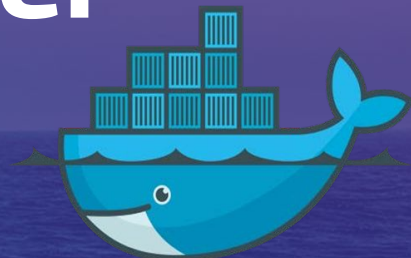




Next Generation Cloud Services

Introdução ao Docker



Objetivos:

- **Ao final desse treinamento, você será capaz de:**
 - O que é o Docker e como ele é usado;
 - Como:
 - criar container utilizando imagens;
 - construir imagens utilizando Dockerfile;
 - trabalhar com volumes/armazenamento;
 - acontece a comunicação entre contêineres.

Agenda:

1. **Motivação para uso de Contêineres;**
2. **O que é o Docker?**
3. **O que é Container?**
4. **O que é imagem de Container?**
5. **Docker: Instalação e primeiros passos;**

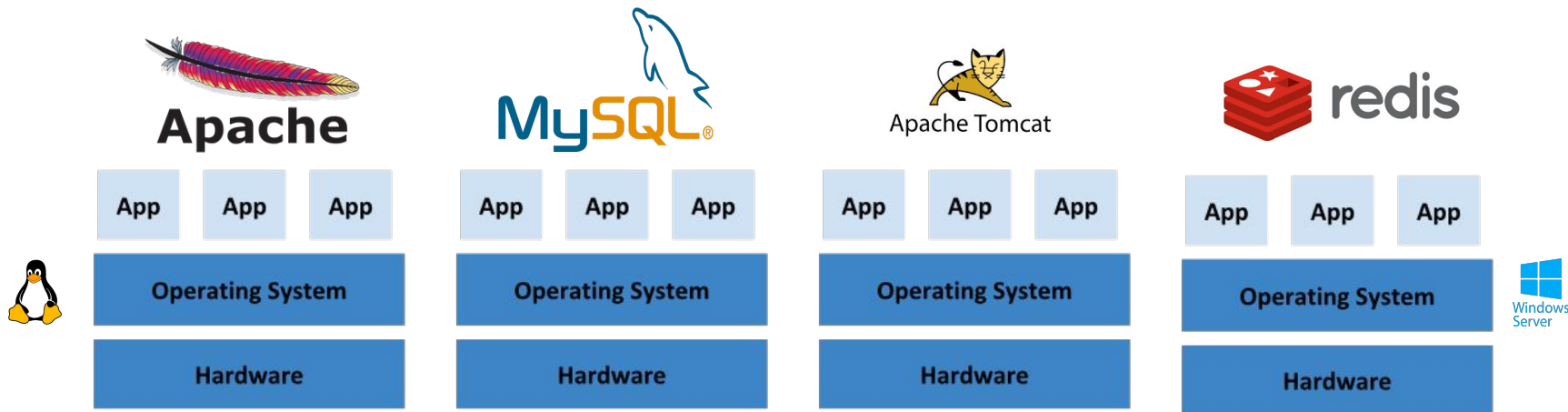


1. Motivação para o uso de Containers.

Como surgiram os containers?

Era da implantação tradicional

- Para cada serviço, utilizávamos uma máquina/servidor.



Traditional Deployment Traditional Deployment Traditional Deployment Traditional Deployment

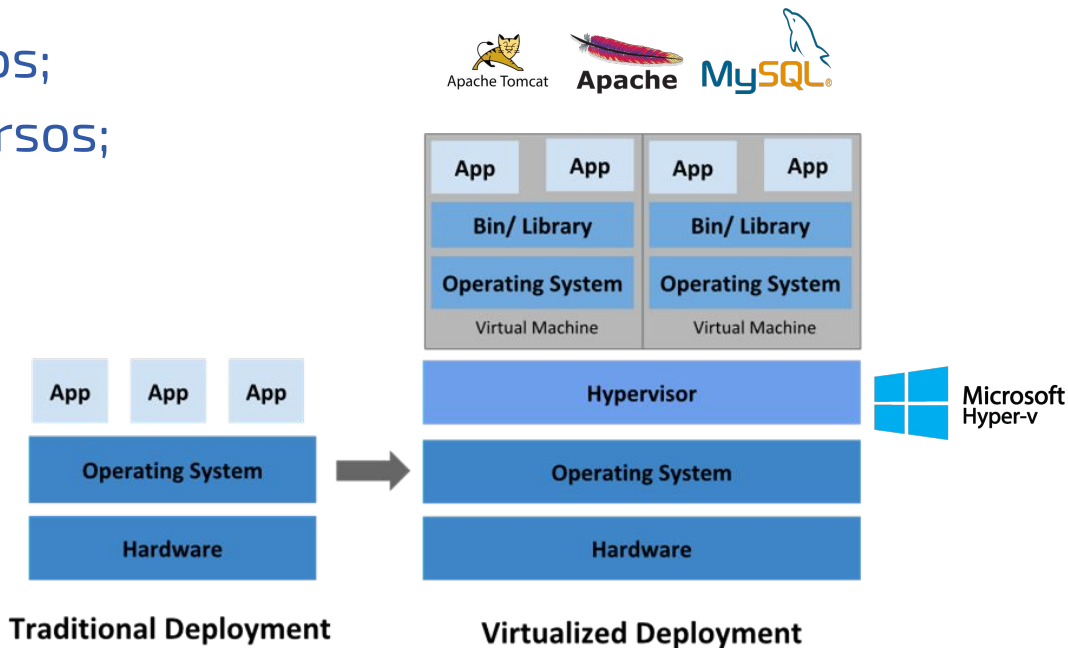
Fonte: <https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/>

Era da implantação tradicional

- Problemas:
 - Custo Elevado
 - Energia;
 - Licenças;
 - Profissionais;
 - Capacidade subutilizada
 - Poucos momentos do ano com alta demanda
 - Exemplos: Black Friday, Matrículas faculdade.

Era da implantação virtualizada

- Virtualizar os serviços;
Otimização dos recursos;

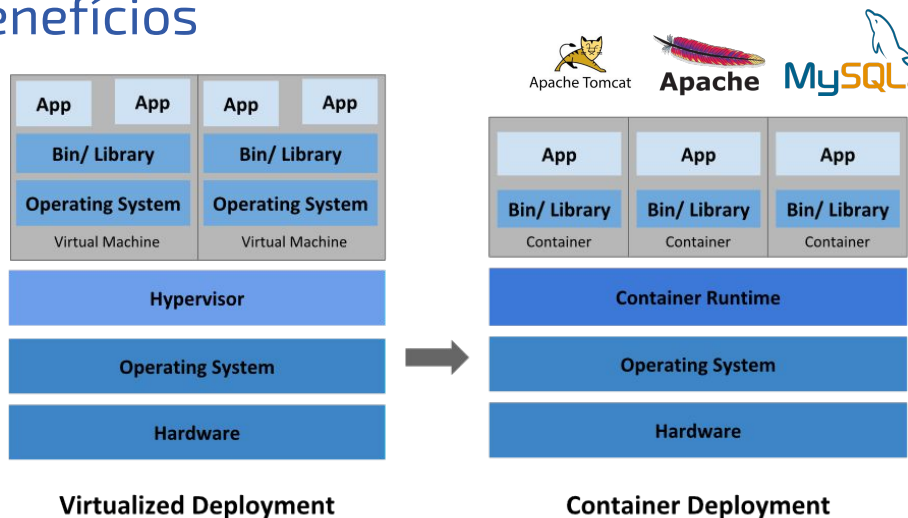


Era da implantação virtualizada

- Vantagens:
 - Melhor utilização de recursos;
 - Escalabilidade;
 - Redução de custos de hardware;
- Desvantagens:
 - Custos com SO
 - Software de virtualização (hypervisor)

Era da implantação em containers

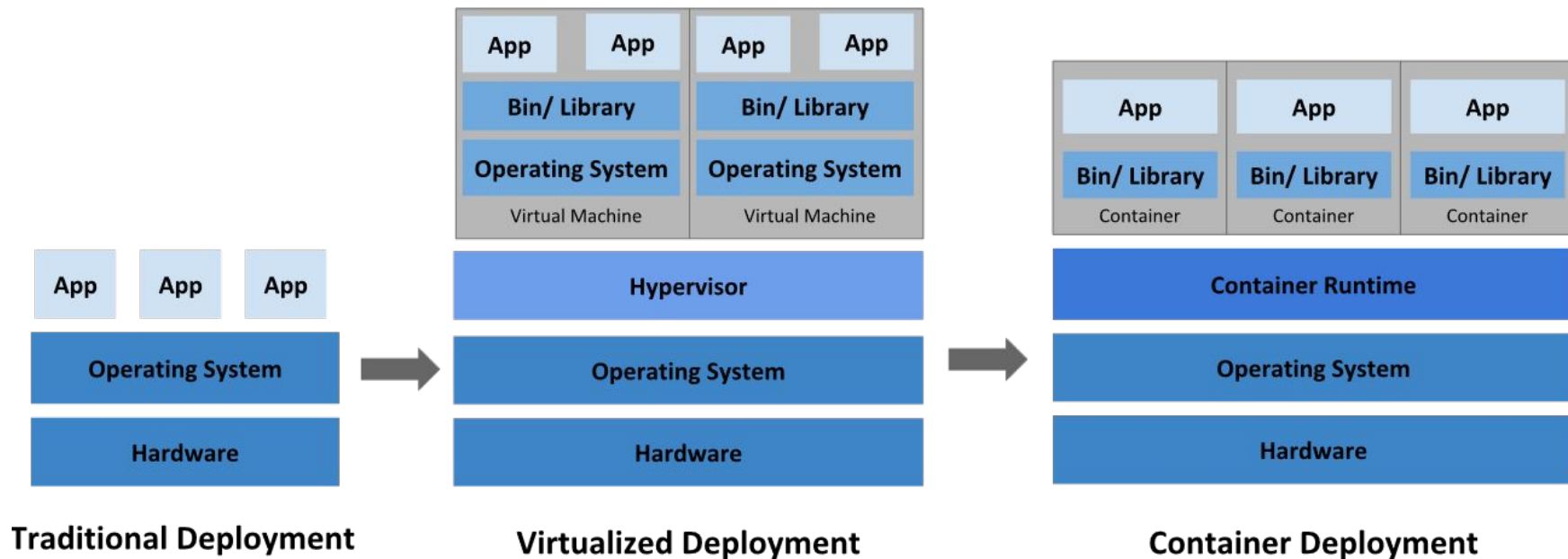
- Semelhante as VMs, mas com diversos benefícios extras;



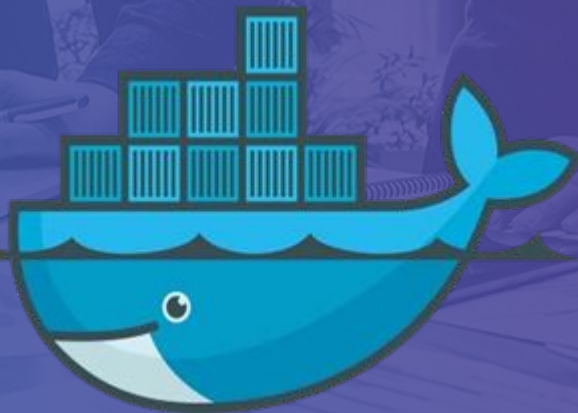
Era da implantação em containers

- **Portabilidade de distribuição** de nuvem e sistema operacional: executa no Ubuntu, RHEL, CoreOS, localmente, nas principais nuvens públicas e em qualquer outro lugar.
- **Utilização de recursos:** alta eficiência e densidade.
- Conheça outras vantagens [clicando aqui](#).

Evolução:



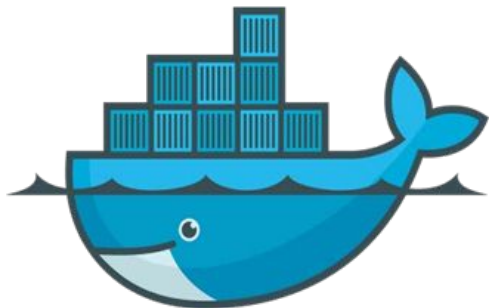
2. O que é o Docker?



O que é Docker?

- Docker é uma plataforma aberta para desenvolvimento, envio e execução de aplicativos.
- É uma ferramenta que se apoia em recursos existentes no kernel, inicialmente Linux, para isolar a execução de processos.
- As ferramentas que o Docker traz são basicamente uma camada de administração de containers, baseado originalmente no [LXC](#).

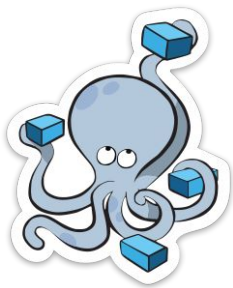
O que é Docker?



- Plataforma leve, aberta e segura
- Simplifique a construção, envio e execução de aplicativos
- Funciona nativamente em Linux ou Windows Server
- Baseia-se em "imagens" e "contêineres"

Docker - Tecnologias

Docker Engine: O Docker Engine é uma tecnologia de containerização de código aberto para construir e containerizar seus aplicativos.



Docker Compose: O Compose é uma ferramenta para definir e executar aplicativos Docker de vários contêineres.

Docker - Tecnologias



Docker Hub: O Docker Hub é um serviço fornecido pelo Docker para localizar e compartilhar imagens de contêiner com sua equipe.

- **Private registries:** Docker Trusted Registry, Harbir, GitLab Container Registry;
- **Public registries:** Google Container Registry (CGR), Amazon Elastic Container Registry (ECR), Azure Container Registry (ACR).

Docker Swarm

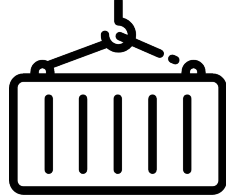


- O Docker Swarm é uma ferramenta nativa de orquestração do Docker.
 - O **Docker Swarm** é uma ferramenta de orquestração leve e fácil de usar com ofertas limitadas em comparação com o Kubernetes.
 - O **Kubernetes** é complexo, mas poderoso e fornece recursos de autocorreção e dimensionamento automáticos prontos para uso.



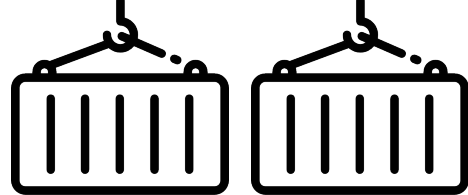
3. O que é o Container?





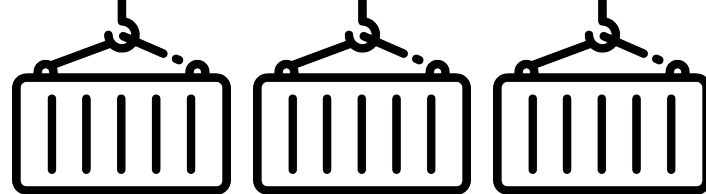
O que é Container?

- Um contêiner é um processo em área restrita em sua máquina que é isolado de todos os outros processos na máquina host. Esse isolamento aproveita **namespaces** e **cgroups** do kernel , recursos que estão no Linux há muito tempo.



O que é Container?

- “Um contêiner é uma instância executável de uma imagem.”
- pode ser executado em **máquinas locais, máquinas virtuais** ou **implantado na nuvem**.
- é **portátil** (pode ser executado em qualquer sistema operacional)
- Os **contêineres são isolados uns dos outros** e executam seus próprios softwares, binários e configurações.



O que é Container?

- Um container será um ambiente de virtualização
 - Com menos consumo de processamento,
 - Com menor custo de manutenção/configuração,
 - Com menor tempo de inicialização/finalização.

Container Docker não são VMs

- Mais leves e integrados ao SO da máquina host
- Compartilham seu Kernel
- Benefícios diferentes



Maquina Virtual



Contêineres

4. O que é Imagem de Container?

O que é **Imagem** de container?

- “Uma imagem é um modelo somente leitura com instruções para criar um contêiner do Docker. Frequentemente, uma imagem é baseada em outra imagem, com alguma personalização adicional.”
- As imagens de container agrupam um programa e suas dependências em um único artefato em um sistema de arquivo raiz.

O que é **Imagem** de container?

- Uma imagem de container é um pacote binário que encapsula todos os arquivos necessários para executar um programa em um contêiner do SO.
- Uma imagem Docker é a materialização de um modelo de um sistema de arquivos, modelo este produzido através de um **processo chamado build**.
- Esta **imagem é representada por um ou mais arquivos** e pode ser **armazenada em um repositório**.



Docker Image

Exemplo: Ubuntu com Node.js e
Application Code



Docker Container

Criado usando uma imagem.
Rodando sua aplicação

Docker File System e Camadas

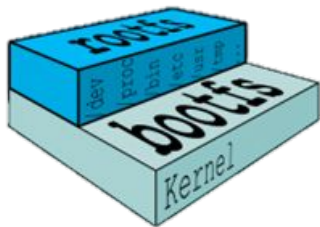
Docker File Systems

- O Docker utiliza file systems especiais para otimizar o uso, transferência e armazenamento das imagens, containers e volumes.
- O principal é o AuFS, que armazena os dados em camadas sobrepostas, e somente a camada mais recente é gravável.
- [Saiba mais.](#)

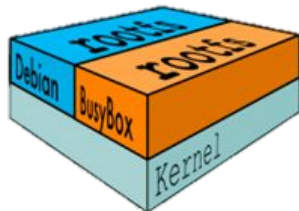
COW (Copy-On-Write)

- O sistema de arquivos do Docker é chamado de **layered**;
 - O bootfs em vez de ser único por sistema/container, é compartilhado entre eles;
 - O rootfs é isolado por camadas;
- AuFS, que monta uma camada de leitura/escrita em cima do filesystem em somente leitura.
 - Copy-on-Write

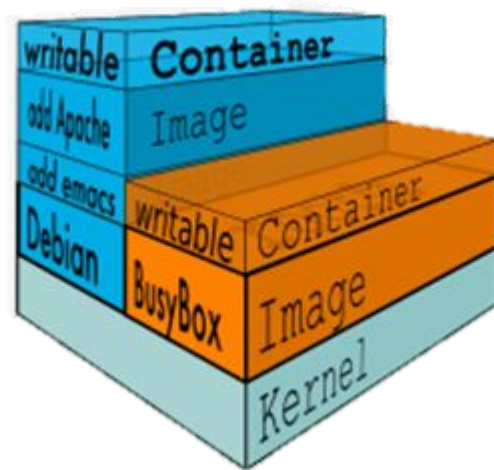
COW (Copy-On-Write)



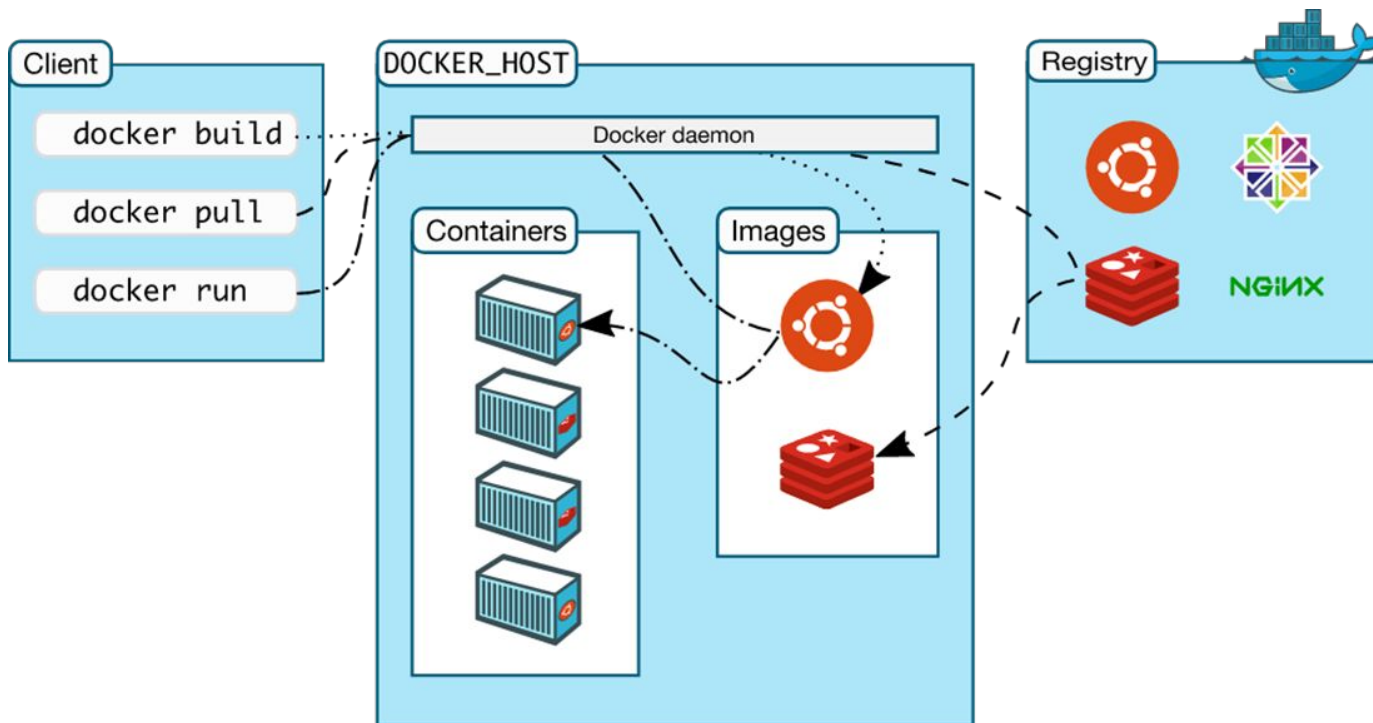
Sistema de arquivo
comum



Filesystem no
docker



Arquitetura do Docker



Docker Certified Associate



- Orchestration (25% of exam)
- Installation and Configuration (15% of exam)
- Image Creation, Management, and Registry (20% of exam)
- Storage and Volumes (10% of exam)
- Networking (15% of exam)
- Security (15% of exam)

5. Docker Instalação e Primeiros Passos

Instalação do Docker:

- Windows ou Mac: Docker Desktop.
 - [Documentação oficial](#)
- Linux:
 - [Documentação oficial](#)
- Simuladores/Emuladores:
 - [Play With Docker](#)
 - [Katacoda](#)
-

CheatSheet:

- [Dockerlabs](#)
- [Linuxtips](#)

Primeiros Exemplos:

<https://github.com/patrickjcardoso/Docker>

**Next
Generation
Cloud
Services**

OBRIGADO!

www.o2b.com.br | contato@o2b.com.br | +55 11 5180-2100