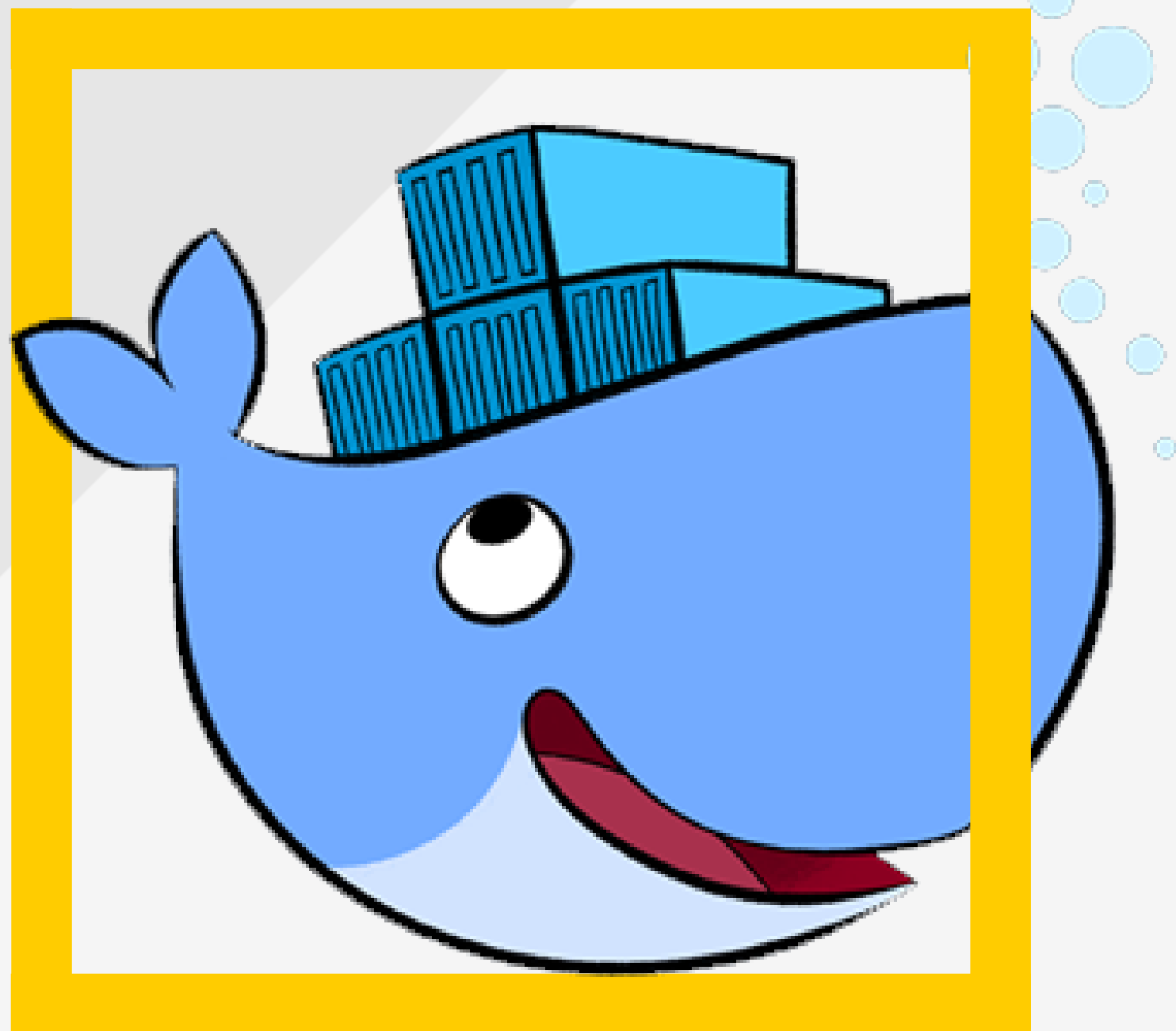
Formação:  
- Sistemas de Informação



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



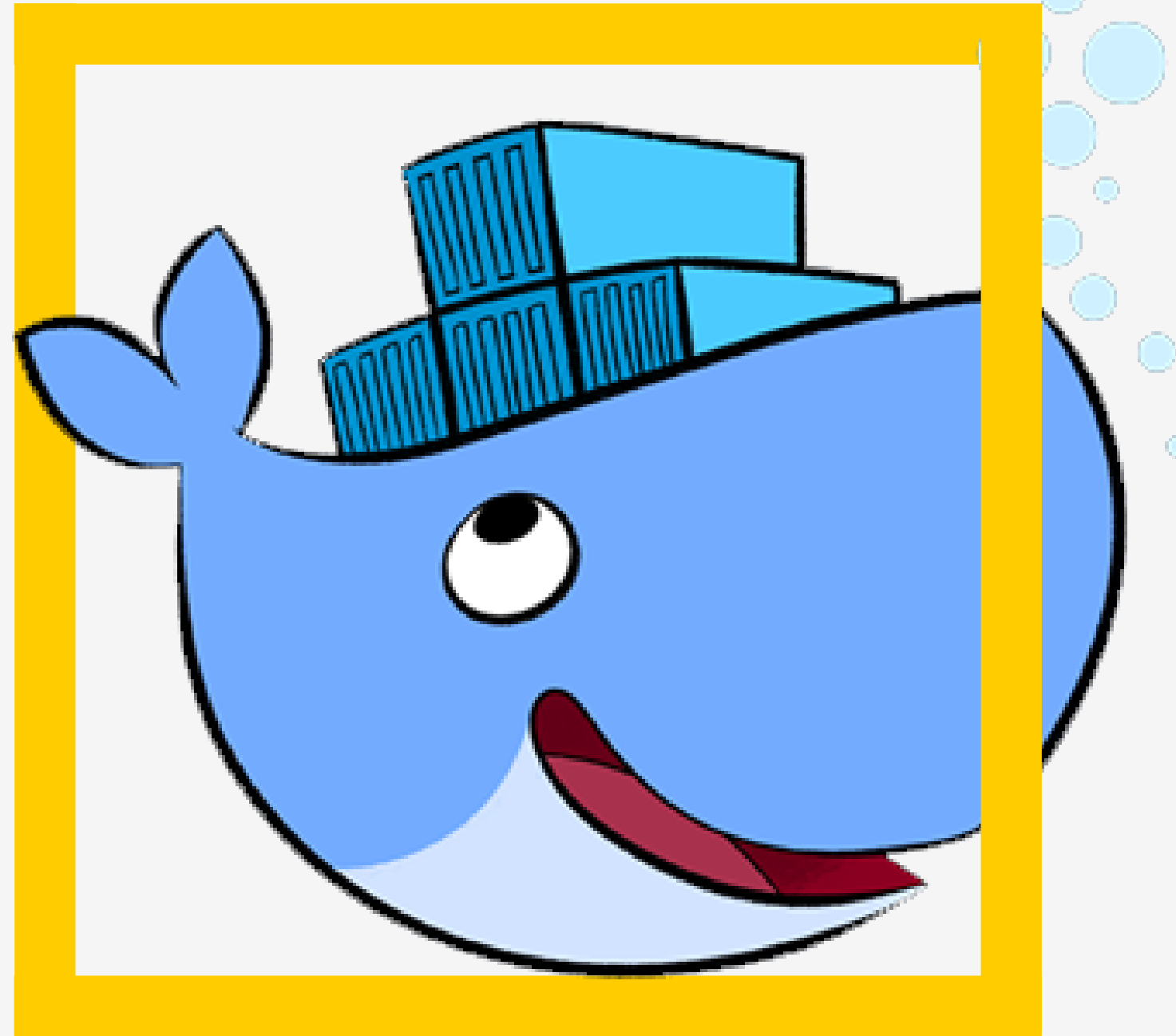
Algumas Certificações:  
- CEH – Ethical Hacking  
- ISO 27001  
- Asterisk Administrator  
- AWS



# Agenda

- O que é DockerHub?
- Armazenamento no Docker
- Entendendo a rede no Docker
- Gerenciando containers com Docker-compose
- Hand-ons

Começando com  
**Docker**



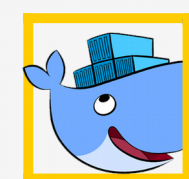
O que é  
**DockerHub?**

# <O que é DockerHub?>



Começando com Docker

- ✓ Repositório compartilhado de Images Docker
- ✓ Armazena imagens oficiais e não-oficiais
- ✓ Suporte a repositórios privados

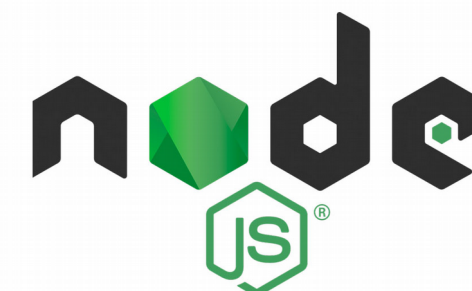


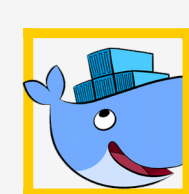
# <O que é DockerHub?>

Exemplos de empresas com Images Docker oficiais



NGINX



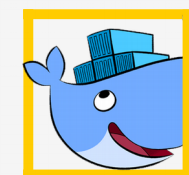


# <O que é DockerHub?>

## Exemplos de Images Docker não-oficiais


The screenshot shows the Docker Hub interface for the `jrcs/letsencrypt-nginx-proxy-companion` image. The page has a blue header with the Docker Hub logo, a search bar containing "nginx", and links for "Explore", "Pricing", "Sign In", and "Sign Up". The main content area features the image icon (a blue cube), the name `jrcs/letsencrypt-nginx-proxy-companion` with a star icon, and the text "By jrcs • Updated 2 months ago" and "LetsEncrypt container to use with nginx as proxy". A "Container" tag is visible. Below this are tabs for "Overview", "Tags", "Dockerfile", and "Builds". The "Overview" tab is active, showing a description: "letsencrypt-nginx-proxy-companion is a lightweight companion container for `nginx-proxy`. It handles the automated creation, renewal and use of Let's Encrypt certificates for proxied Docker containers. Please note that `letsencrypt-nginx-proxy-companion` does not work with ACME v2 endpoints yet." A "Features:" section lists several bullet points. On the right, there are three sections: "Docker Pull Command" with the command `docker pull jrcs/letsencrypt-nginx-proxy-companion`, "Owner" showing the user `jrcs` with a profile picture, and "Source Repository" linking to the GitHub repository `JrCs/docker-letsencrypt-nginx-proxy-companion`. At the top of the main content area, there are statistics: "build error", "release v1.10", "32.3MB", "12 layers", "docker stars 507", and "docker pulls 98M".

<https://hub.docker.com/r/jrcs/letsencrypt-nginx-proxy-companion>



# <O que é DockerHub?>

## Criando seu docker ID



### Docker Identification


In order to get you started, let us get you a Docker ID.  
Already have an account? [Sign In](#)

! Docker ID is required.

☐ I agree to Docker's [Terms of Service](#).

☐ I agree to Docker's [Privacy Policy](#) and [Data Processing Terms](#).

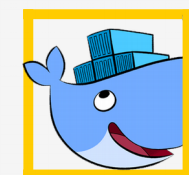
☐ (Optional) I would like to receive email updates from Docker, including its various services and products.

☐ I'm not a robot 

[Continue](#)

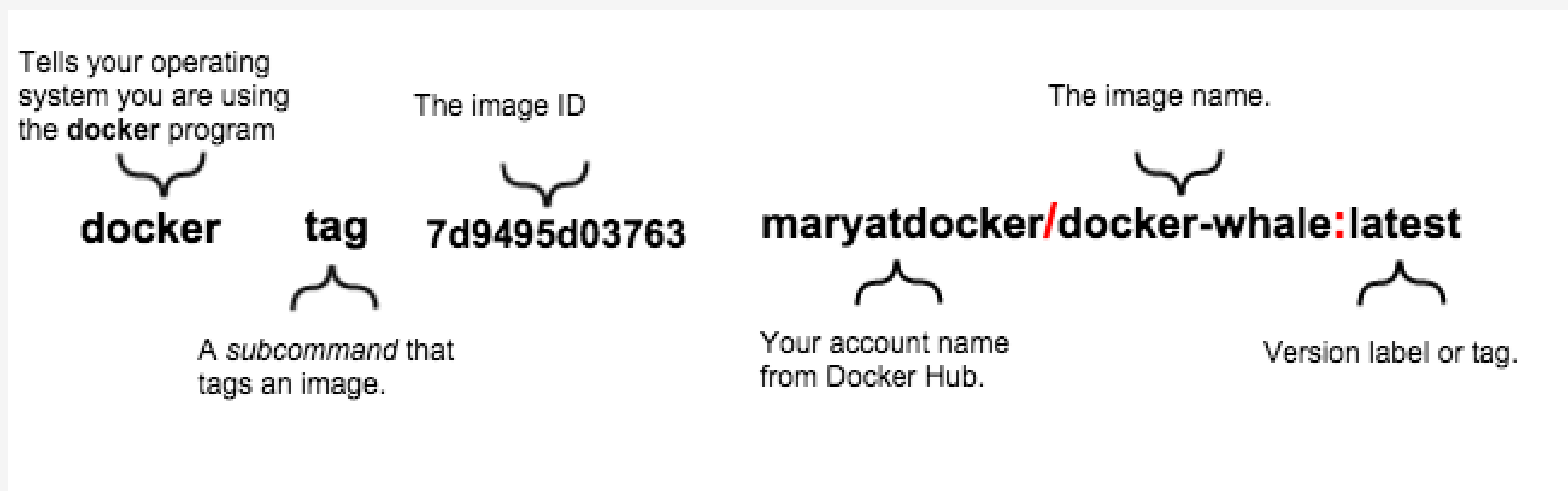
<https://hub.docker.com/signup>





# <O que é DockerHub?>

## Taggeando sua imagem



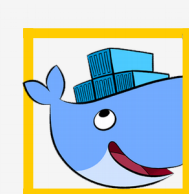
# <O que é DockerHub?>



Começando com Docker

**> No terminal execute**

```
$ docker tag nome-da-imagem SEU_DOCKER_ID/nome-da-  
imagem:latest
```



# <O que é DockerHub?>

## > Push

```
$ docker login
```

```
$ docker image push dockerID/nome-da-imagem
```

## > Pull

```
$ docker image pull dockerID/nome-da-imagem
```

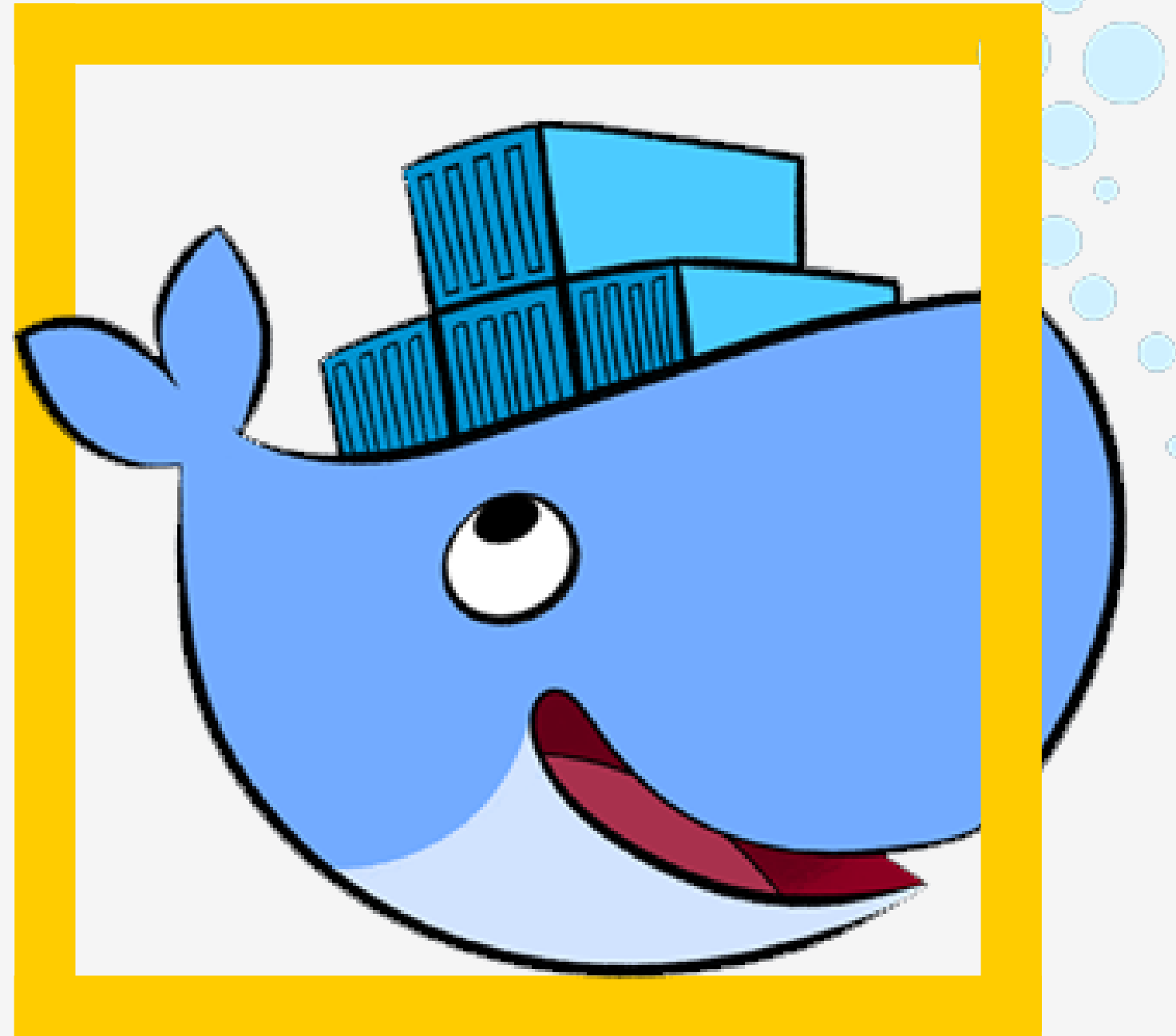
# <O que é DockerHub?>



Começando com Docker

## > Buscando imagens

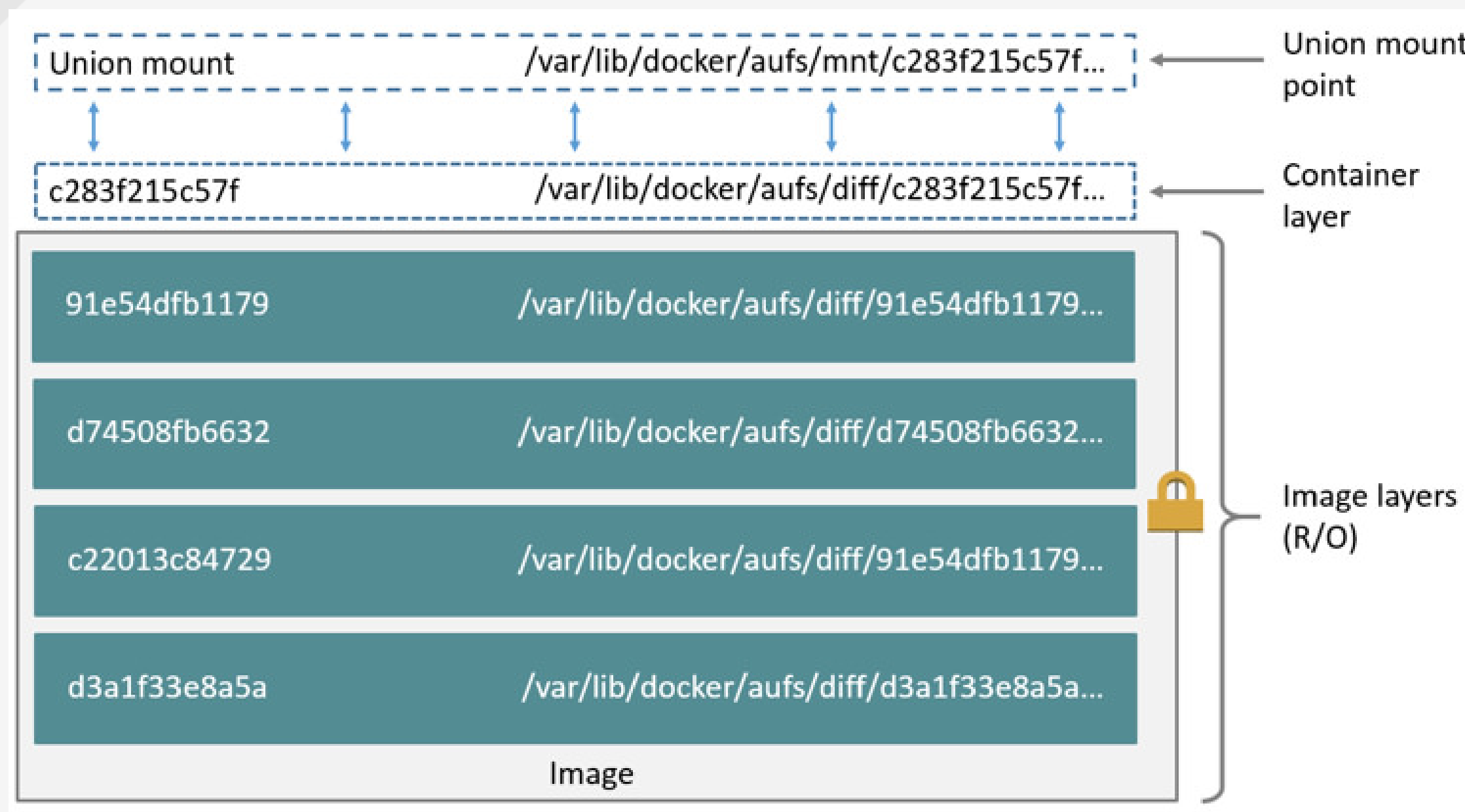
```
$ docker search nome-da-imagem
```



Armanzenamento no  
**Docker**

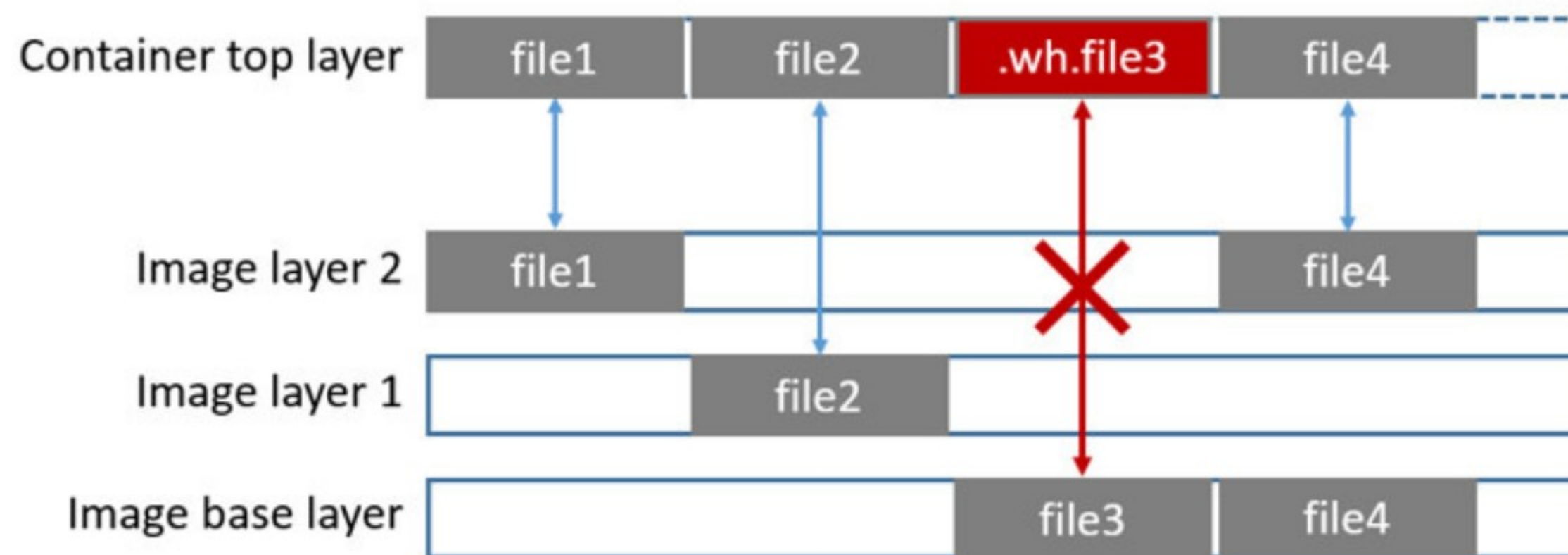


# <Armazenamento no Docker>

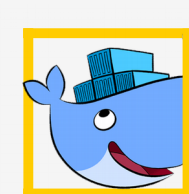




# <Armazenamento no Docker>



Docker container  
(AUFS storage-driver demonstrating whiteout file)



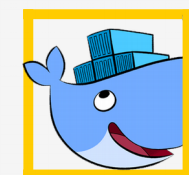
# <Armazenamento no Docker>

## > Utilizando Volumes

```
$ mkdir /home/user/container1
```

```
$ docker container run -v  
/home/user/container1:/home/user/container1 ubuntu
```



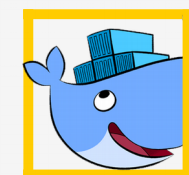


# <Armazenamento no Docker>

## > Mapeamento via container de dados

```
$ docker create -v /dbdata --name dbdata postgres  
/bin/true
```

```
$ docker container run -d --volumes-from dbdata --  
name db2 postgres
```

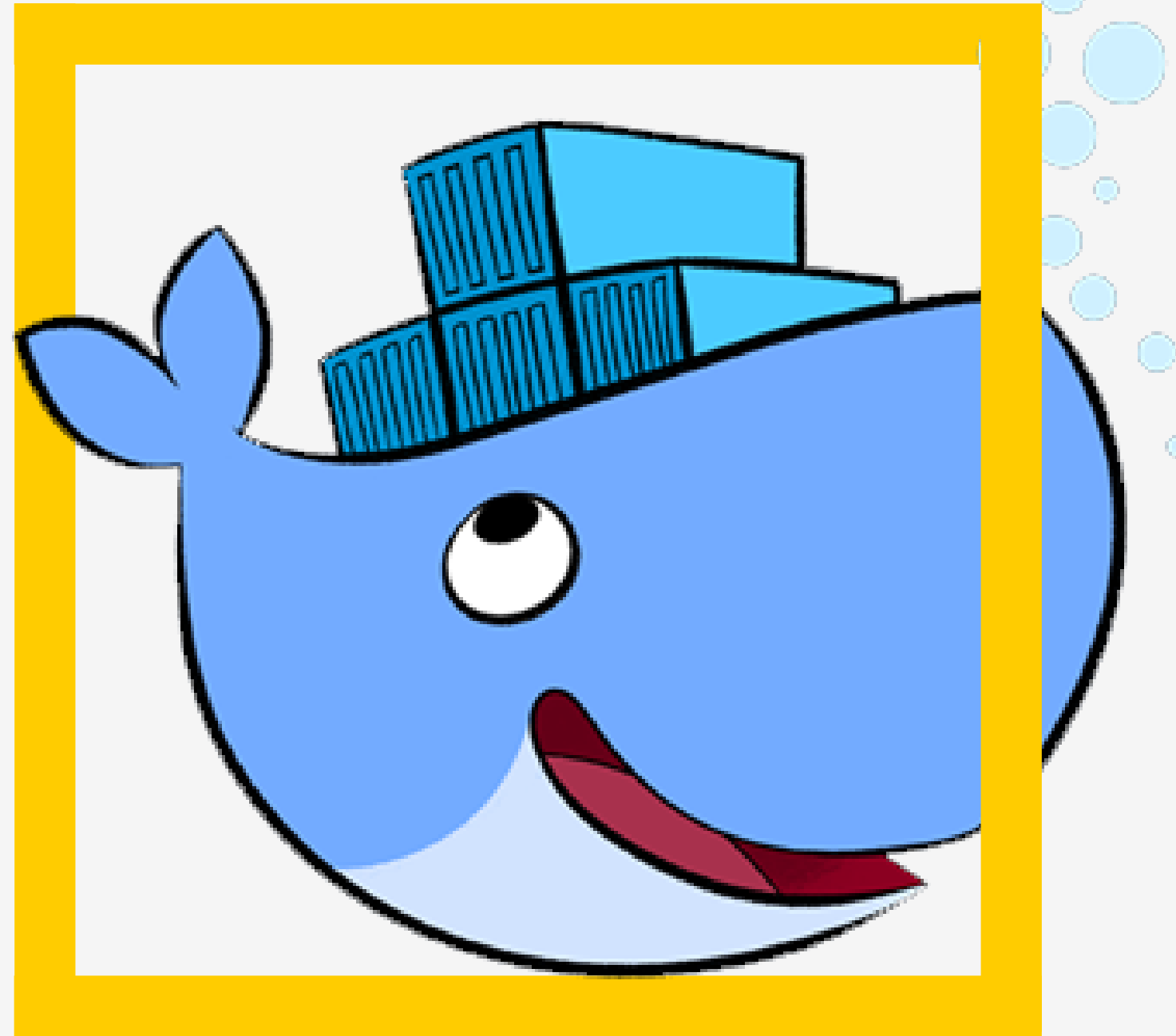


# <Armazenamento no Docker>

## > Mapeamento de volumes

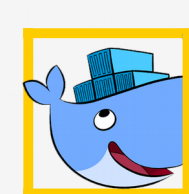
```
$ docker volume create --name dbdata
```

```
$ docker container run -d -v dbdata:/var/lib/data  
postgres
```



Entendendo a rede no

# Docker



# < Entendendo a rede Docker >

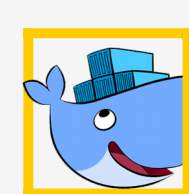
- > Redes Bridge
- > Redes None
- > Redes Overlay
- > Redes Usuários



# < Entendendo a rede Docker >

## > Redes Docker

```
$ docker network ls
```



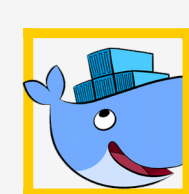
# <Entendendo a rede Docker>

## > Exemplo

```
$ docker container run -d --name db -e  
MYSQL_ROOT_PASSWORD=minhasenha mysql
```

```
$ docker container run -d -p 80:80 --name app  
--link db tutum/apache-php
```

```
$ docker container exec -it app ping db
```

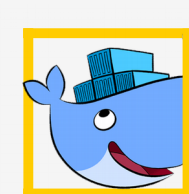


# <Entendendo a rede Docker>

## > Exemplo

```
$ docker network create --driver bridge rede-  
isolada
```

```
$ docker container run -itd --net rede-isolada  
alpine sh
```

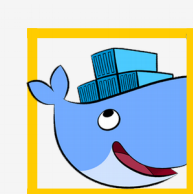


# < Entendendo a rede Docker >

## > Inspeccionado redes Docker

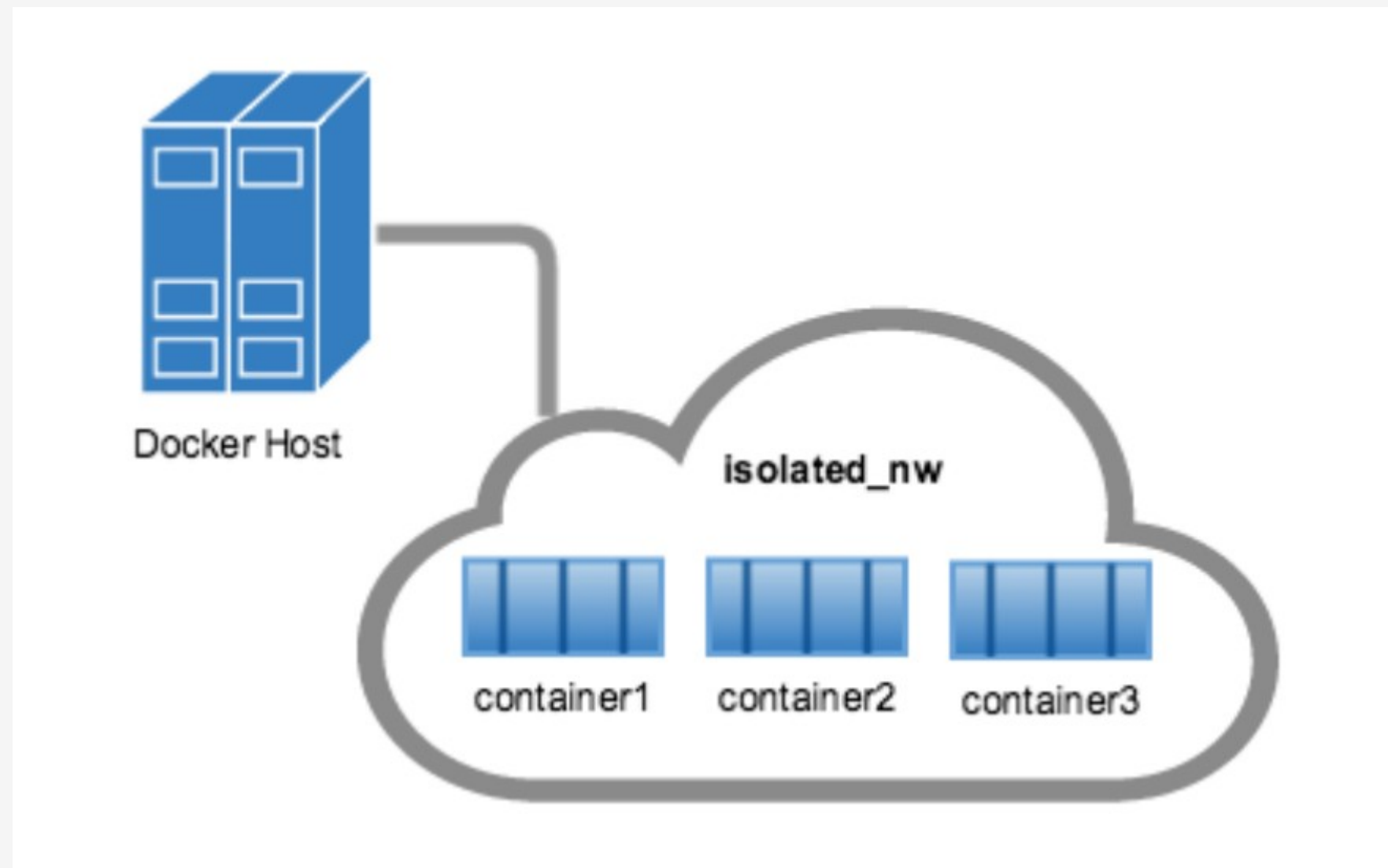
```
$ docker network inspect rede
```





# < Entendendo a rede Docker >

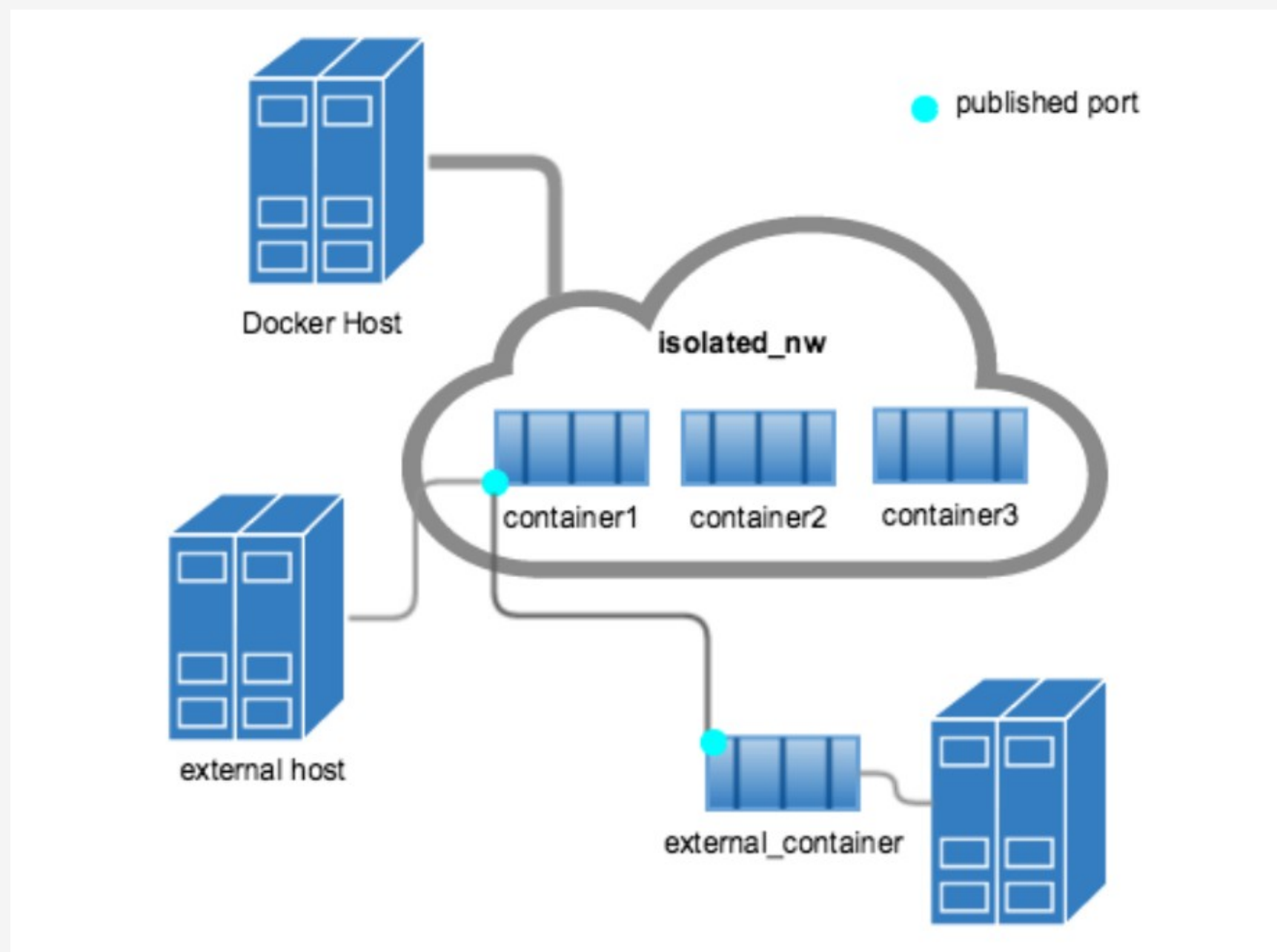
## > Redes Isoladas

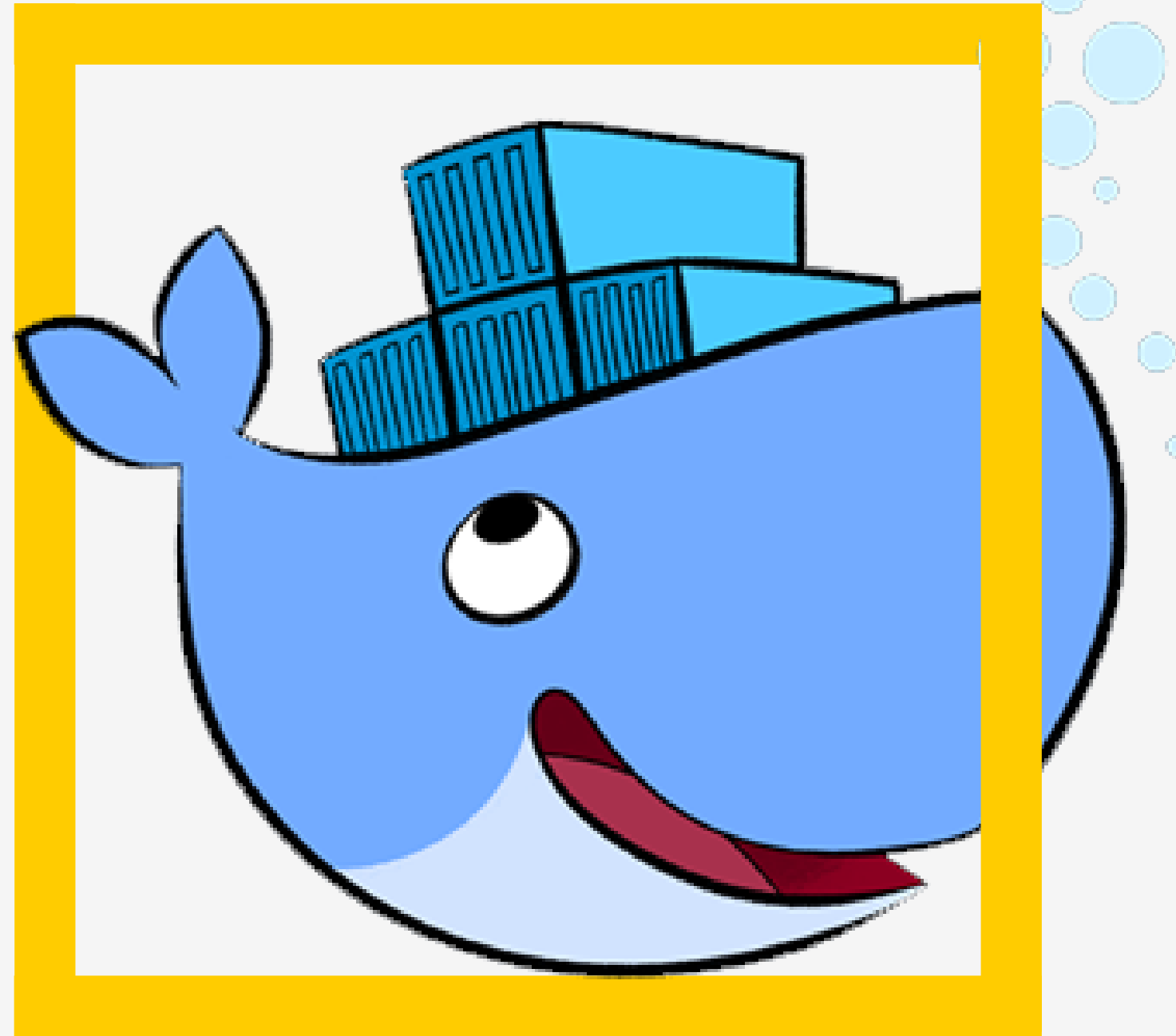




# < Entendendo a rede Docker >

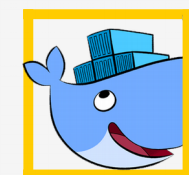
## > Redes Overlay





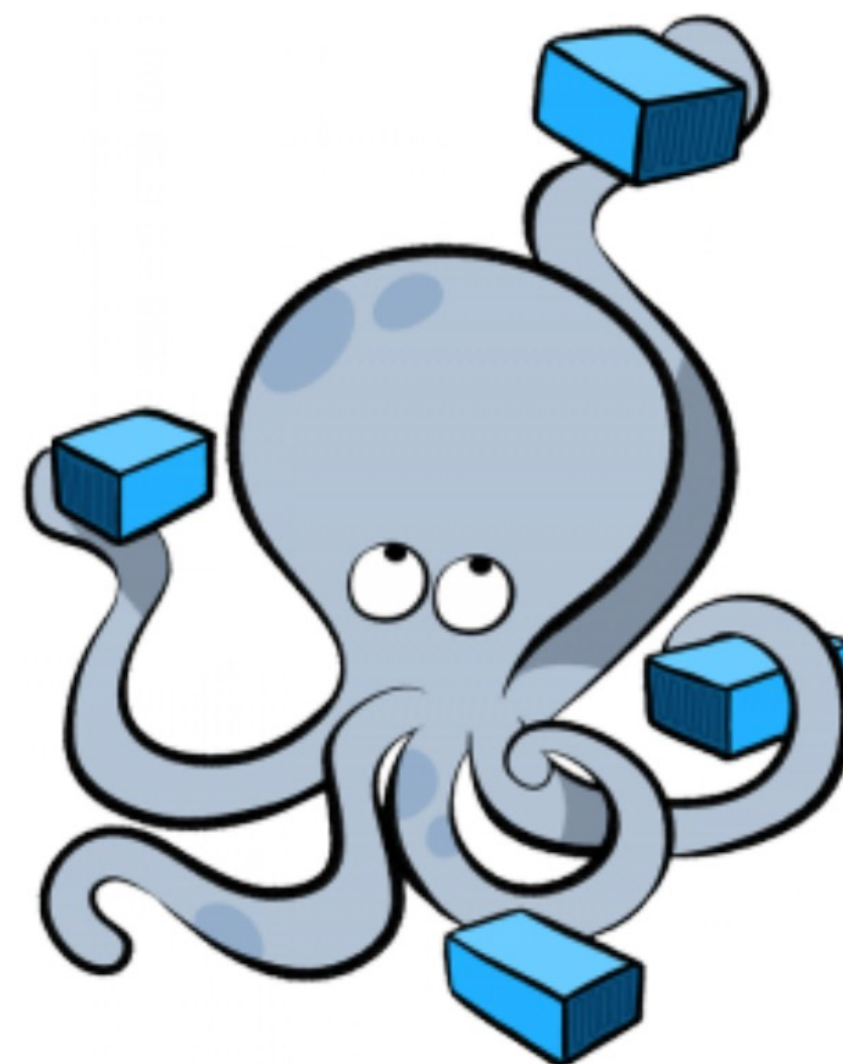
Gerenciando containers com

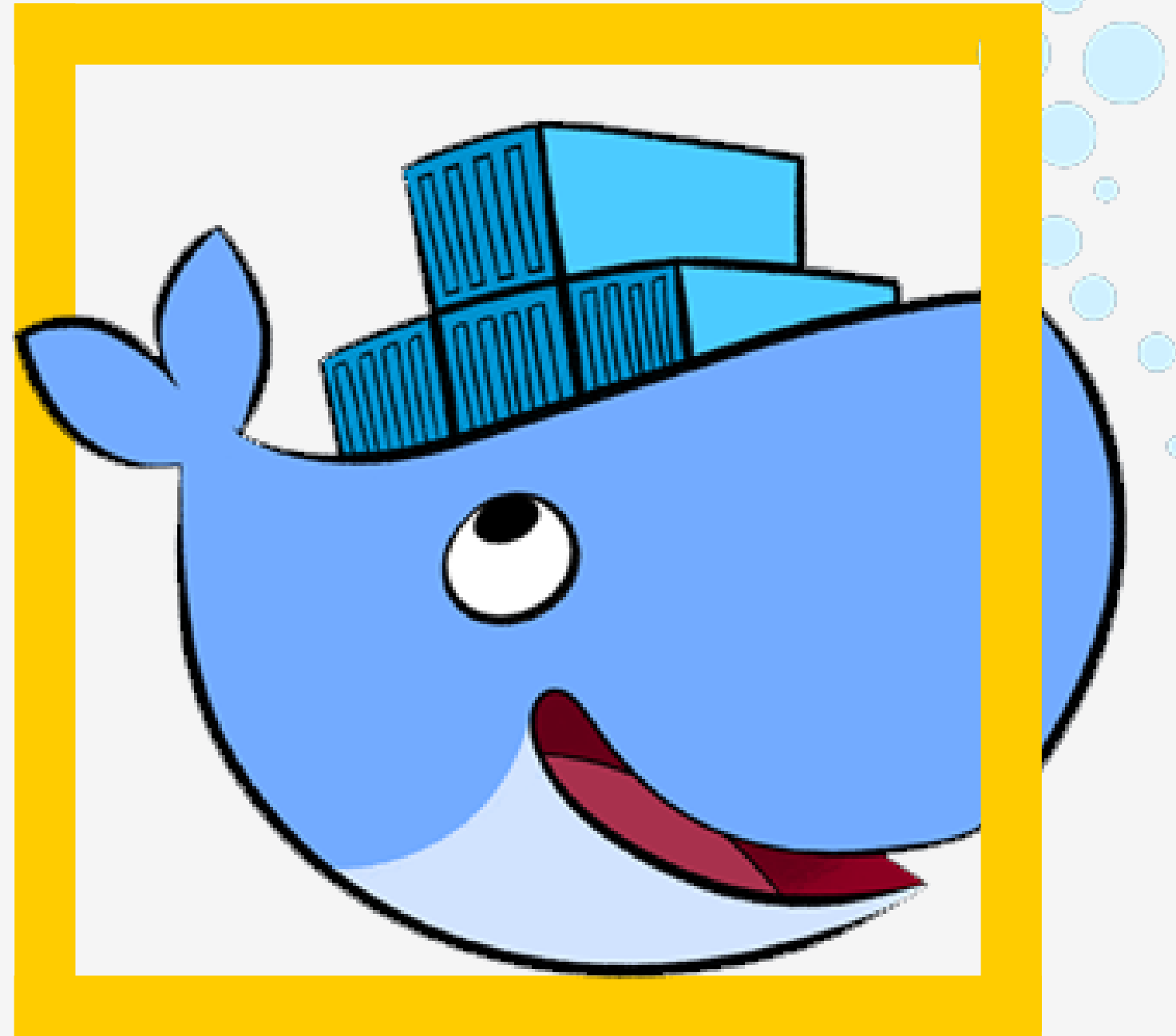
# Docker-Compose



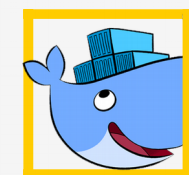
# <Gerenciando containers com Docker-Compose>

```
version: '3'
services:
  web:
    build:
      context: ./dir
      dockerfile: Dockerfile-alternate
    args:
      versao: 1
  ports:
    - "5000:5000"
  redis:
    image: redis
```





# Hand-ons



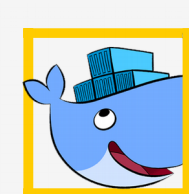
# <Hand-ons>

## >Vamos lá?

\$ Criar um docker-compose com aplicação  
Nginx expondo a porta 8080 para o host e  
porta 80 para container

\$ Rede segregada

\$ persistir as configurações do Nginx em  
volume /home/user/nginx



# Obrigado



<https://www.linkedin.com/in/diogoalvesbarbosa/>  
<https://www.instagram.com/diogo.alvesoficial/>  
[diogo.alves.barbosa@gmail.com](mailto:diogo.alves.barbosa@gmail.com)