

Infraestrutura para Sistemas de Software

Prof. Dr. Carlos Alberto da Silva



Módulo 5 - Arquitetura de micros serviços

Unidade 1 - Os conceitos de *containers* e *docker*



Os conceitos de virtualização por *Container* e *Docker*[®]

- *Container*
- *Docker*[®]
- Arquitetura *Docker*[®]
- Plataforma *Docker*[®]

Container

Virtualização por *Container*

- É uma tecnologia que permite a criação de múltiplas instâncias isoladas de um determinado sistema operacional dentro de um único hospedeiro;
- É uma maneira de virtualizar aplicações dentro de um servidor.

Container

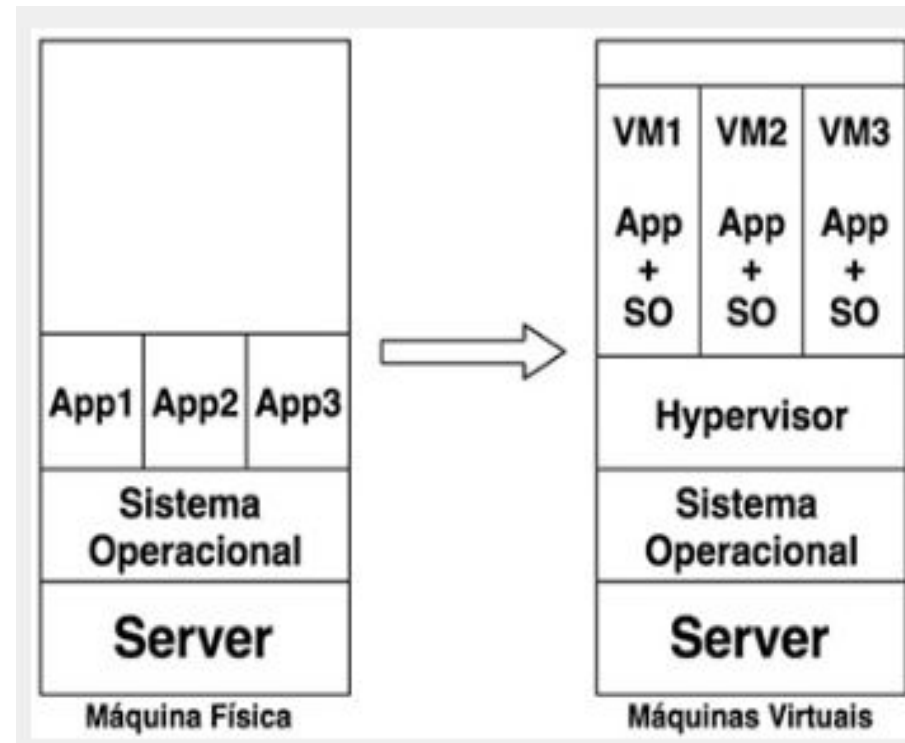
Virtualização por *Container*

Container usa uma quantidade muito menor de memória e espaço em disco que as VMs.

⇒ ***Container*** utiliza os recursos de seu próprio SO (*host*).

Container

Virtualização por *Container*



Fonte: Vitalino, 2016, p. 13 (plataforma de leitura).

Vantagens da virtualização por *Container*

- Tudo pode ser instalado em *Containers* no servidor;
- Usam bibliotecas do *kernel* do SO:
 - O ambiente torna-se portátil;
 - Torna o trabalho conjunto mais eficiente.

Vantagens da virtualização por *Container*

- Simplifica a metodologia *DevOps* facilitando o desenvolvimento ágil;
- Permite o escalonamento de aplicações;
- Torna processos de implantação e configuração mais simples.

Vantagens da virtualização por *Container*

- Facilita o gerenciamento do ambiente:
 - Economia de recursos;
 - Aplicação como pacote completo;
 - Replicação e padronização.

Vantagens da virtualização por *Container*

- No compartilhamento de recursos:
 - Os arquivos podem ser compartilhados entre o *host* e o *Container*;
 - Ou um volume pode ser distribuído para outros.

Docker[®]



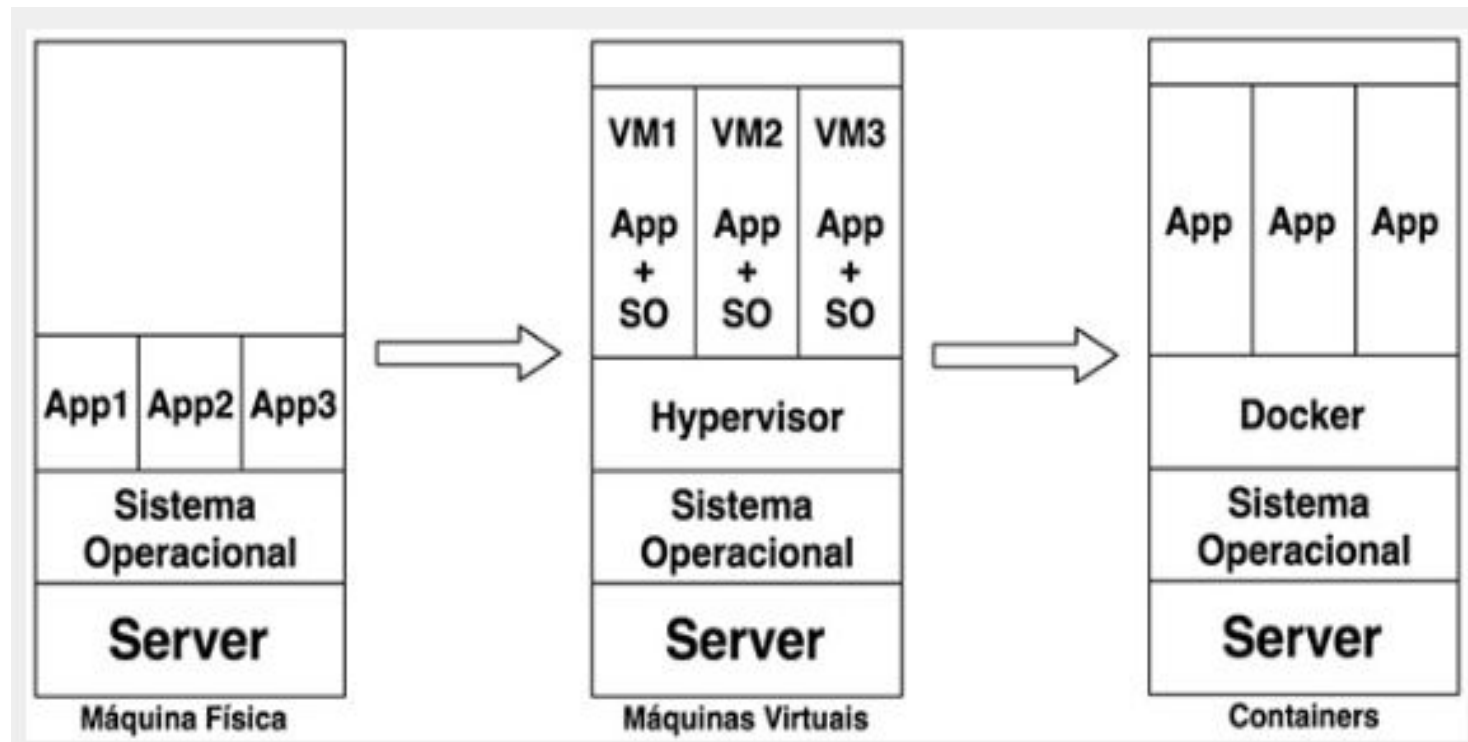
Fonte: www.docker.com

- Tecnologia *Open Source* permite dentro de um *container* de software as aplicações distribuídas para:
 - Desenvolver;
 - Testar;
 - Executar.
- ⇒ Empacotar uma aplicação de forma padronizada contendo: código, *runtime*, ferramentas, bibliotecas, etc.

Vantagens

- Possibilita a implantação de aplicações de forma mais ágil, confiável e estável, em qualquer ambiente.
- Controle mais granular dos recursos gerando mais eficiência da infraestrutura.

Virtualização por *Container*

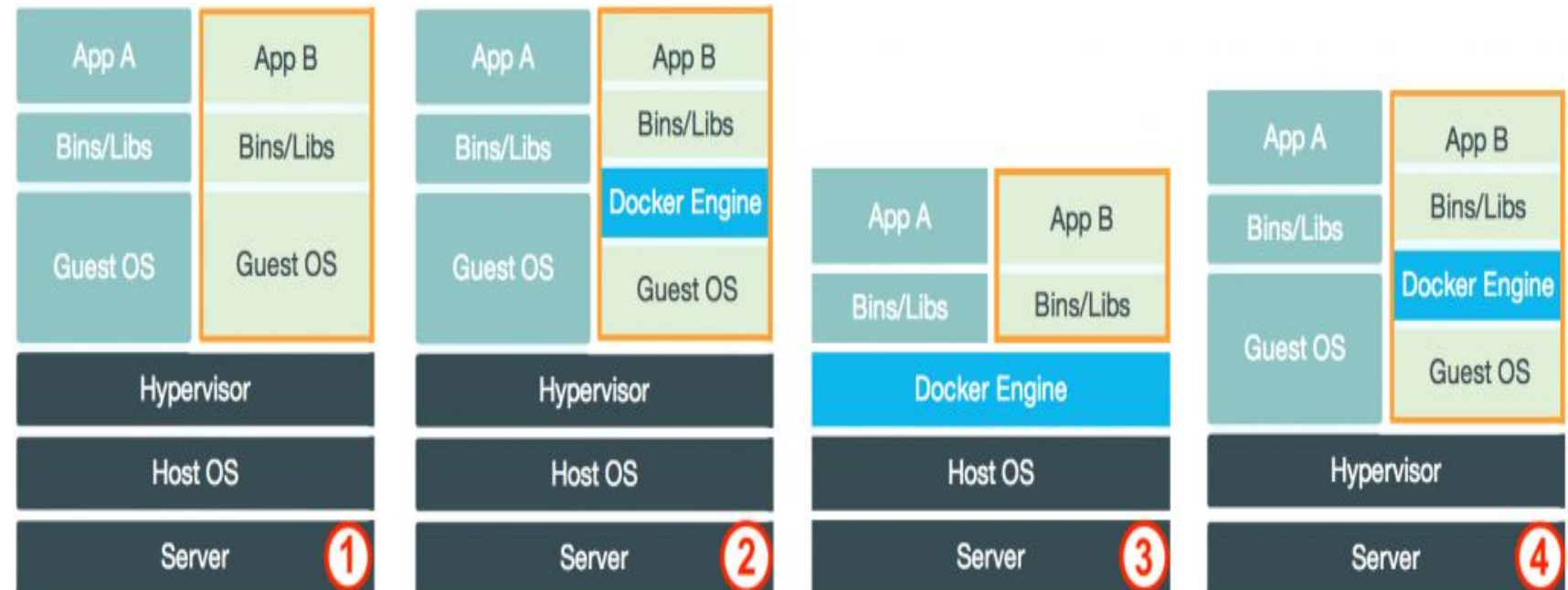


Fonte: Neto, 2016, p. 13 (plataforma de leitura).

Virtualização por *Container*

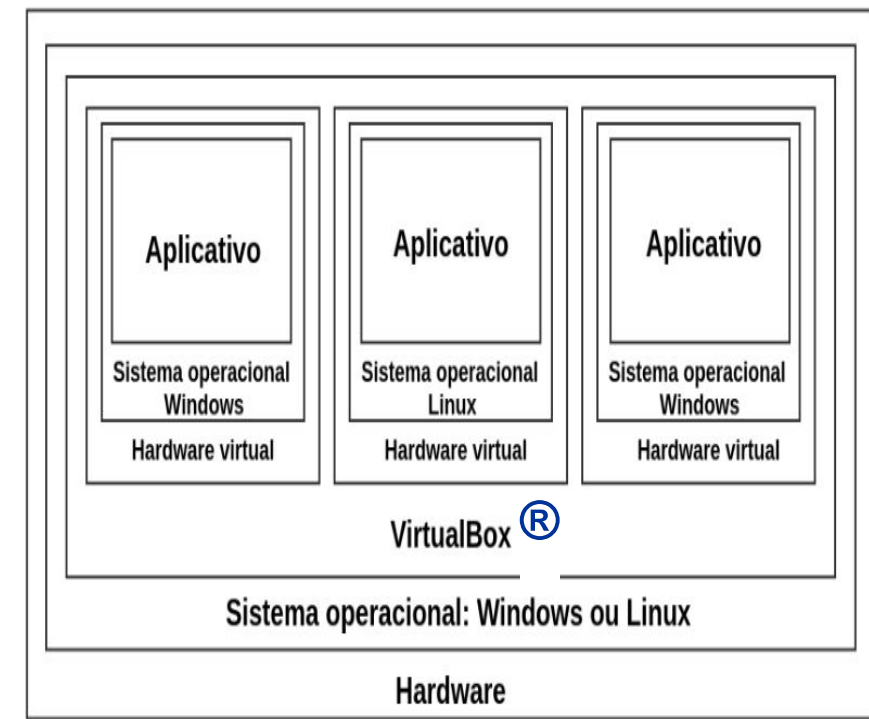
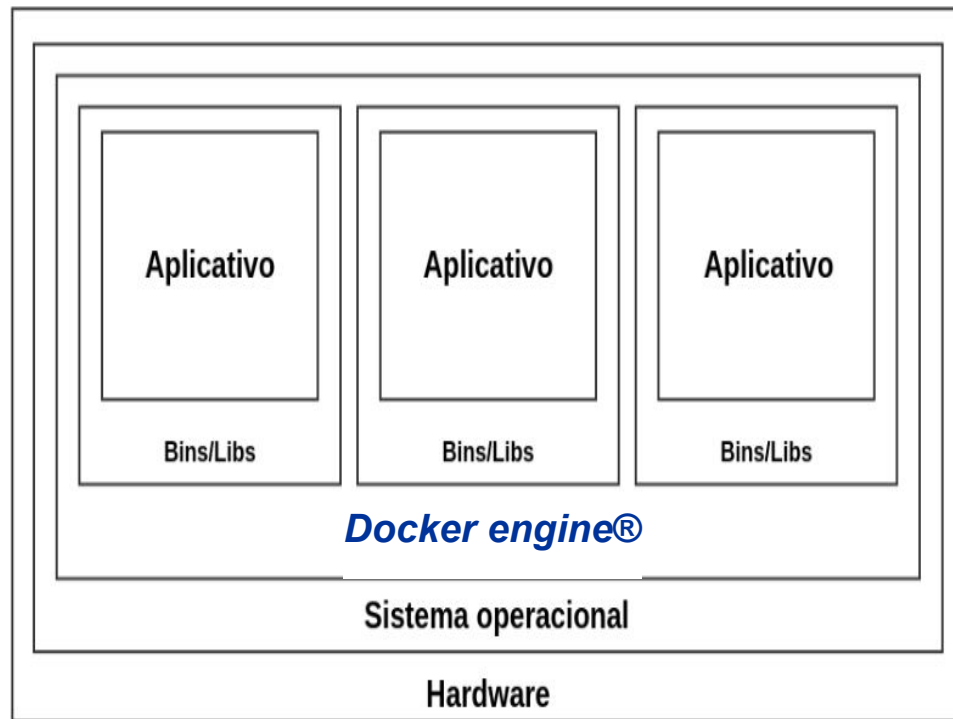
(1) Tradicional

(2-3-4) Docker®



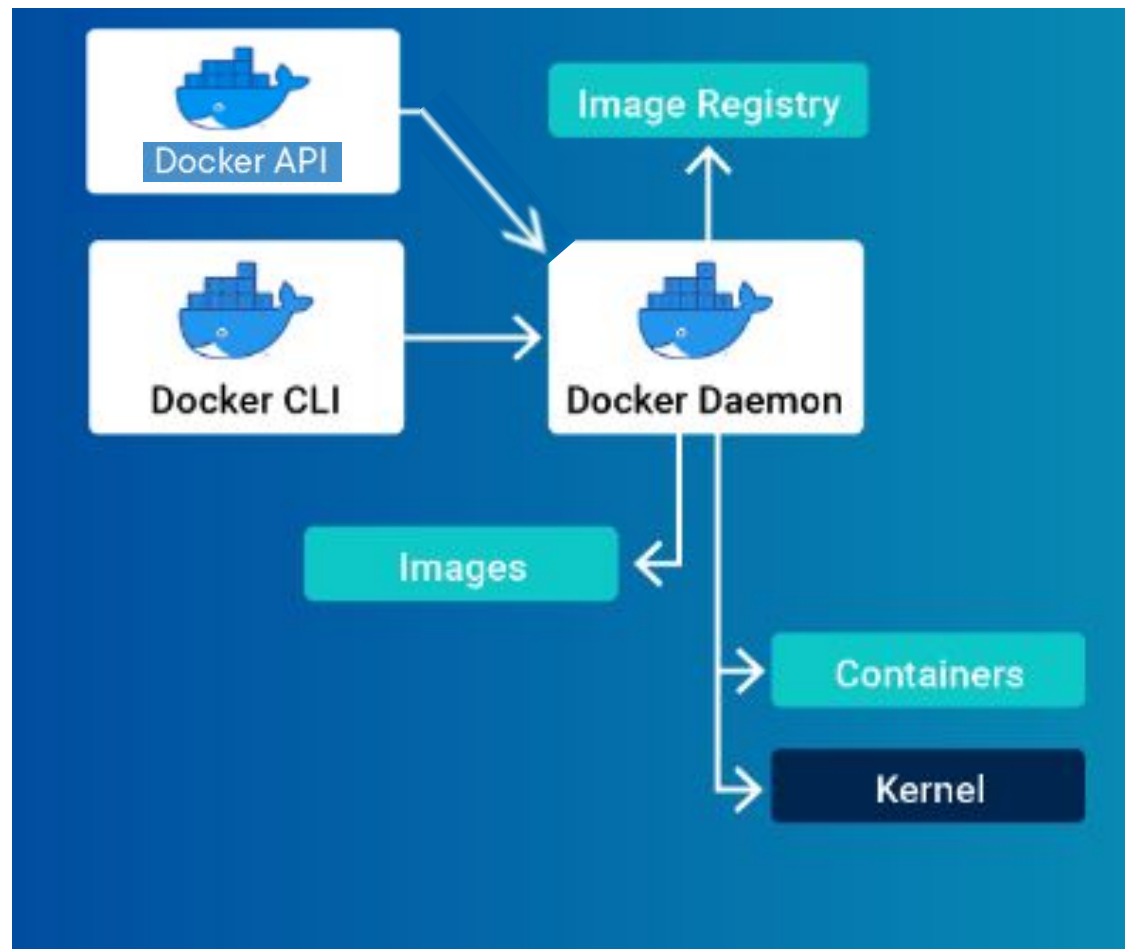
Montagem: Carlos Alberto da Silva, 2023.

Virtualização por *Container*

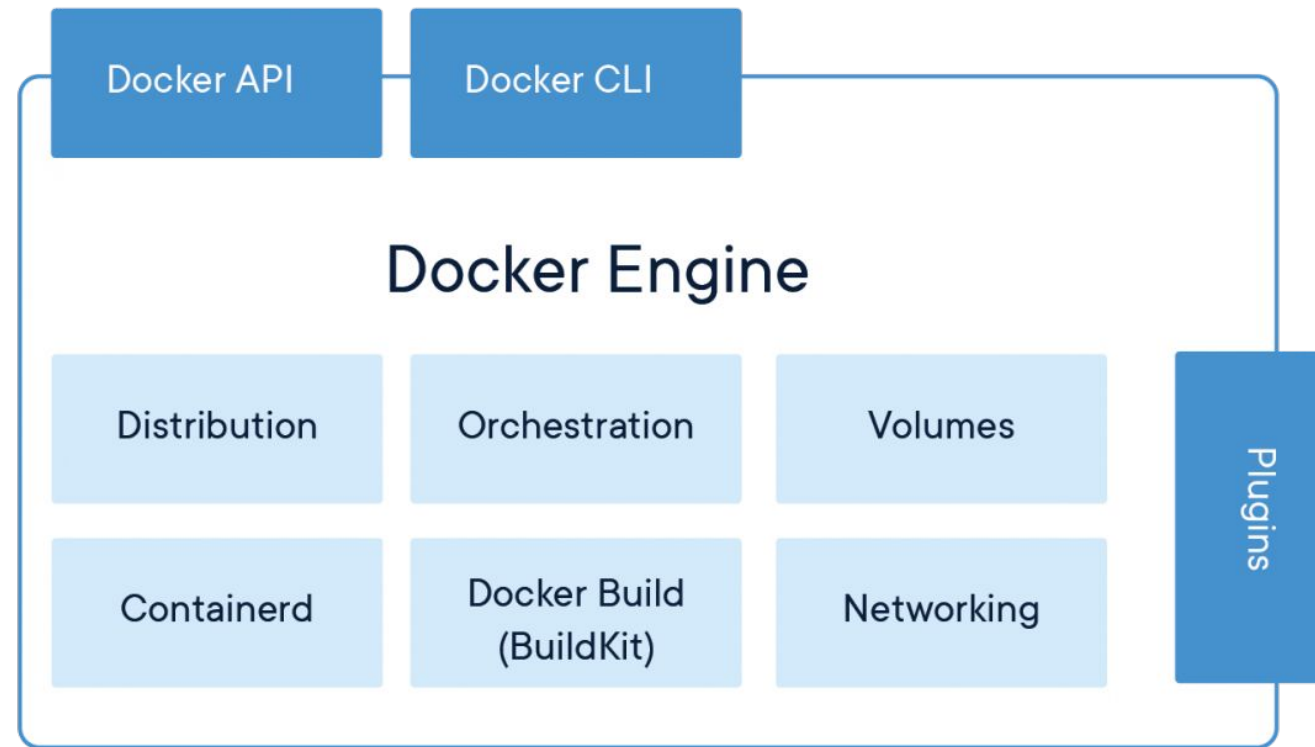


Fonte: Viatlino, 2016, p. 15 (plataforma de leitura).

Visão geral



Visão geral

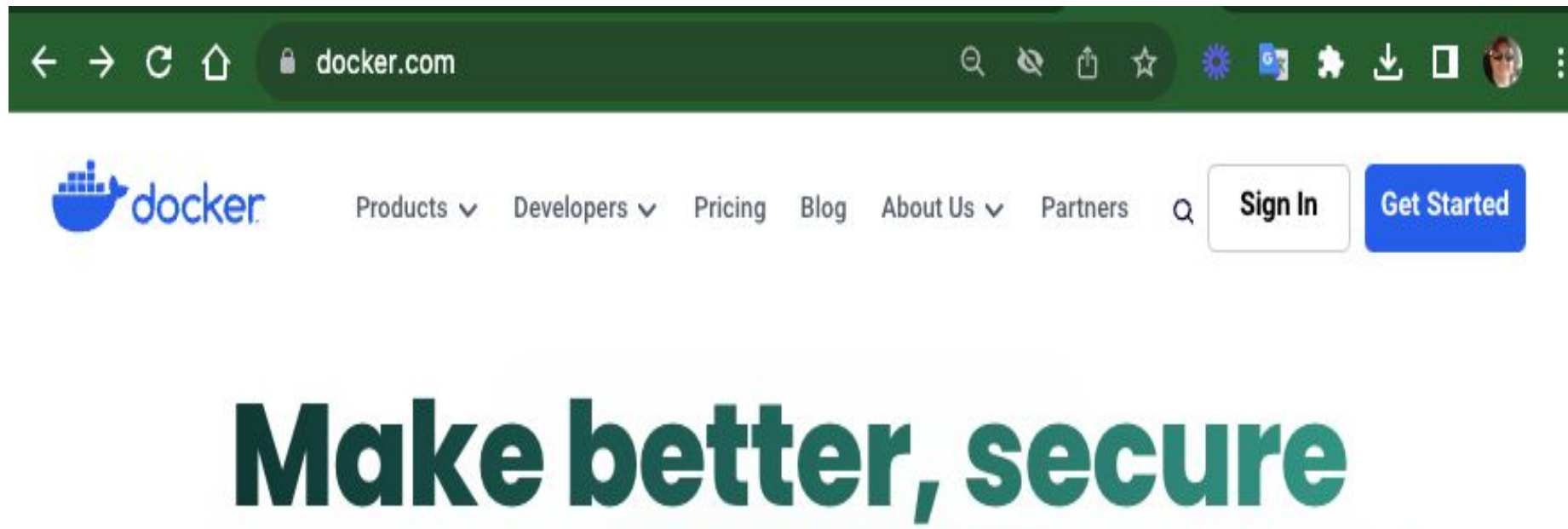


Fonte: www.docker.com.

Docker®

Docs (manuais)

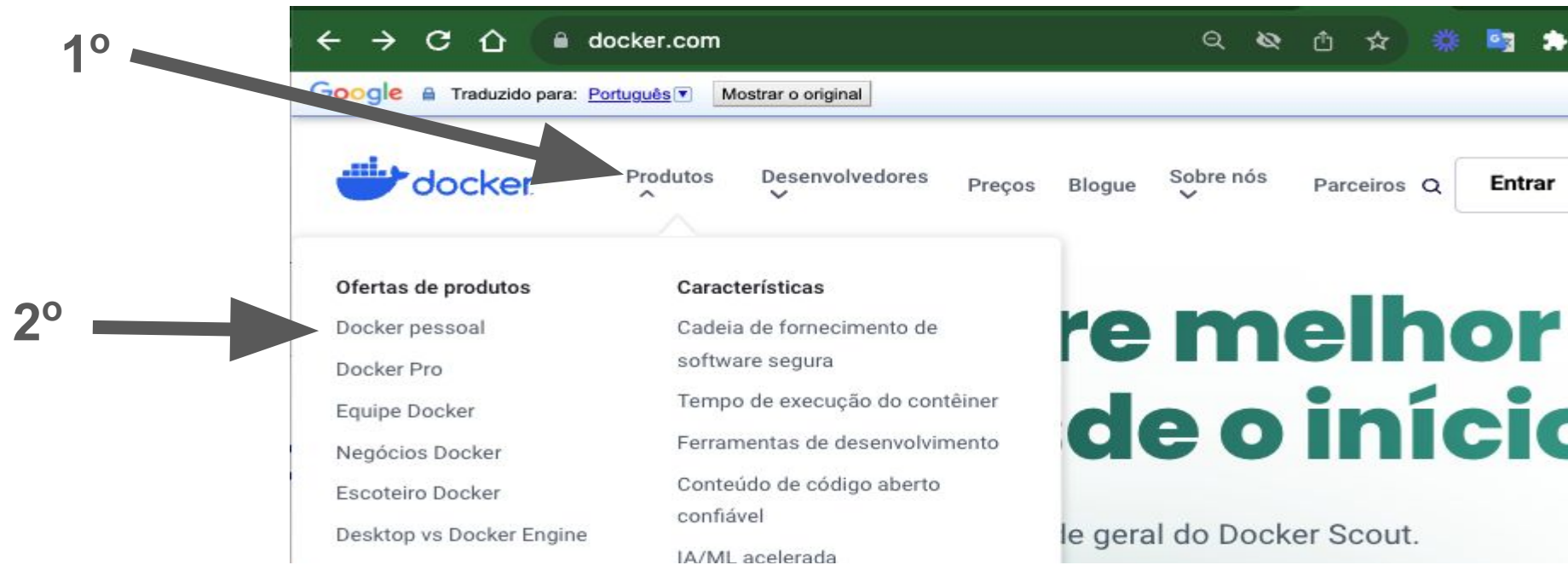
Site www.Docker.com



Fonte: www.docker.com

Docs (manual)

Site www.Docker.com (site traduzido)



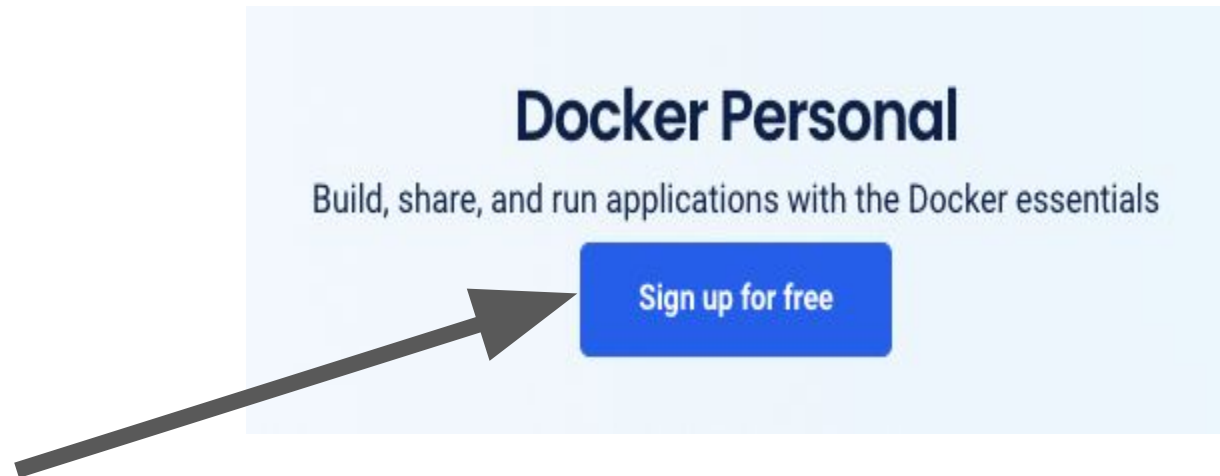
Fonte: www.docker.com

Docker®

Docs (manual)

Versão

Docker Personal®



Fonte:

<https://www.docker.com/products/personal/>

Docker®

Docs (manual)

Versão

Docker Personal®

Criar uma conta para acesso




AGEAD
Agência de Educação
Digital e a Distância



Create your account

 Continue with Google

 Continue with GitHub

OR

Email

Username

Password



☐ Send me occasional product updates and announcements.

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

Sign up

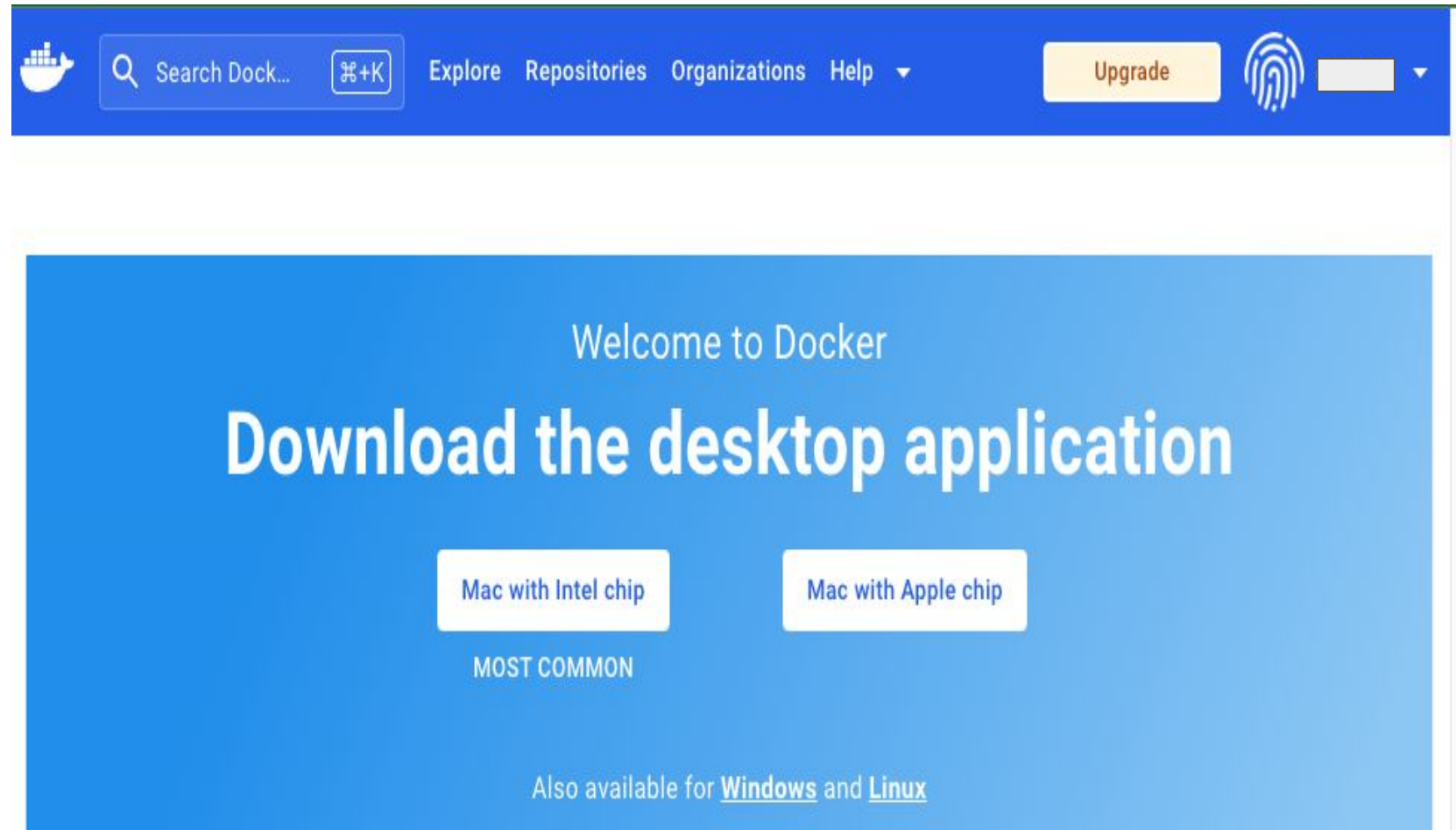
Fonte: <https://hub.docker.com/signup?redirectTo=/subscription%3Fplan%3Dfree>

Docker®

Docs (manual)

Acesso

Docker Personal®

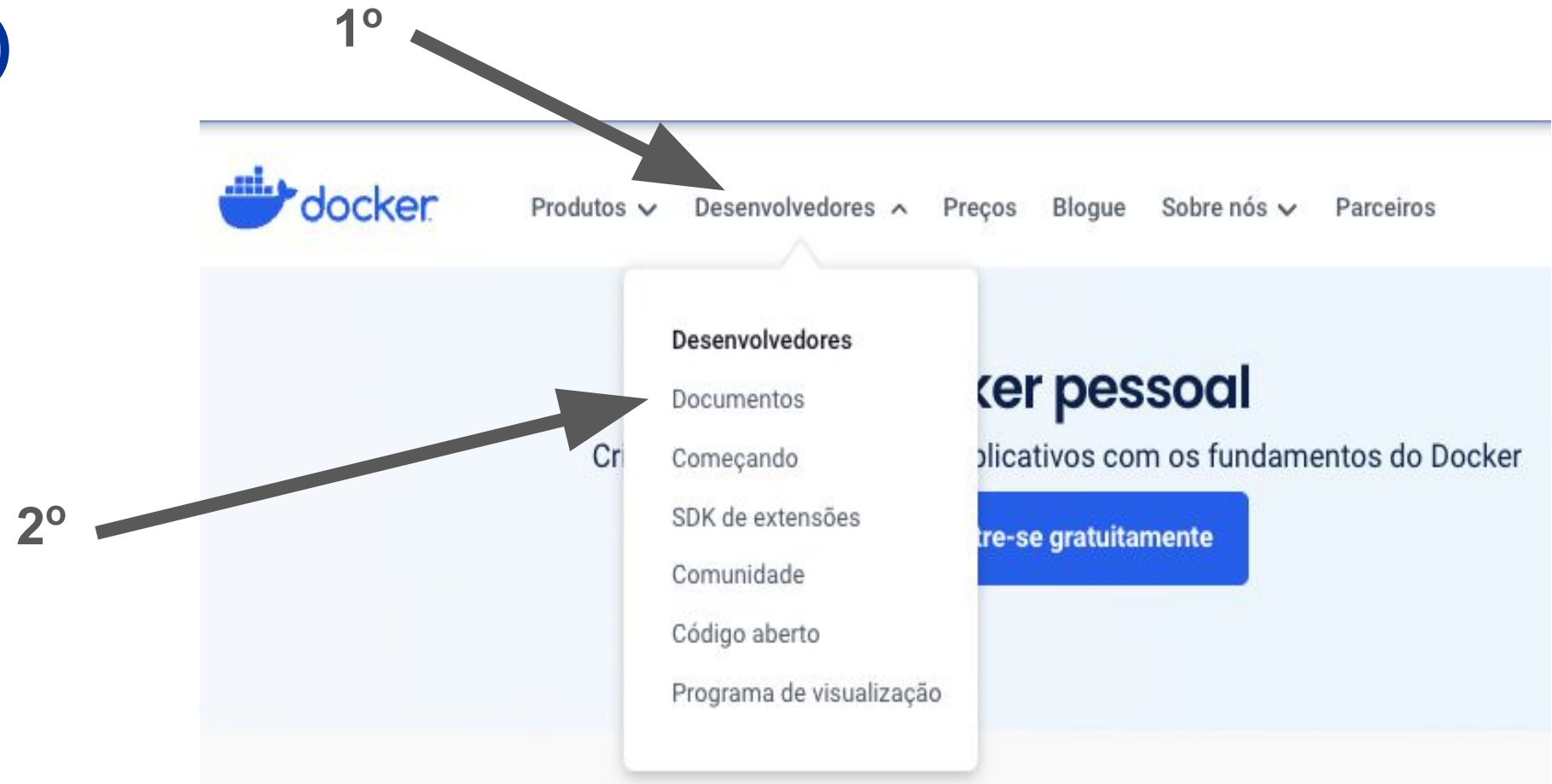


Fonte: <https://hub.docker.com/>

Docker®

Docs (manual)

Acesso Documentos



Fonte: <https://www.docker.com/products/personal/>



Docs (manual)

Acesso Documentos



AGEAD
Agência de Educação
Digital e a Distância



O que podemos ajudá-lo a encontrar?



iniciar

Aprenda os conceitos básicos do Docker e os benefícios da containerização de seus aplicativos.



Baixar e instalar

Baixe e instale o Docker em sua máquina em algumas etapas fáceis.



Guias

Aprenda como configurar seu ambiente Docker e começar a containerizar seus aplicativos.



Guias específicos de idioma

Aprenda como usar o Docker com sua linguagem de programação favorita.



Manuais

Navegue pelos manuais e aprenda como usar os produtos Docker.



Referência

Navegue pela documentação de referência da CLI e da API.

Fonte: <https://docs.docker.com/>

Arquitetura *Docker*[®]

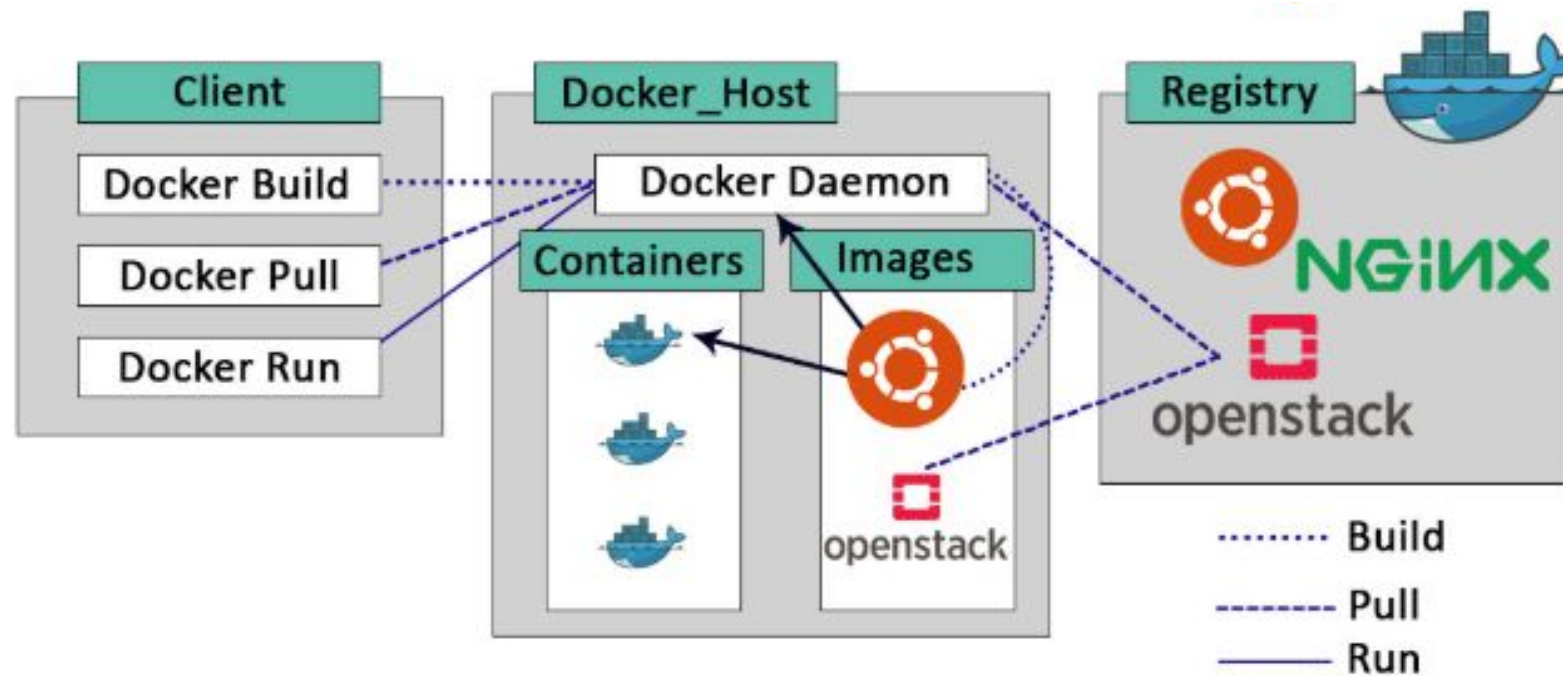
Arquitetura *Docker*[®]

Arquitetura cliente-servidor



Arquitetura *Docker*®

Client executa comandos



Fonte: <https://docs.docker.com/>

Client docker e daemon docker se comunicam por meio de:

- *REST API*;
- *Sockets*;
- *Outros*.

Arquitetura *Docker*[®]

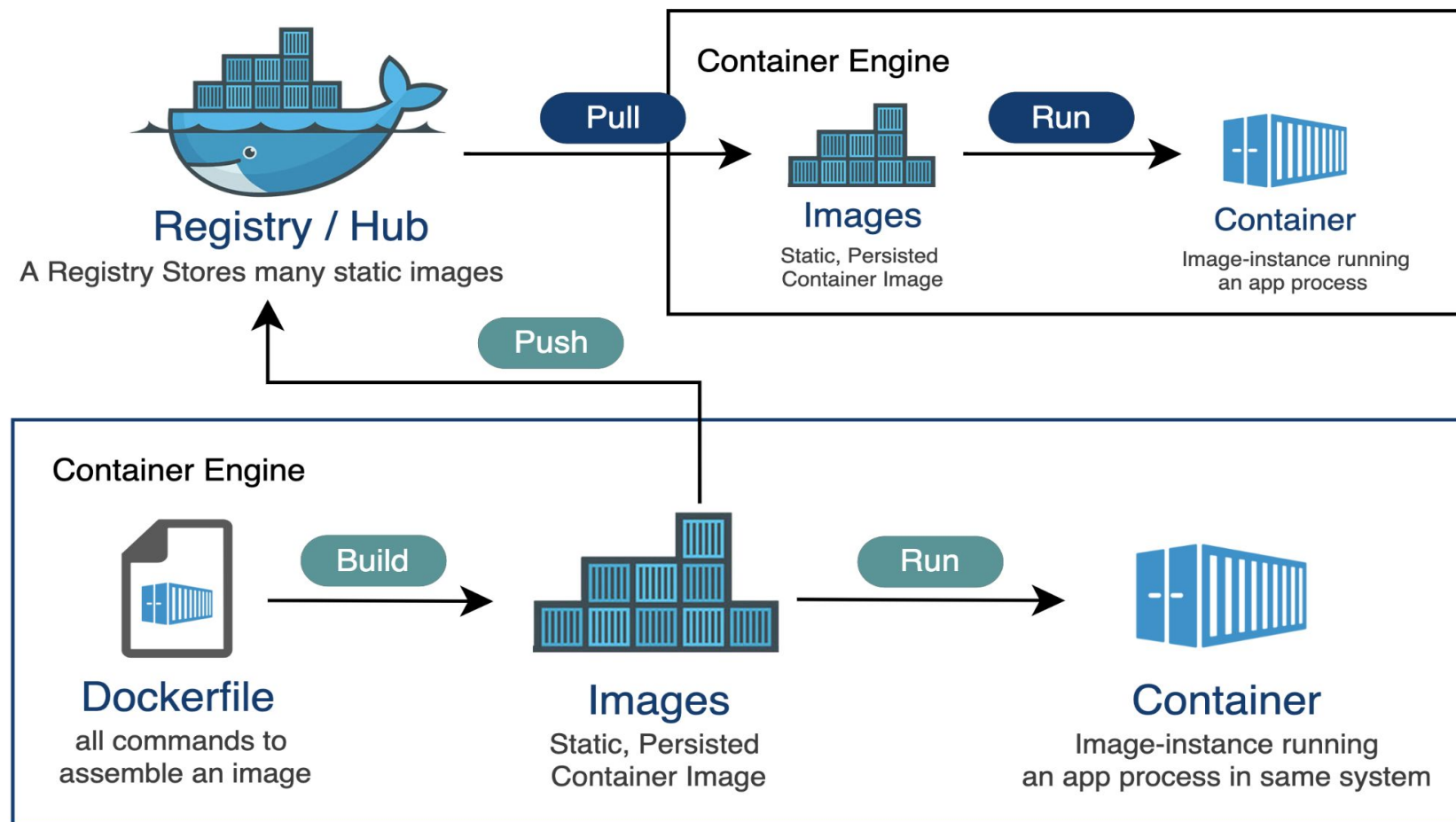
Componentes *Docker*[®]

- *Container engine*[®]
- *Container docker*[®]
- *Image docker*[®]
- *Dockerfile*[®]
- *Docker Registry*[®]
- *Docker Hub*[®]

Arquitetura Docker®

Componentes Docker®

Container engine®



Arquitetura *Docker*[®]

Componentes *Docker*[®]

Container docker[®]

- Como base no *image docker*:
 - ***Image*** é uma classe java;
 - ***Container*** é uma instância desta classe.

Arquitetura *Docker*[®]

Componentes *Docker*[®]

Image docker

- ***Templates*** para criação de ***container***;
- São imutáveis, pois para executá-las é necessário criar um instância da mesma (*container*).

Arquitetura *Docker*[®]

Componentes *Docker*[®]

Image docker

- São construídas em camadas:
 - Facilita a reutilização e manutenção.

⇒ Ambiente encapsulado pronto para ser replicado.

Arquitetura *Docker*[®]

Componentes *Docker*[®]

Dockerfile[®]

- São ***scripts*** com comandos para:
 - Criar imagens;
 - Executar comandos;
 - Criar variáveis de ambiente;
 - Copiar arquivos e pastas do *host* para a imagem.

Arquitetura *Docker*[®]

Componentes *Docker*[®]

Dockerfile[®]



Fonte: www.docker.com.

Arquitetura *Docker*[®]

Componentes *Docker*[®]

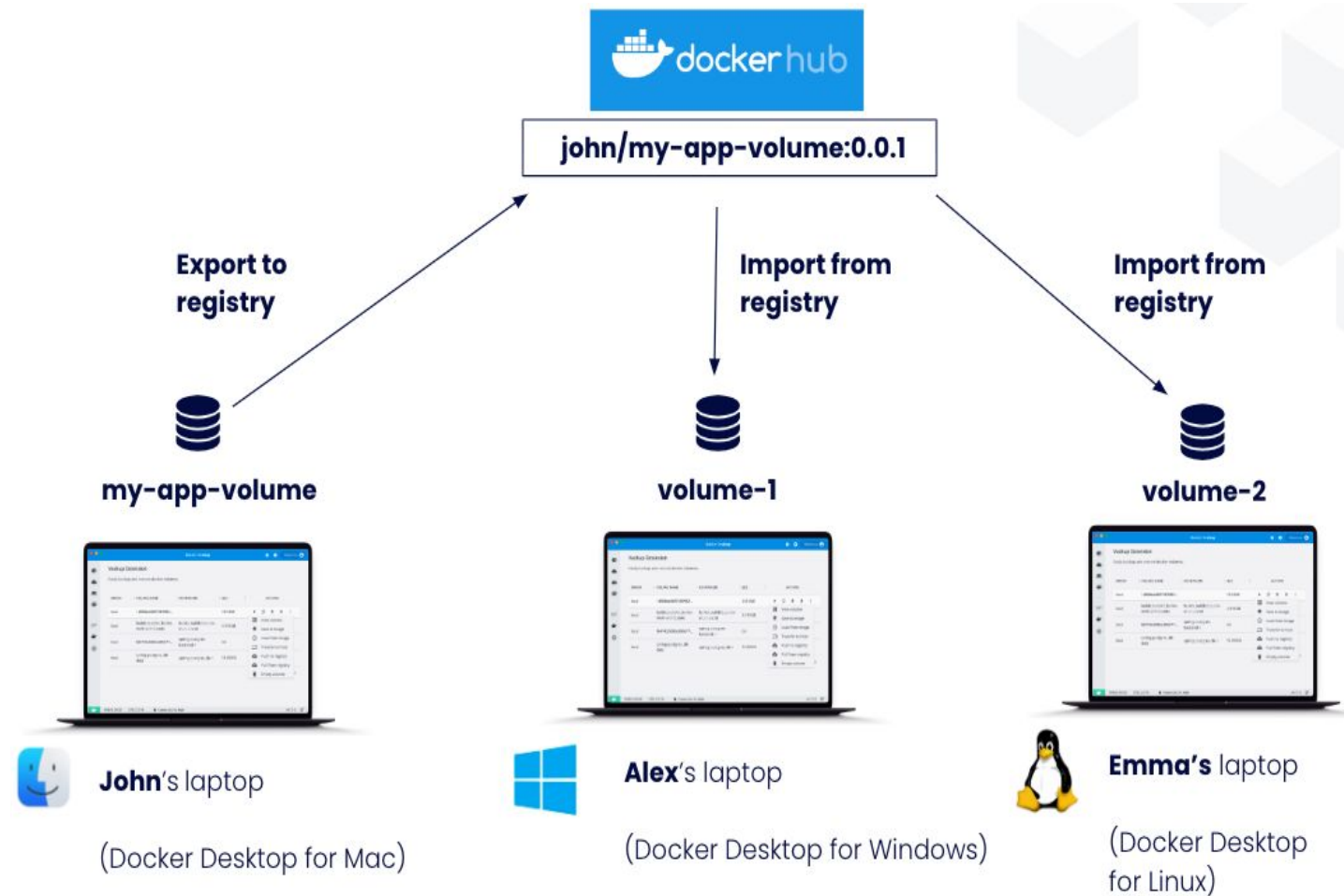
Docker Registry[®]

- Um repositório **Git**:
 - Imagens podem ser versionadas.
- ***Docker Hub*[®]** é o repositório oficial do ***Docker*[®]**:
 - Permite versionar e hospedar imagens privadas e públicas.

Arquitetura Docker®

Componentes do Docker®

Docker Hub®



Plataforma *Docker*[®]

Plataforma *Docker*[®]

Plataforma composta por:

Docker Engine[®] + *Docker Hub*[®]

Docker Engine[®]

Docker Daemon[®] + (*Docker CLI*[®] e/ou *Docker API*[®])

Plataforma *Docker*®

Docker Desktop®

*Disponível para
os sistemas
operacionais*

(seu Docker engine)



Docker Desktop for Mac

A native application using the macOS sandbox security model that delivers all Docker tools to your Mac.



Docker Desktop for Windows

A native Windows application that delivers all Docker tools to your Windows computer.



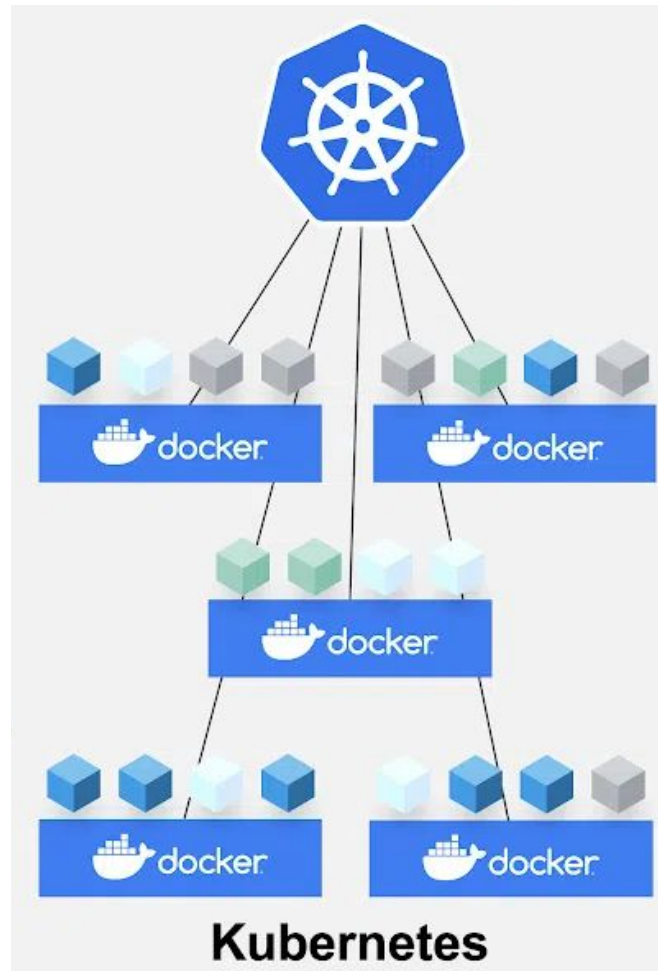
Docker Desktop for Linux

A native Linux application that delivers all Docker tools to your Linux computer.

Plataforma *Docker*[®]

Kubernetes[®]

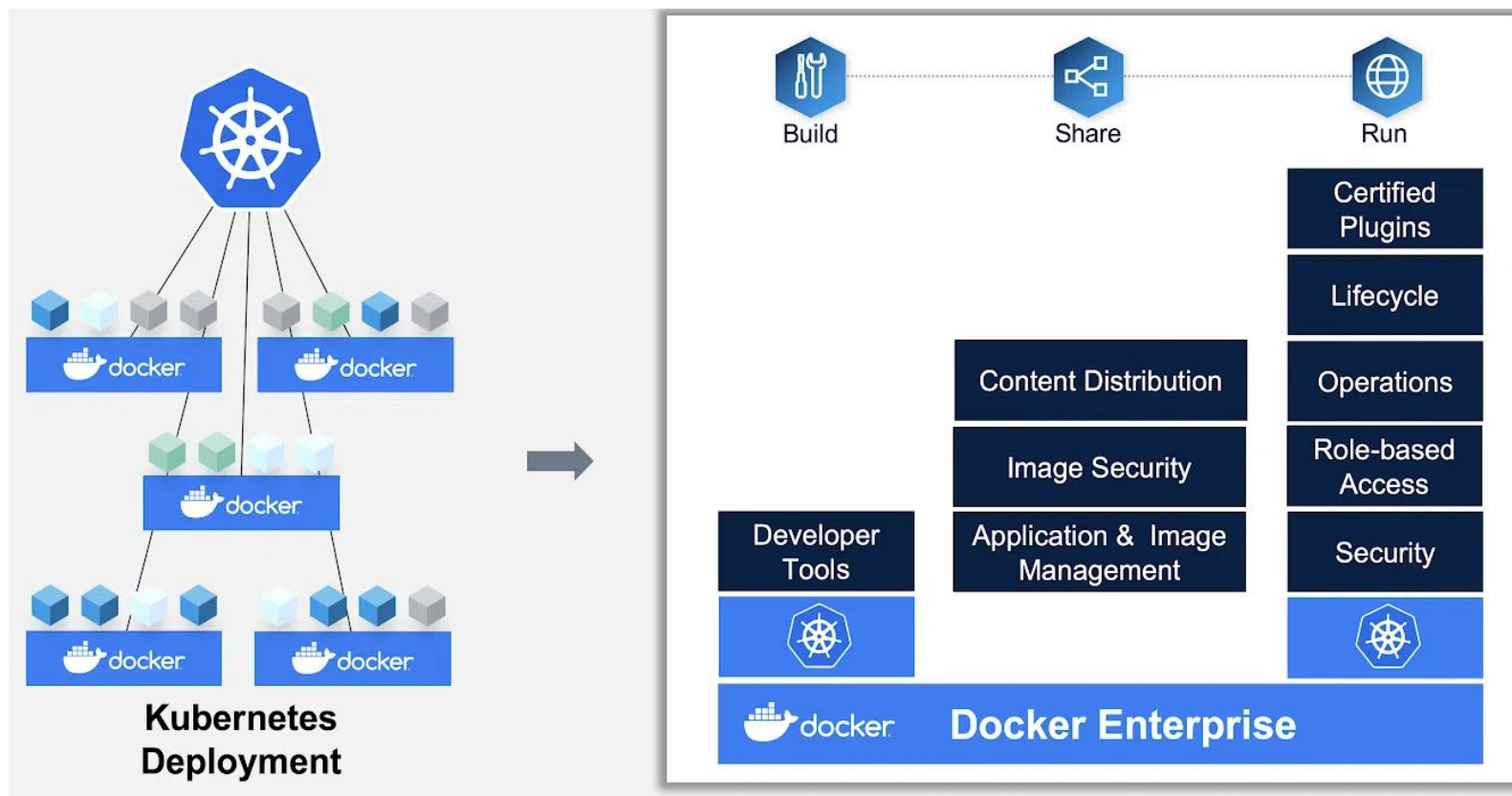
Gerenciamento de vários *containers* *Docker*[®]



Fonte: <https://docs.docker.com/>

Plataforma *Docker*®

Enterprise Docker®



Fonte: <https://docs.docker.com/>

Referências

DOCKER. [Online]. Disponível em: <https://www.docker.com/>. Acessado em: 18 de novembro de 2023.

KUBERNETES. [Online]. Disponível em: <https://kubernetes.io/pt-br/>. Acessado em: 18 de novembro de 2023.

VITALINO, Jeferson Fernando Noronha; CASTRO, Marcus André Nunes. **Descomplicando o Docker**. 1a edição. Rio de Janeiro: Brasport, 2016. Capítulo 1 e 2. ISBN 9788574528007. [Disponível na Biblioteca Digital da UFMS](#).

Licenciamento



Respeitadas as formas de citação formal de autores de acordo com as normas da ABNT NBR 6023 (2018), a não ser que esteja indicado de outra forma, todo material desta apresentação está licenciado sob uma [Licença Creative Commons - Atribuição 4.0 Internacional](https://creativecommons.org/licenses/by/4.0/).