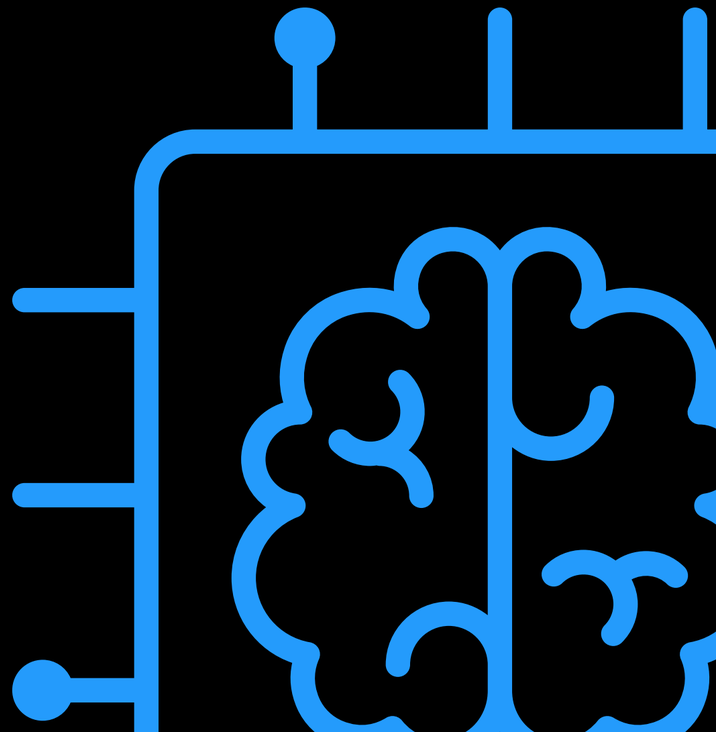
MLOps: Machine Learning Operations

CEIA
CENTRO DE EXCELÊNCIA EM INTELIGÊNCIA ARTIFICAL

AKCIT
CENTRO DE COMPETÊNCIA EMBRAPII EM TECNOLOGIAS IMERSIVAS



Quem sou

Técnico em Eletrônica - Instituto Federal de Goiás

- Projetos de extensão: Lógica de programação em escolas da rede metropolitana
- Projetos de ensino: IoT para estudantes da graduação e técnicos do IFG
- Projetos de Pesquisa: Atuação em laboratório de inovação
- Desenvolvedor: Startup para monitoramento de placas solares com ML

Inteligência Artificial - Universidade Federal de Goiás

- Atuação no Centro de Excelência em IA (CEIA)
- Desenvolvimento Fullstack
- **Infraestrutura e DevOPs: AWS e GCP**

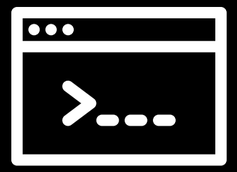
Tecnologia

- **Tecnologia:** o estudo da **técnica**
- Não se trata apenas da tecnologia da informação
- Transformações de **hábitos** também são necessários para que uma nova técnica tenha efeito
- **Cultura:** etimologia parecida com “cultivar”

Vaso Jomon

DevOps

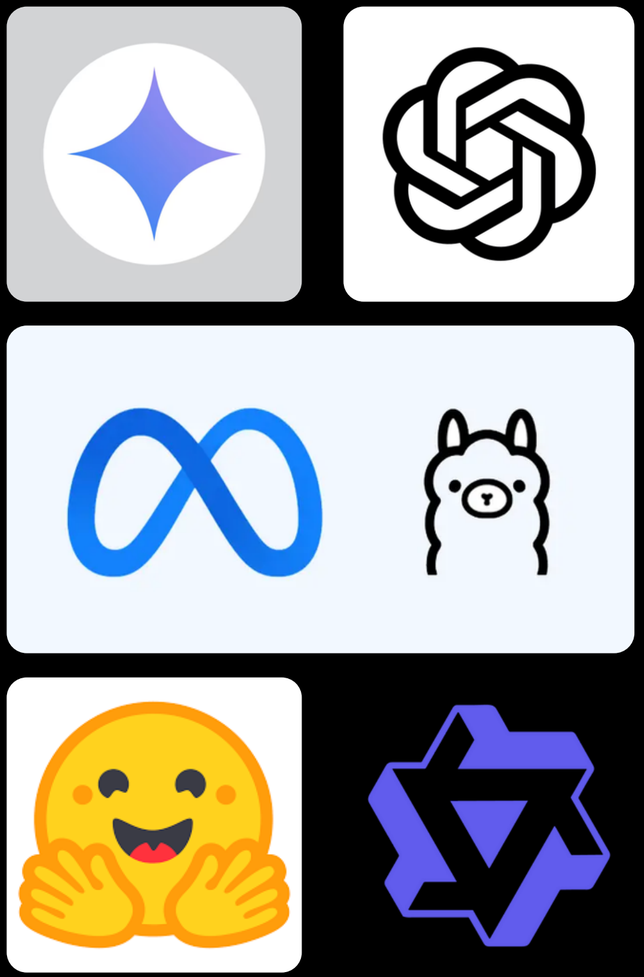
- “Development Operations”
- **Cultura** de cooperação entre times (não só do de desenvolvimento) visando entrega de valor contínua para o cliente



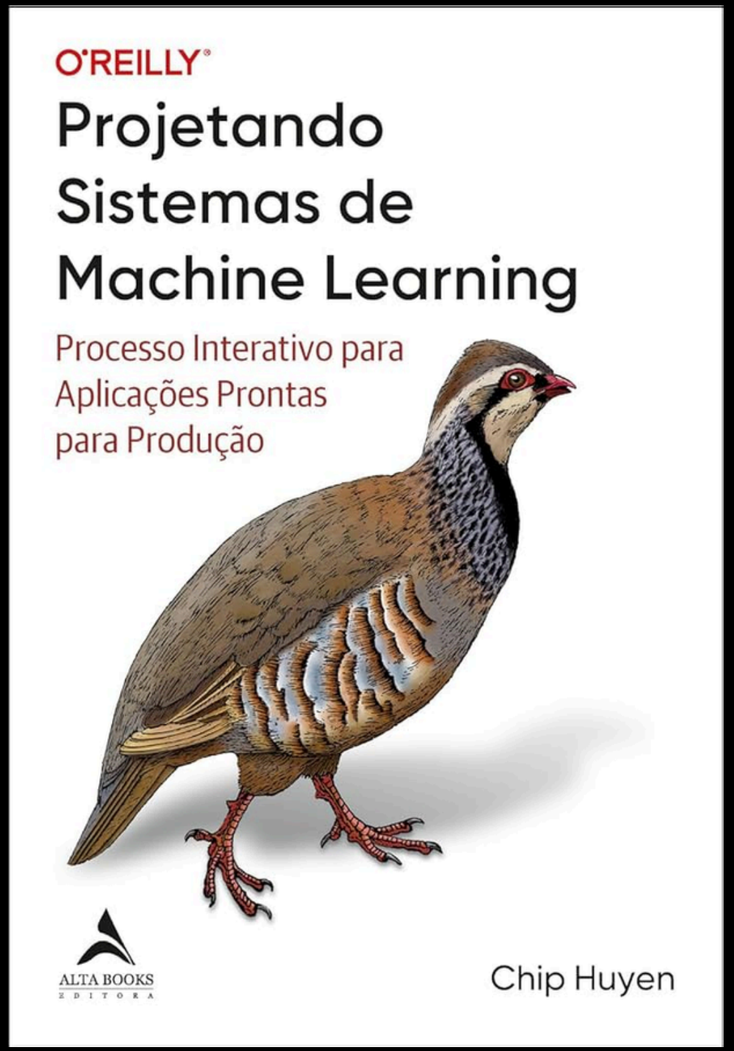

MLOps

- **Machine Learning Operations**
- A maioria dos modelos em produção não são tão “grandes” dentro da escala de modelos de linguagem que temos no mercado hoje
- **LLMOps: Large Language Models Operations**



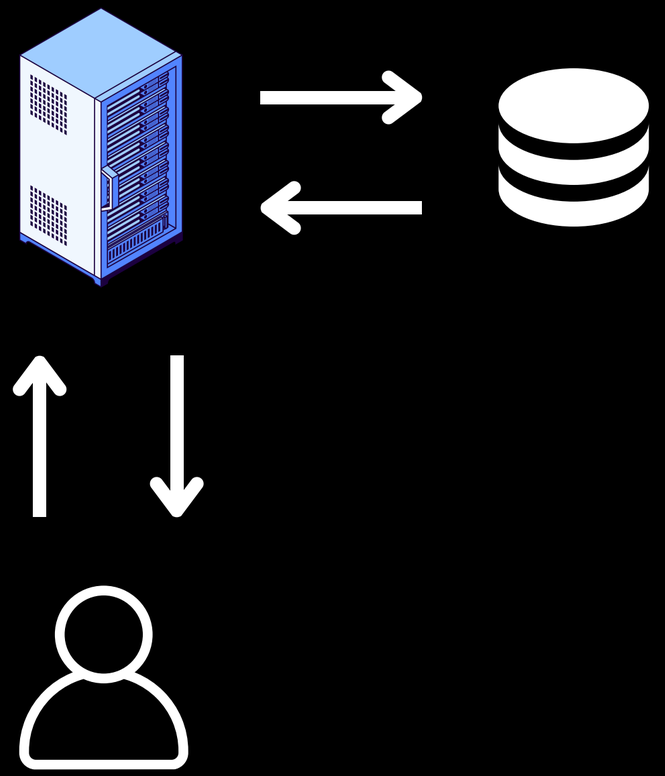
MLOps

- Na realidade, alguns sistemas chegam a ter centenas de modelos orquestrados de uma única vez
- Adaptação para regiões, épocas do ano, datas comemorativas, horários do dia, etc.
- Requisitam monitoramente contínuo, automatizado e com métricas de qualidade confiáveis



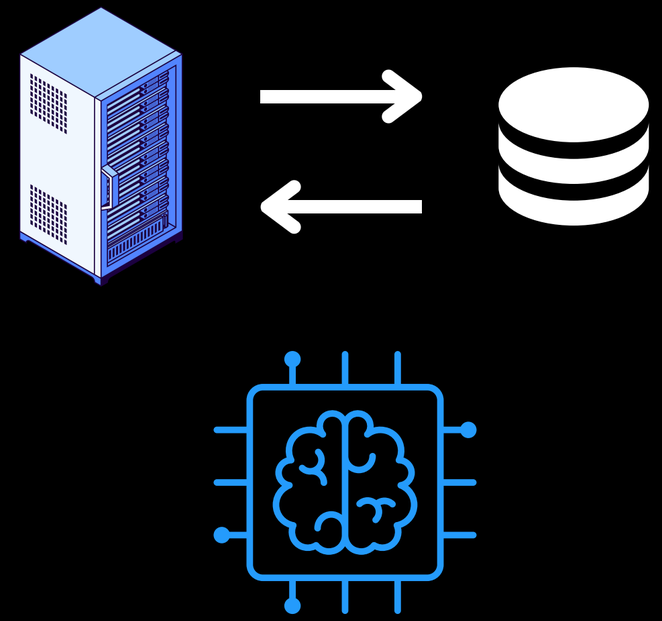
MLOps: Dificuldades

- REST APIs possuem um modelo de desenvolvimento, testes e implementação bem consolidados
- Realizar CRUD em bancos de dados se tornou uma tarefa mais simples na maioria dos casos com o uso de tecnologias como ORMs
- Modelos tradicionais de APIs e aplicações se tornaram simples a nível computacional e de implementação
- Artefatos de código são aproveitados independente dos dados no banco ou do SGBD



MLOps: Dificuldades

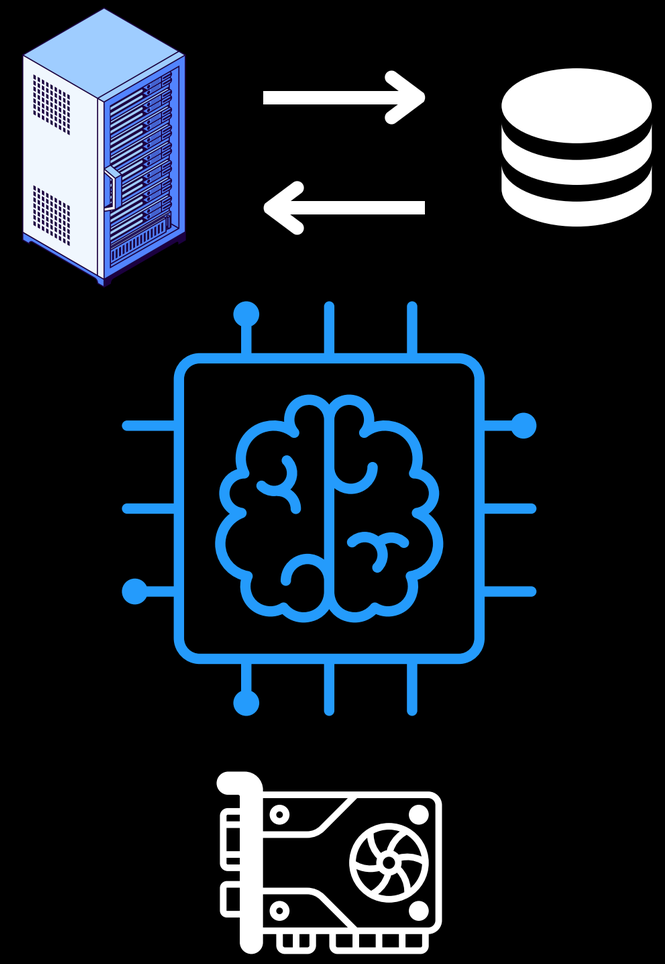
- Modelos de IA conversam com os dados e com o server
- Eles fazem parte dos artefatos de código e dos dados que compuseram seu treinamento
- Modelos podem **errar** sem lançar nenhum tipo de problema ao sistema (gerar uma resposta errada)
- Testar e avaliar esses modelos, em alguns casos, deixa de ser uma tarefa trivial



Modelo

LLMOps: Dificuldades

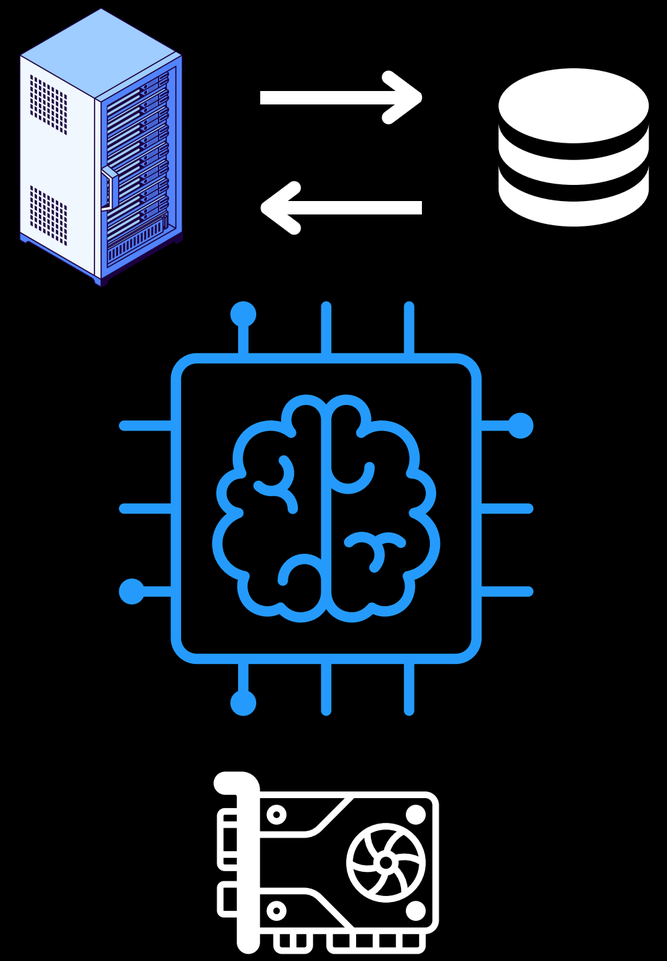
- Necessidade de orquestração de um grande volume de dados
- Custo computacional elevado
  - Treinamento
  - Validação
  - Produção
- Necessidade de aproveitar de forma inteligente os recursos computacionais disponíveis



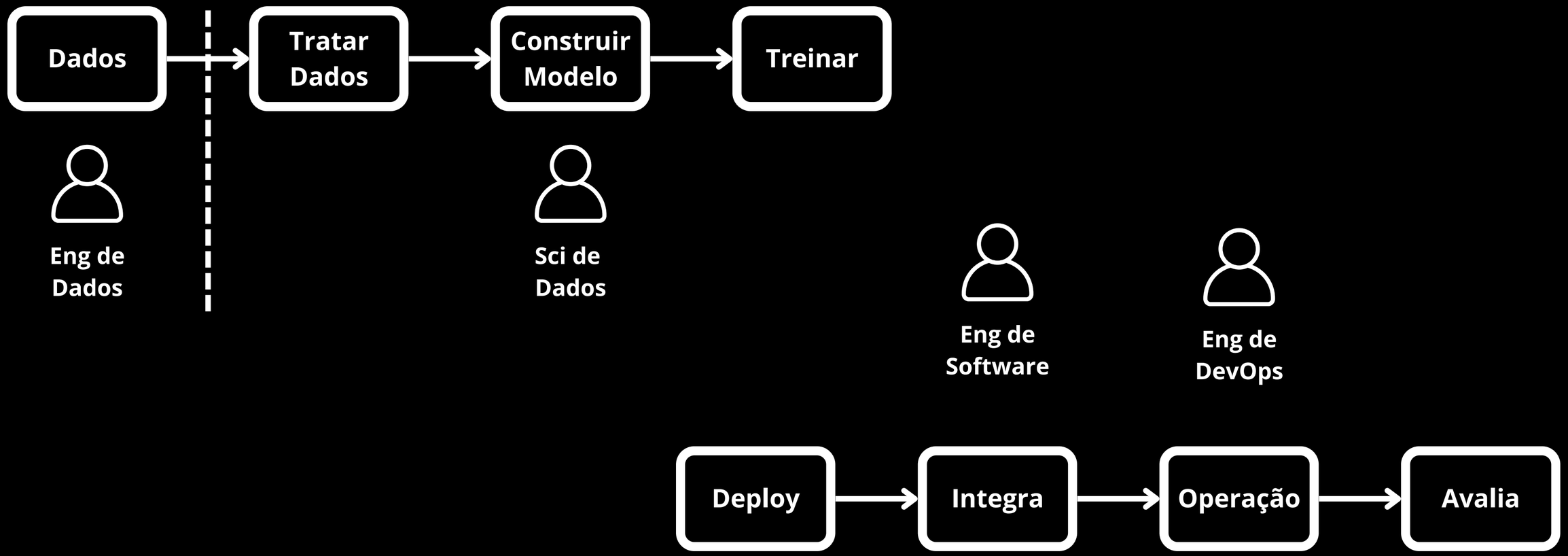
LLMOps: Dificuldades

- Atender a diferentes requisitos:
  - Time de vendas
  - Time de ML
  - Time de finanças
  - Clientes
  - Líderes

- Muitos desses requisitos acabam virando métricas de avaliação de performance

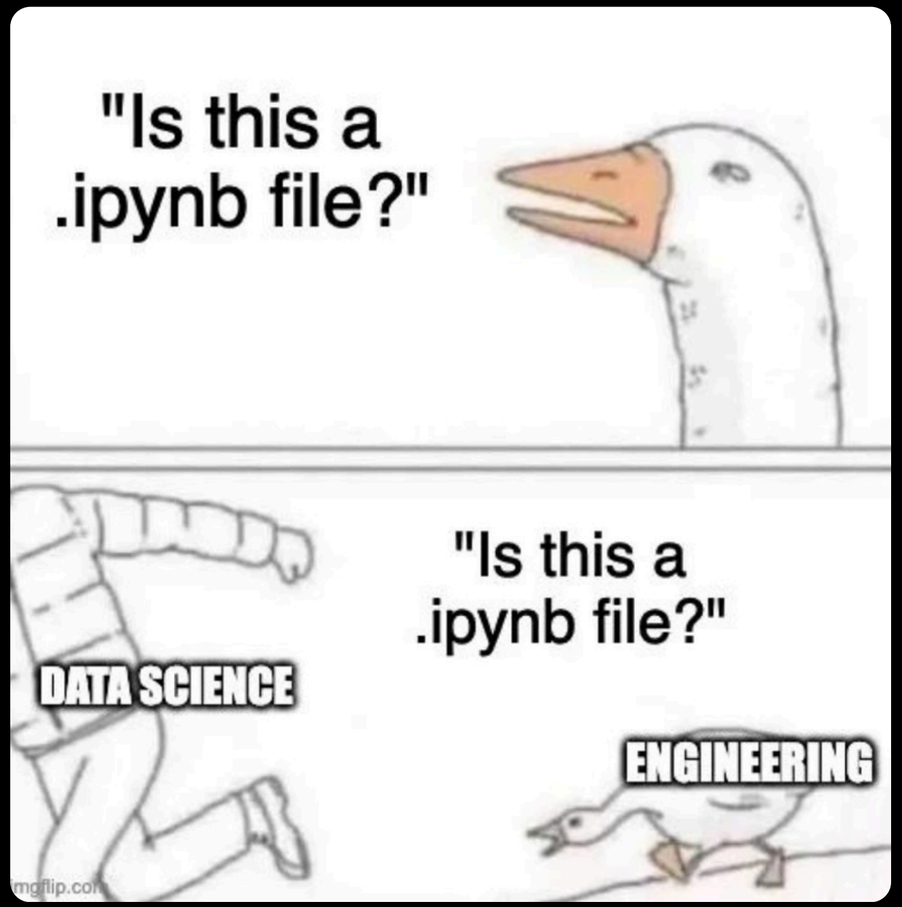


MLOps: Dificuldades



MLOps: Dificuldades

- Lidar com integrações entre times de desenvolvedores, cientistas de dados, engenheiros de dados, etc.
- **Isso quando essas funções estão bem definidas**



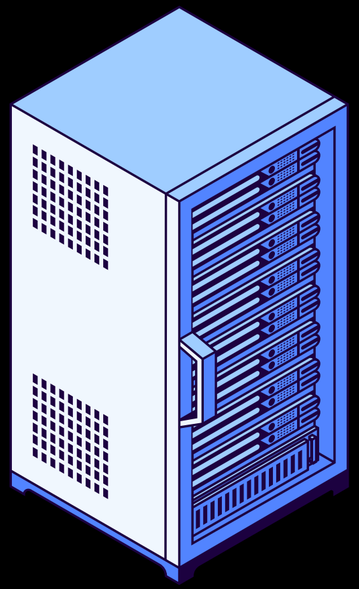
MLOps: Manutenção

- Assim como qualquer projeto de software, projetos de ML não ficam prontos

- Atualização e manutenção contínuas são necessárias para lidar com fatores do ambiente de deploy
  - Mudança na distribuição dos dados
  - Surgimento de eventos especiais que aletaram as requisições
  - Mudanças de comportamento dos usuários

Fundamentos de Docker

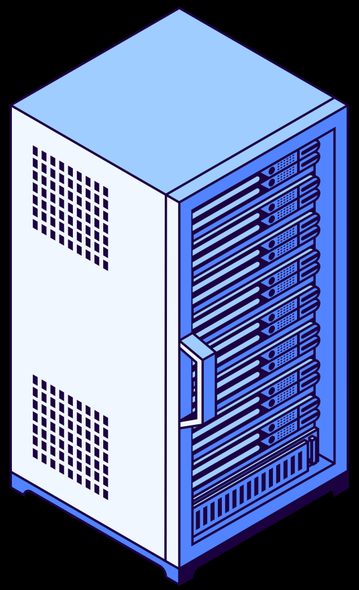
Motivação

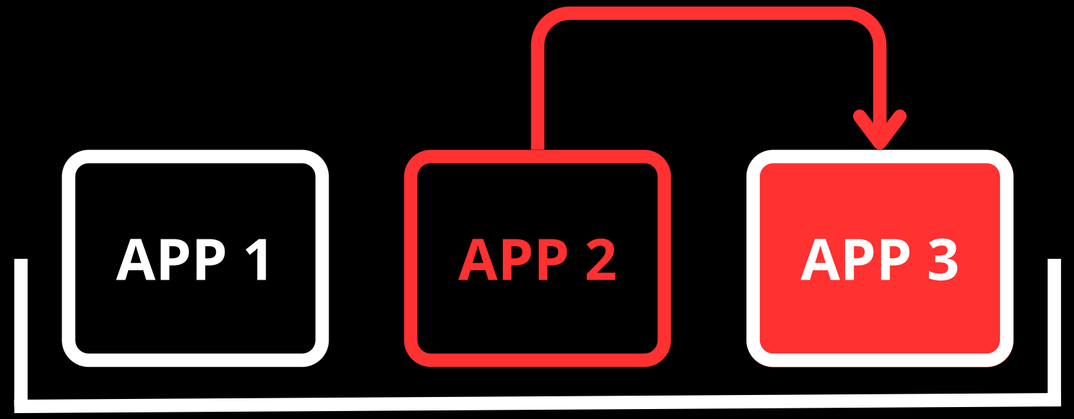


Servidores

Único ambiente
SO

Motivação

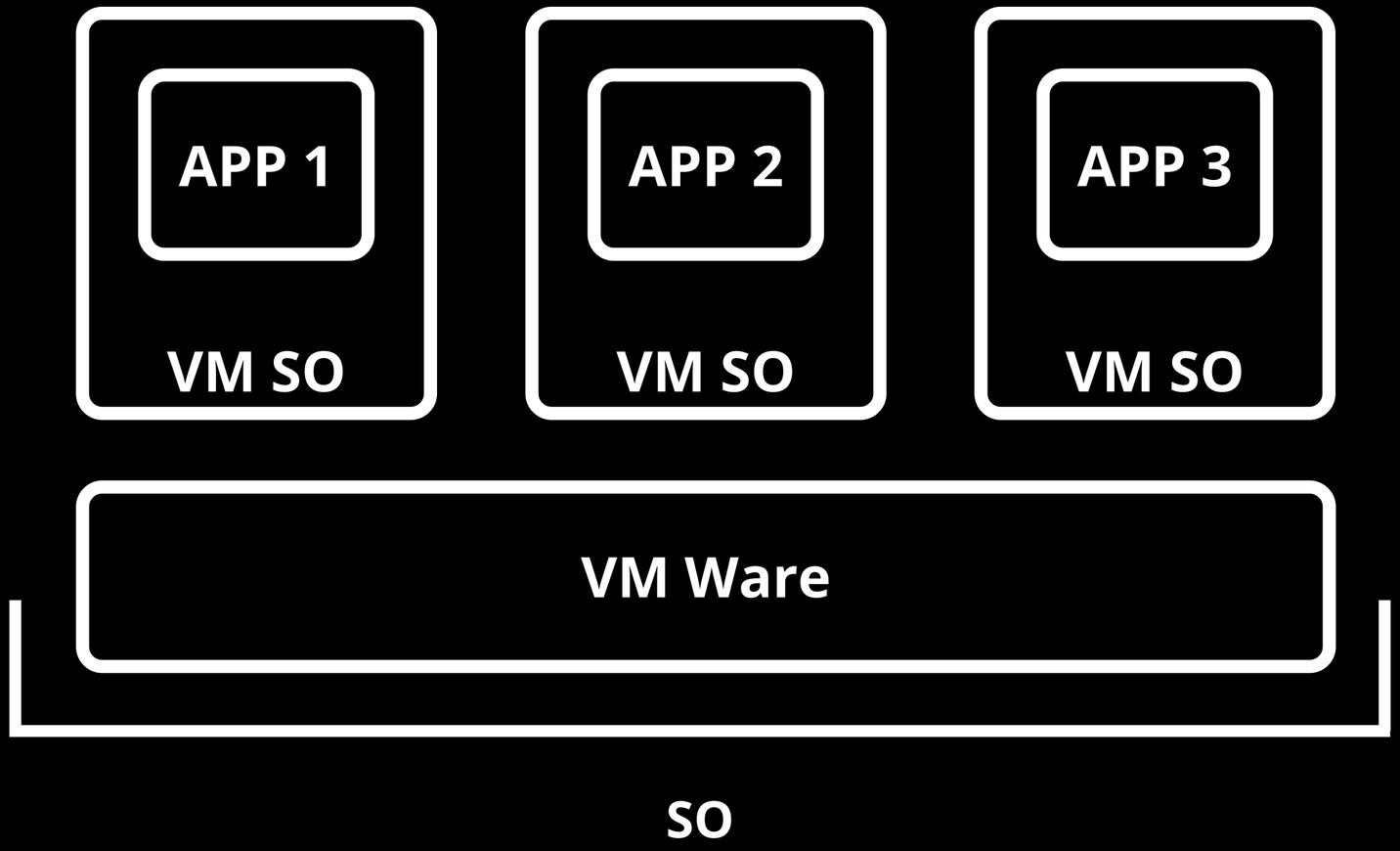




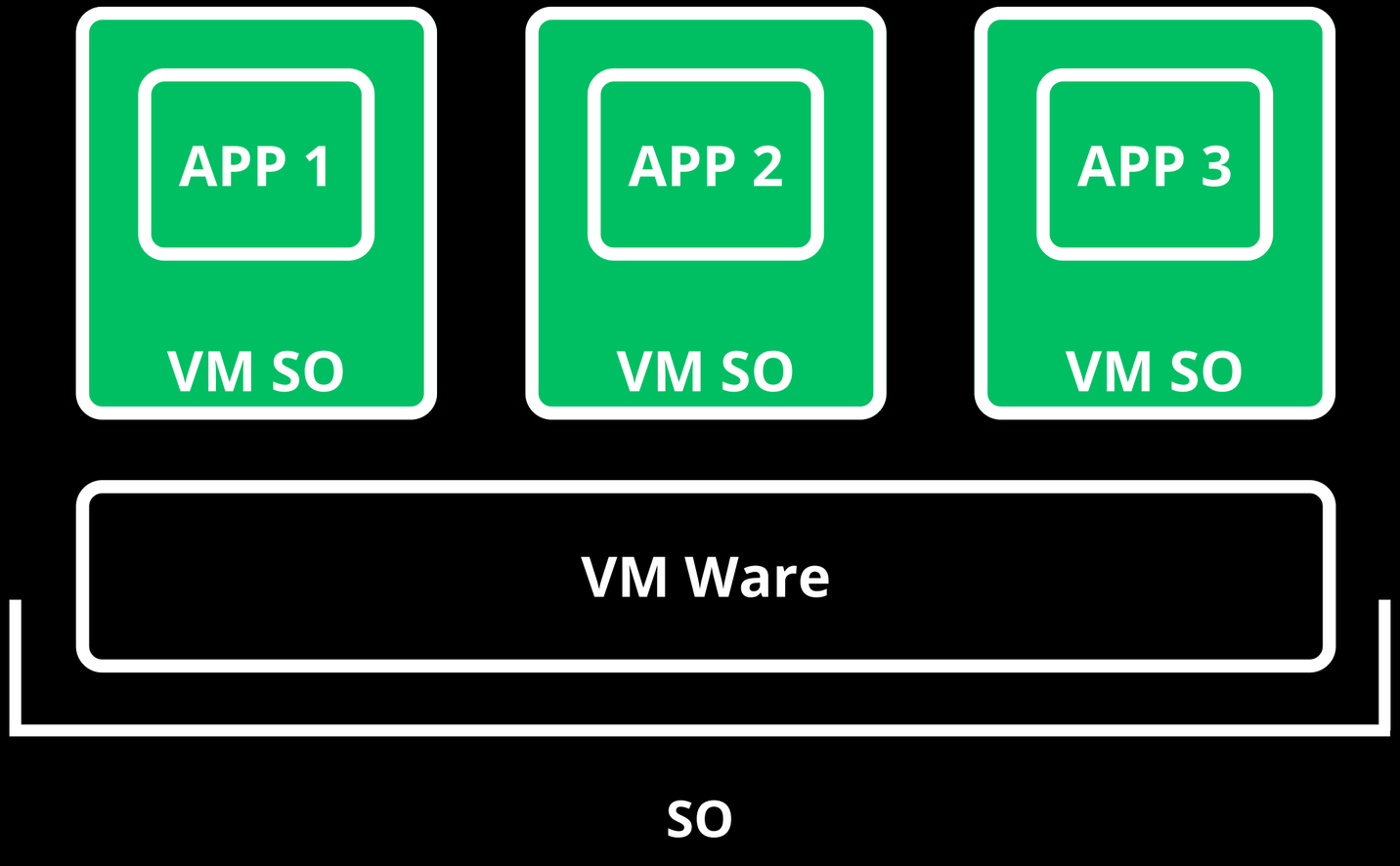
Servidores

Problemas com dependências/conflitos

Virtualização

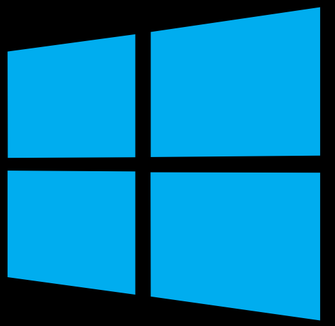


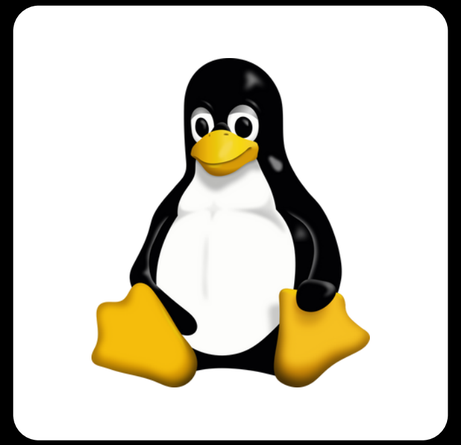
Virtualização



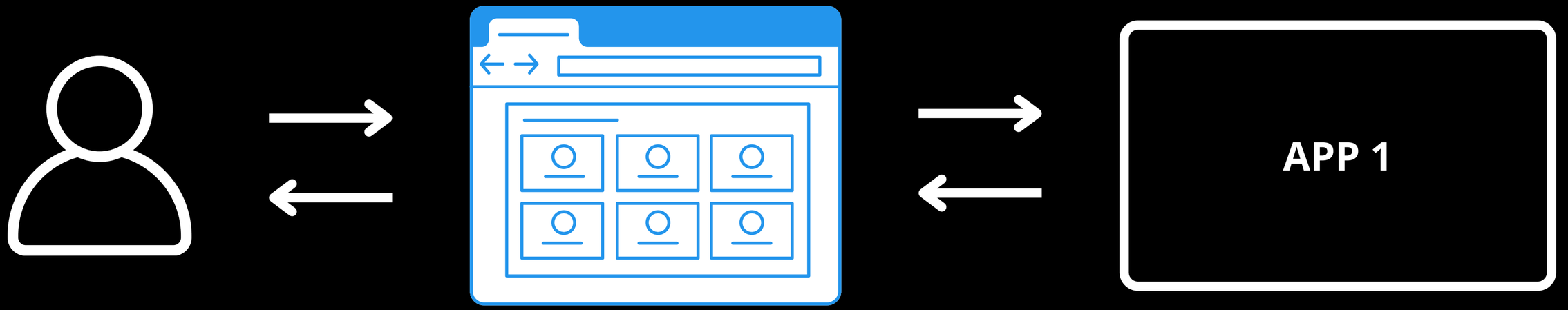
Recursos de um SO

- Sistema de Arquivos
- Interface Gráfica (GUI)
- Gerenciador de Processos
- Controlador de I/O
- Controle de Rede
- Segurança em acesso
- [...]



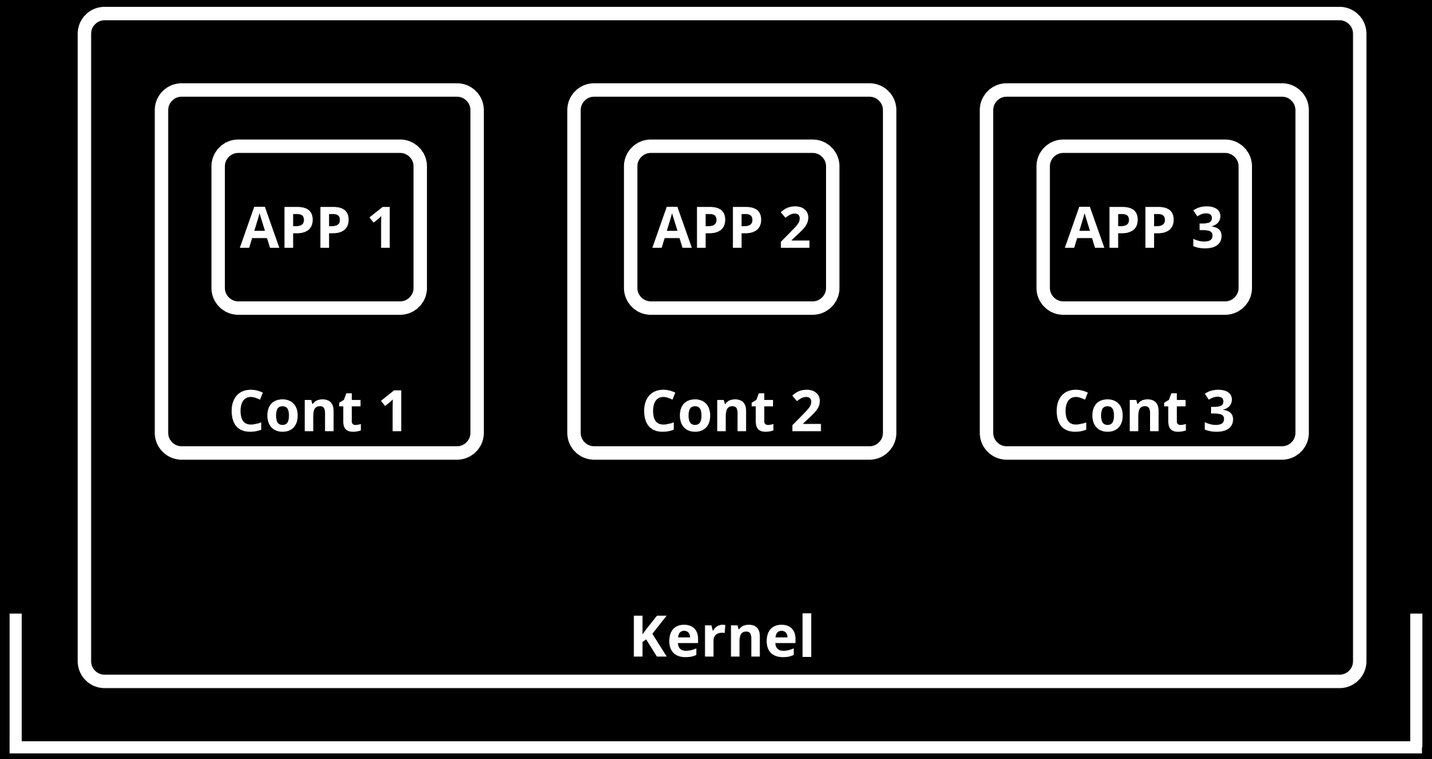


Recursos de um SO

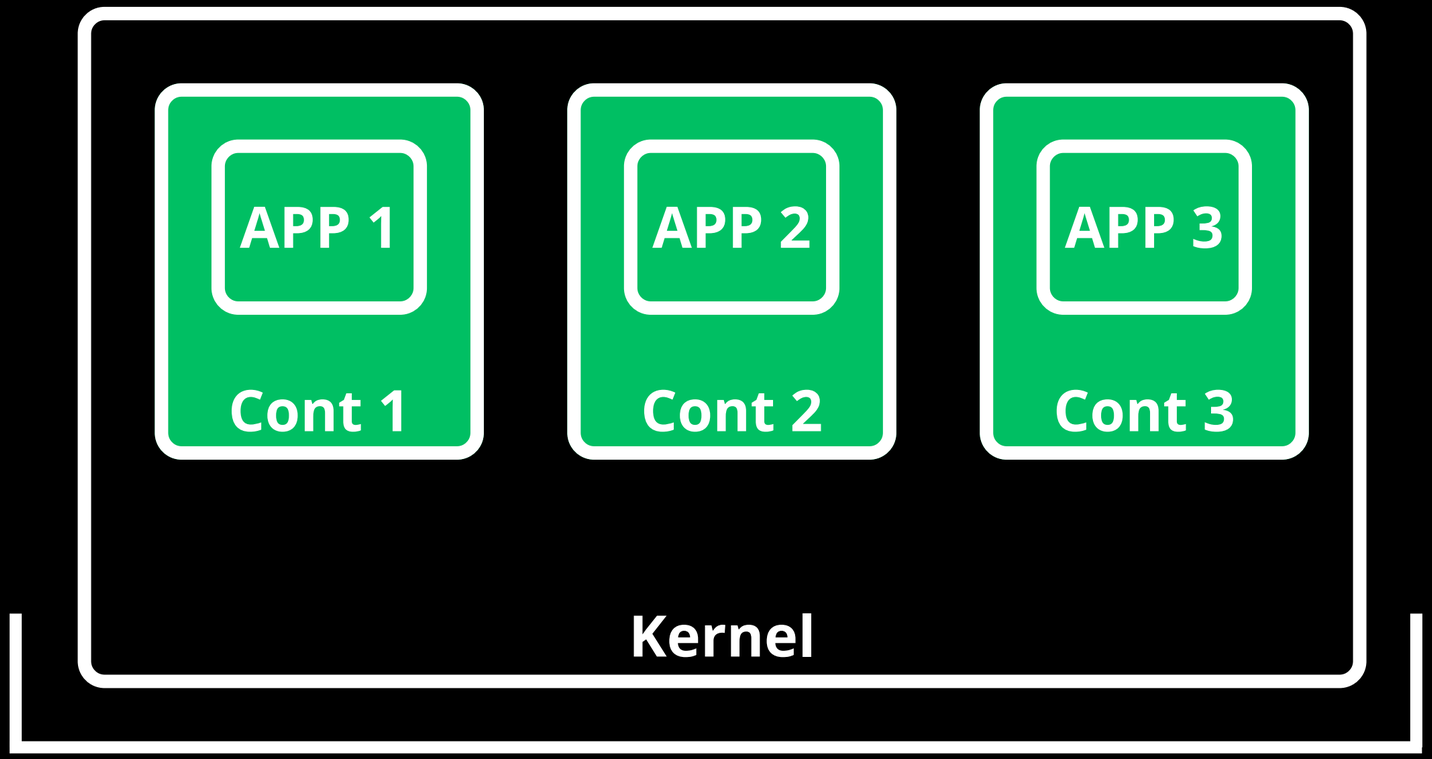


Sem necessidade de tantos recursos

Containers



Containers

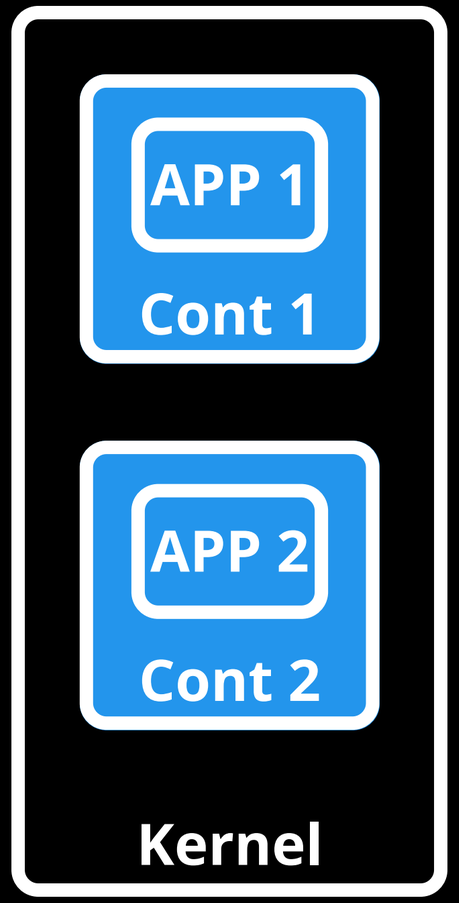


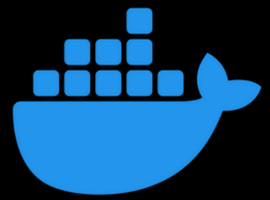
Docker

- Plataforma para implementação de containers
- Tecnologia popular
- Comunidade forte e criação de um Hub
- Ampla adoção em ambientes de cloud
- Open Source

GCP
Cloud Run

AWS
Fargate



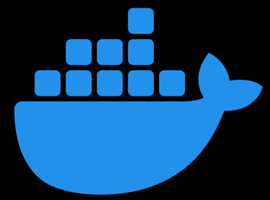


Docker: Imagens

- Containers são inicializdos a partir de uma imagem
- Uma imagem especifica quais recursos e comandos o container precisa para executar a aplicação
- **Imagens não são containers**

docker images

```
C:\Users\lucas>docker images
REPOSITORY                                TAG                 IMAGE ID       CREATED          SIZE
streamlit                                 latest              f0891c033434   20 hours ago     1.59GB
medallion-architecture-medallion-services latest              a3761af05d9d   39 hours ago     3.47GB
generic                                   latest              2e4b0581b44e   42 hours ago     2.93GB
mvp-qdrant                                latest              86237736a543   5 weeks ago      1.53GB
mvp-tron-front                            latest              0b43f088aade   5 weeks ago      2.03GB
memcached                                 latest              89b2dfa7e55e   2 months ago     131MB
searxng/searxng                           2025.3.16-84636ef49 d904830b5d61   2 months ago     279MB
ankane/pgvector                           latest              956744bd14e9   20 months ago    628MB
```

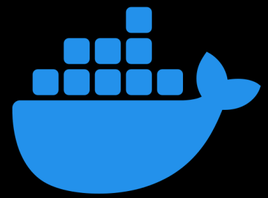


Docker: Dockerfile

- Arquivo que descreve como a imagem será montada
- Ela geralmente é montada com referência a uma pasta local

```
FROM python:3.10.14-slim

COPY requirements.txt requirements.txt

RUN pip install -r requirements.txt

WORKDIR /app

COPY streamlit/* .

EXPOSE 8501

CMD ["python", "-m", "streamlit", "run", "app_stream.py"]
```

Dockerfile



Docker: Dockerfile

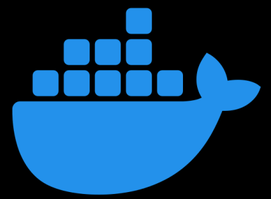
docker build -f <DOCKERFILE> -t <CONTAINER_NAME> .

```
$ docker build -f docker/Dockerfile.app -t streamlit-app .
[+] Building 1.2s (10/10) FINISHED                                                                   docker:desktop-linux
 => [internal] load build definition from Dockerfile.app                                                             0.0s
 => => transferring dockerfile: 262B                                                                                 0.0s
 => [internal] load metadata for docker.io/library/python:3.10.14-slim                                               0.9s
 => [internal] load .dockerignore                                                                                    0.0s
 => => transferring context: 2B                                                                                      0.0s
 => [1/5] FROM docker.io/library/python:3.10.14-slim@sha256:2407c61b1a18067393fecd8a22cf6fceede893b6aaca817bf9fbfe65e33614a3   0.0s
 => => resolve docker.io/library/python:3.10.14-slim@sha256:2407c61b1a18067393fecd8a22cf6fceede893b6aaca817bf9fbfe65e33614a3  0.0s
 => [internal] load build context                                                                                    0.0s
 => => transferring context: 188B                                                                                    0.0s
 => CACHED [2/5] COPY requirements.txt requirements.txt                                                              0.0s
 => CACHED [3/5] RUN pip install -r requirements.txt                                                                 0.0s
 => CACHED [4/5] WORKDIR /app                                                                                        0.0s
 => CACHED [5/5] COPY streamlit/* .                                                                                  0.0s
 => exporting to image                                                                                               0.1s
 => => exporting layers                                                                                              0.0s
 => => exporting manifest sha256:a91506f6ab487919e1c85bd9c941a940b4a44518e593c304cc3972d0a22095a4                    0.0s
 => => exporting config sha256:1b3fbe2eb3936b803d81afce3ea17509ce20fd6ab75e92d3af804b5908594624                      0.0s
 => => exporting attestation manifest sha256:e06f5eac7ab7face391112fdd7ecd6f9c239987b19d1b7070f68fd7fd0e09671        0.0s
 => => exporting manifest list sha256:95c469775f893571f97f56f70e090c012fb7822ed4e87e51d604fa2c630d5163               0.0s
 => => naming to docker.io/library/streamlit-app:latest                                                              0.0s
 => => unpacking to docker.io/library/streamlit-app:latest                                                           0.0s
```

Docker: Execução

- Após a etapa de build, a imagem está pronta, mas o container precisa ser executado com base na imagem
- O “run” também pode ter parâmetros específicos que diferenciam containers, mesmo que criados de uma mesma imagem

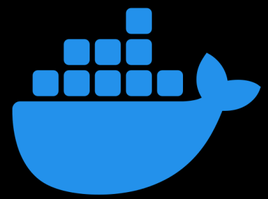
docker run -p <OUT_PORT>:<IN_PORT> -it <CONTAINER_NAME>



Docker: Execução

docker run -p <OUT_PORT>:<IN_PORT> -it <CONTAINER_NAME>

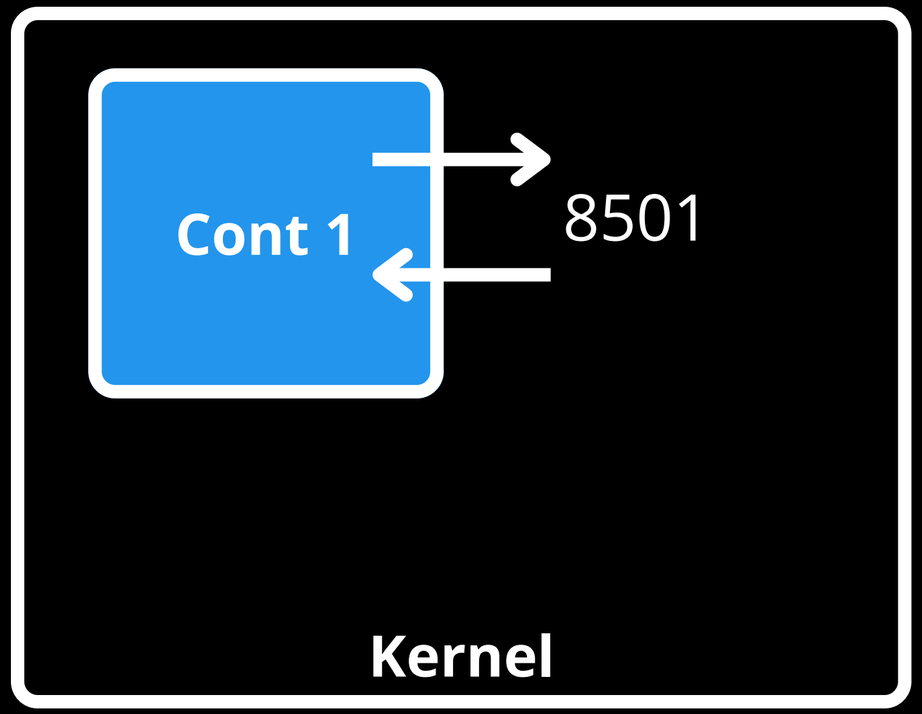
```
$ docker run -p 8501:8501 -it streamlit-app

Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.


  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://172.17.0.2:8501
  External URL: http://200.137.197.75:8501
```



Docker: “Fluxo”

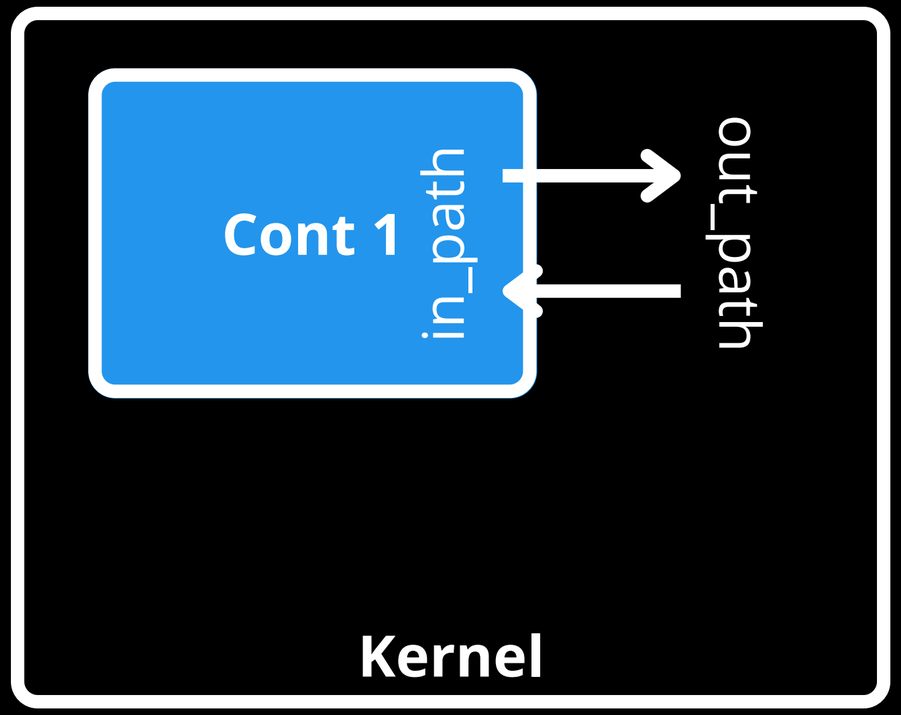
Container: Rede

docker run **-p 8501:8501** -it <CONTAINER_NAME>



Container: Volume

docker run **-v <out_path>:<in_path>** -it <CONTAINER_NAME>



Referências

- https://docs.docker.com/guides/
- https://github.com/kamranahmedse/developer-roadmap
- https://roadmap.sh/docker
- https://www2.decom.ufop.br/terralab/um-breve-historico-sobre-virtualizacao/?utm_source=chatgpt.com
- https://www.techtarget.com/searchitoperations/feature/Dive-into-the-decades-long-history-of-container-technology
- https://www.grupounibra.com/repositorio/REDES/2022/analise-de-desempenho-entre-maquinas-virtuais-e-containers-utilizando-o-docker3.pdf?utm_source=chatgpt.com
- https://www.targetso.com/artigos/containers-e-virtualizacao/