FIAP GRADUAÇÃO

FIAP

DATA SCIENCE

BIG DATA ARCHITECTURING & DATA INTEGRATION

Prof. Dr. Renê de Ávila Mendes

Objetivos da disciplina

DISCIPLINA: Big Data Architecturing & Data Integration

OBJETIVOS: Entenda as principais **arquiteturas** para ingestão, processamento e análise de grandes volumes de dados. Conheça as principais **ferramentas** open-source de Big Data como Hadoop, MapReduce, Spark, Sqoop, NiFi, Flume, Kafka, Zookeeper, HBase, Hive e as **integre** com as ferramentas de **extração, transformação e carga** de dados em modelos dimensionais. Entenda conceitos sobre **computação paralela e distribuída**, aplicação do Hadoop e bases Apache e arquiteturas *serverless* e desacopladas. Veja como **visualizar os dados** estruturados ou não estruturados com ferramentas de Self-Service Business Intelligence como PowerBI, utilizando as melhores práticas de **visualização de dados**.

Assuntos – 1º Semestre

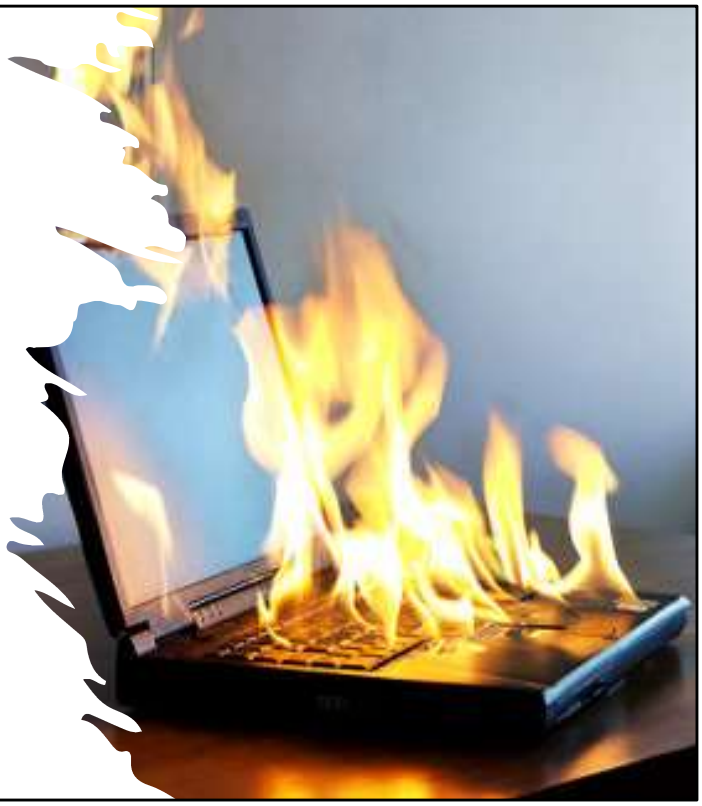
- Introdução a Big Data
- **Conceitos Computação Paralela e Distribuída, Lei de Moore, Sistema HDFS**
- Aplicação Hadoop. Usos e Administração de Ambientes Hadoop
- Intodução a MapReduce
- Introdução à Integração de Dados
- Integração entre SQL e Hadoop - SQOOP
- Bases Apache
- Bases Apache - PIG
- Introdução da Data Streaming - FLUME
- Introdução a análise de dados com SPARK

“Lei” de Moore

- Gordon Moore
- 1965/1975
- Fundador da Intel
- Número de transistors em um circuito integrado dobraria a cada 2 anos
- É uma observação de uma tendência histórica
- Guiou os objetivos da indústria de microprocessadores
- Resultados
  - Diminuição do custo de processadores, memórias, computadores
  - Melhoria das memórias, sensores e câmeras

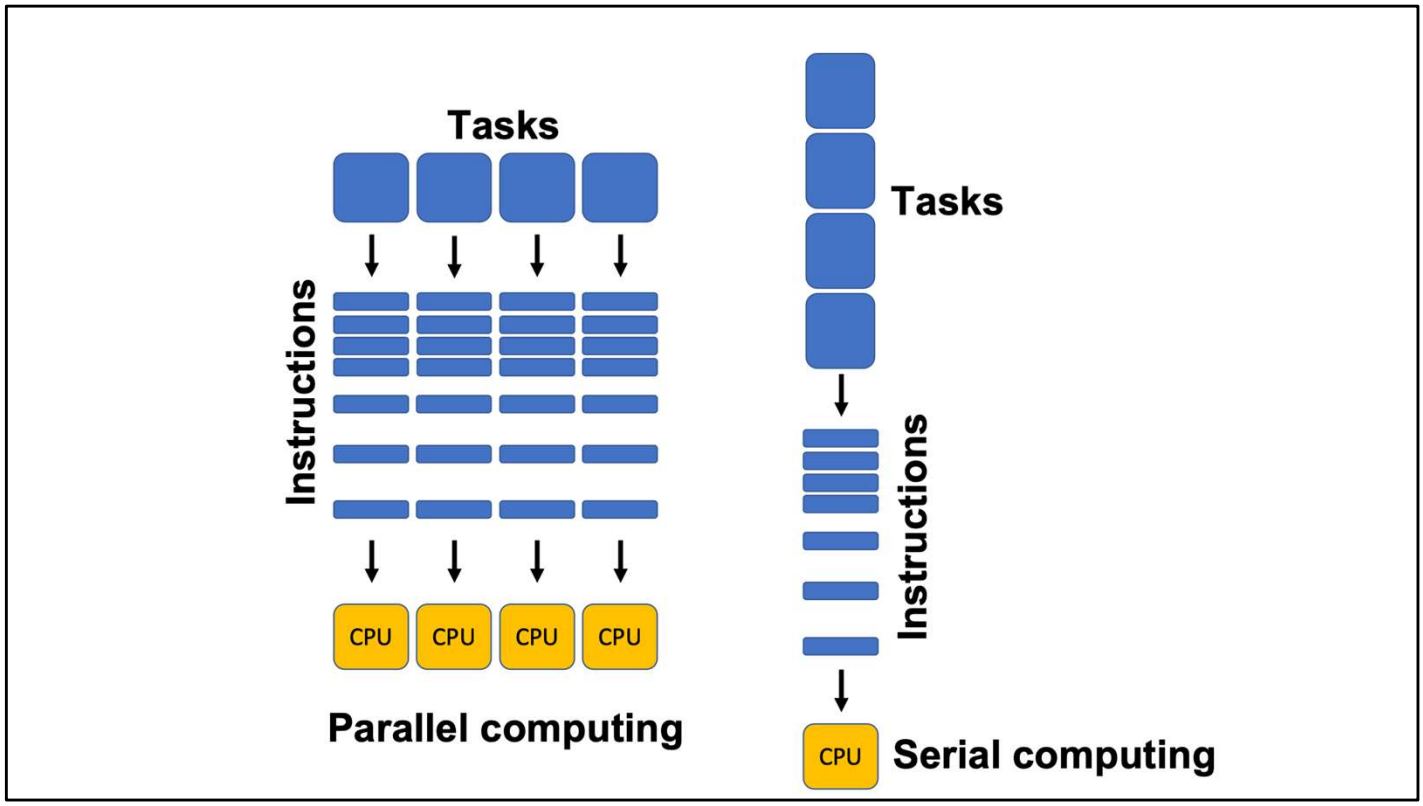
Problema

Limitação do *single core:* o **consumo de energia** e o **aquecimento** limitam o aumento de velocidade que se obtém de um transistor



Computação paralela

- Saída para o processamento serial
- A tarefa computacional é quebrada em **sub-tarefas** que podem ser processadas independentemente e os resultados são combinados posteriormente


Tasks
Instructions
CPU
CPU
CPU
CPU
Parallel computing
Tasks
Instructions
CPU
Serial computing

Computação paralela

- Um único servidor
  - Multi-core
  - Multi-processor
  - MPP - massively parallel processing
    - GPU (Graphics processing unit)
- Múltiplos servidores
  - Cluster
  - Grid
- Escalamento vertical: mais hardware para mais processamento

Computação distribuída

- Múltiplos computadores em rede
- Ação coordenada
- Comunicação entre os componentes
- Escalamento horizontal
  - Adicione computadores para aumentar o processamento

Computação paralela x distribuída

	PARALELA	DISTRIBUÍDA
Número de computadores	1 computador com múltiplos processadores ou cores	Múltiplos computadores autônomos, separados geograficamente
Escalabilidade	Vertical e limitada à quantidade de acessos simultâneos à memória	Horizontal, pela adição de novos computadores
Memória	Compartilhada entre todos os processadores	Cada computador tem a sua
Comunicação	Bus	Mensagens por rede
Uso	Quando a quantidade de computadores disponíveis é limitada	Compartilhamento de recursos e escalabilidade crescente


Big Data

Afinal, o que é Big Data ?

- Dados com características como estas:
  - grandes (volume)
  - rápidos (velocidade)
  - difíceis de serem tratados pelos sistemas atuais (variedade)
- Dimensões
  - **Volume:** estima-se que o volume de dados disponível no universo digital seja de 2,7 zetabytes, ou 2,7 x 10 21 bytes, ou ainda 2,7 trilhões de gigabytes;
  - **Variedade:** dados estruturados, semiestruturados e desestruturados;
  - **Velocidade:** fluxos de dados são gerados continuamente em grandes volumes;
  - **Variabilidade:** os dados podem assumir variedades de formas e interpretações diferentes;
  - **Valor:** possibilidade de extrair respostas aplicáveis ao negócio

MADDEN, S. From databases to big data. IEEE Internet Computing, IEEE, n. 3, p. 4–6, 2012.

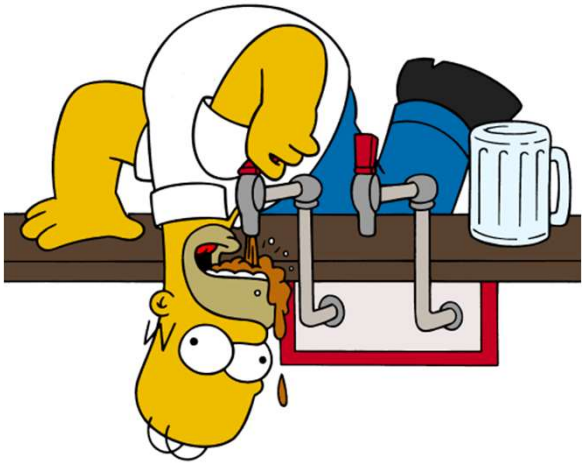
Dados com características como estas, grandes (volume), rápidos (velocidade) e difíceis de serem tratados pelos sistemas atuais (variedade) definem o termo Big Data (MADDEN, 2012)

Tipos de processamento

- Processamento em lote

Tipos de processamento

- Processamento em fluxo



Tipos de processamento

- Processamento em lote
  - Computa o **volume completo de dados** gerando visões a partir dele
  - Os volumes de dados são **grandes**
  - As **restrições de tempo** de resposta para a análise são menos exigentes
- Processamento em fluxo
  - Computa apenas o **volume de dados mais recente**
  - Requerido por aplicações de análise em **tempo real, ou quase real**
  - A variável tempo de resposta, ou latência, torna-se um **fator crítico** para o sucesso

Computação paralela ou distribuída ?

Computação paralela ou distribuída ?

- As duas combinadas
- Milhares de servidores distribuídos
- Múltiplos processadores em paralelo
- Onde está o gargalo então ?

Sistemas distribuídos – o gargalo dos dados

- Tradicionalmente os dados são armazenados em um local central
- Os dados são copiados para os processadores em tempo de execução
- Este modelo funciona bem para poucos dados
- No entanto, os sistemas modernos têm muito mais dados:
  - Terabytes por dia
  - Petabytes no total

http://www.cloudera.com/content/cloudera/en/resources/library/training/cloudera-essentials-for-apache-hadoop-the-motivation-for-hadoop.html


Hadoop

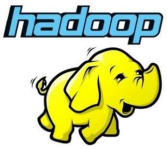
FIAP

Hadoop

- Projeto código aberto mantido pela Apache Foundation.
- Fornece uma implementação de código aberto do modelo de programação *MapReduce* de forma confiável e escalável.
- Projetado para ampliar o processamento de um único servidor em milhares de máquinas, onde cada uma das máquinas oferece poder de processamento e armazenamento local.
- Esta ferramenta é utilizada para processamento em *batch* de grandes volumes de dados (*Big Data*).

Haddop – Uma boa solução FIAP

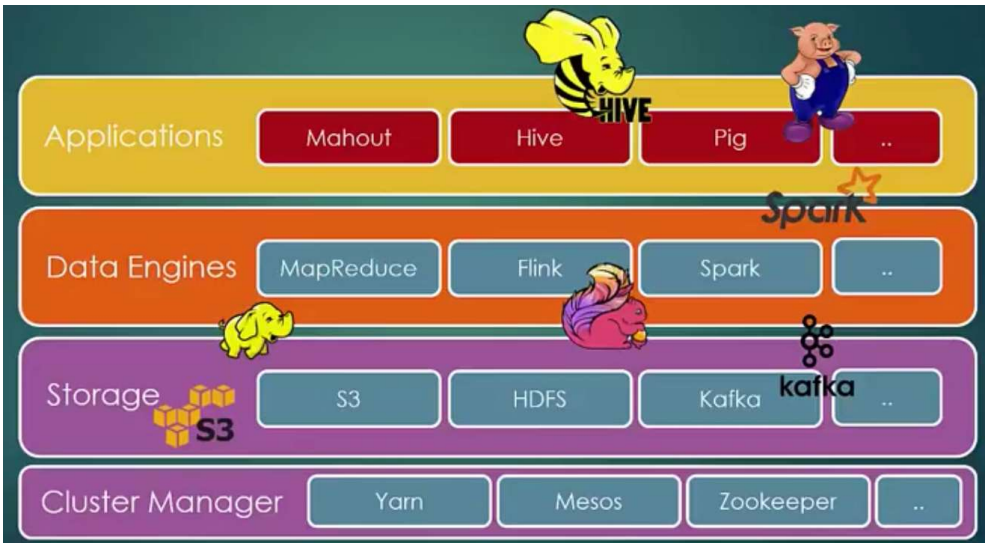
- Suporta grande volume de dados
- Armazenamento eficiente
- Boa solução de recuperação de dados
- Escalabilidade Horizontal
- Bom custo/benefício
- Simples para programadores e não-programa



Fonte: IBM

FIAP

Panorama do Big Data Apache



FIAP

Hadoop

- “Hadoop é um *storage* confiável **e** um sistema analítico”
- Composto por duas partes essenciais:
  - o **Hadoop Distributed Filesystem** (HDFS), sistema de arquivos distribuído e confiável, responsável pelo armazenamento dos dados
  - **Hadoop MapReduce**, responsável pela análise e processamento dos dados.
- O nome do projeto veio do elefante de pelúcia que pertencia ao filho do criador, Doug Cutting.

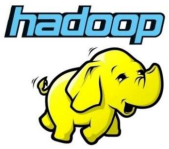
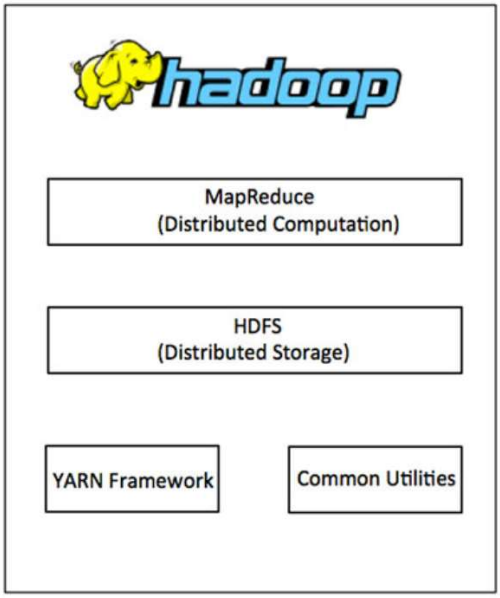


Diagrama Básico dos Componentes



Fonte: http://www.tutorialspoint.com/hadoop/hadoop_introduction.htm

Componentes do Hadoop

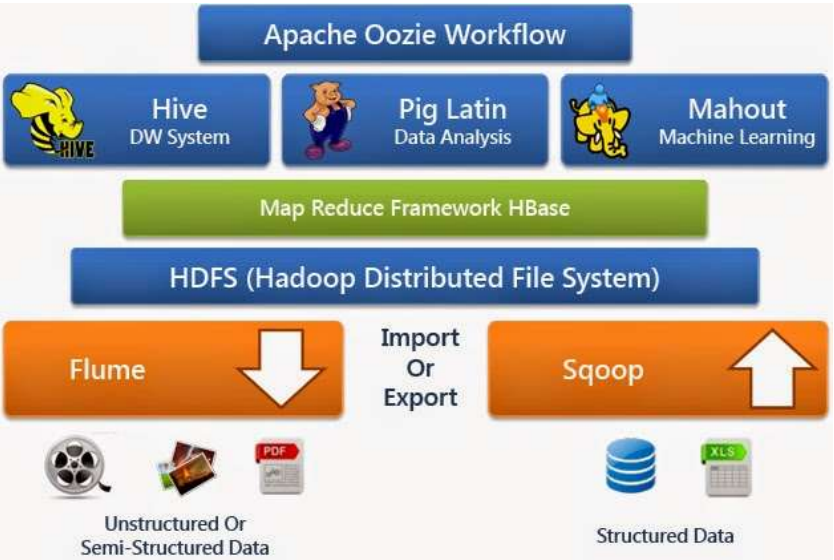
- O *framework* Hadoop é composto pelos seguintes módulos:
  - **Hadoop Common**: composto por bibliotecas JAVA e utilitários necessários para outros módulos. As bibliotecas oferecem abstração no nível do Sistema Operacional e sistema de arquivos e também possuem todos os arquivos e scripts necessários para inicializar o Hadoop;

Componentes do Hadoop

- **Hadoop YARN**: *framework* para agendamento de tarefas (*jobs*) e gerenciamento de recursos do *cluster*;
- **Hadoop *Distributed File System*** *(HDFS)*: sistema de arquivos distribuído que fornece acesso aos dados com elevadas taxas de transferência;
- **Hadoop MapReduce**: sistema baseado no Hadoop YARN para processamento paralelo de grandes volume de dados.

Ecosistema Hadoop

FIAP

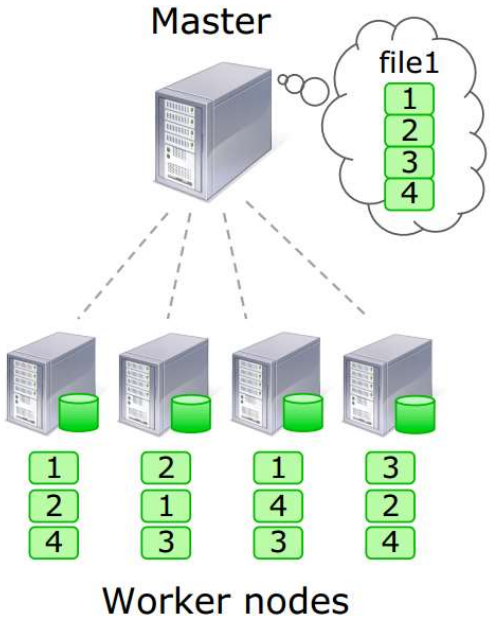


Fonte: EMC²

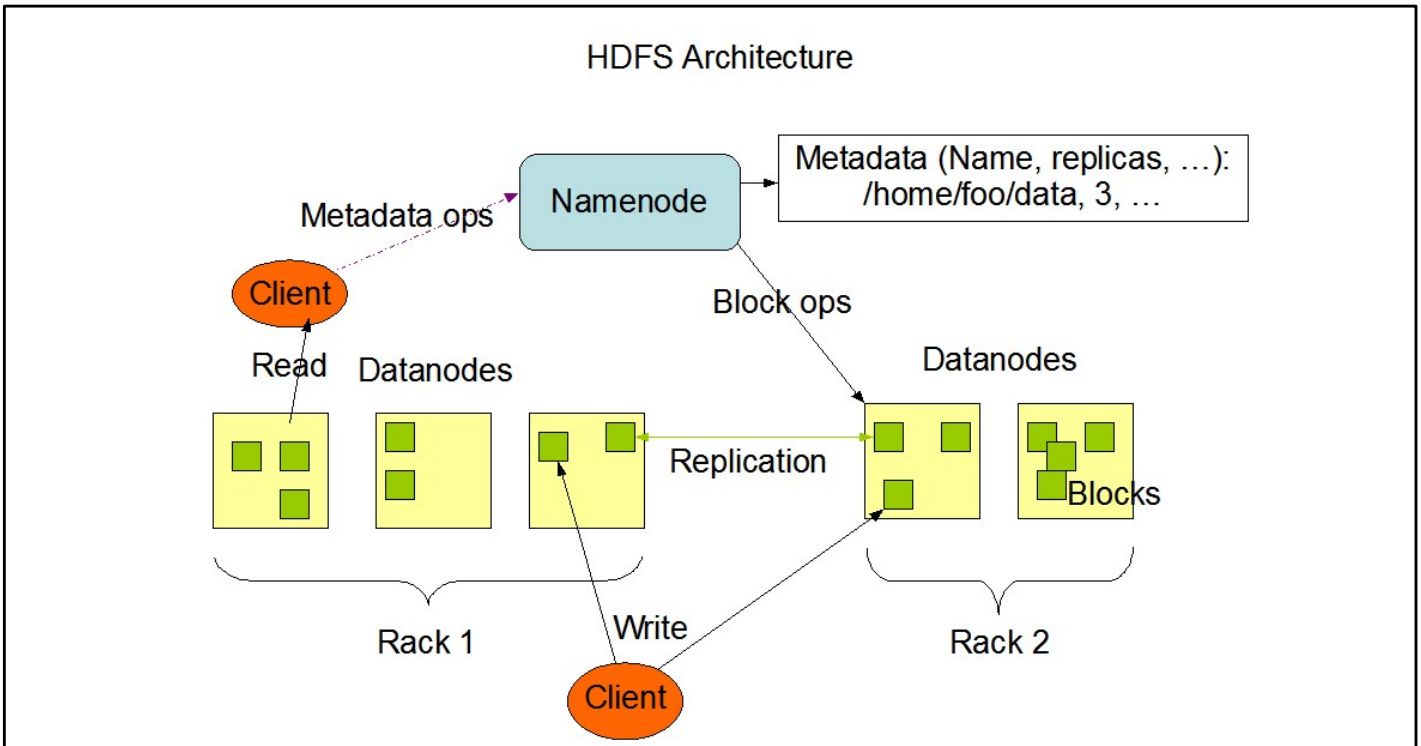
HADOOP DISTRIBUTED FILE SYSTEM

HDFS – Hadoop Distributed File System

- Arquivos são quebrados em blocos
- Blocos são replicados entre os nós
- Nó master armazena os metadados (nomes dos arquivos, localizações etc.)
- Otimizado para arquivos grandes e leituras sequenciais



Fonte: "Tackling the Challenges of Big Data - Big Data Storage: Distributed Computing Platforms" - Matei Zaharia , 2014, Massachusetts Institute of Technology

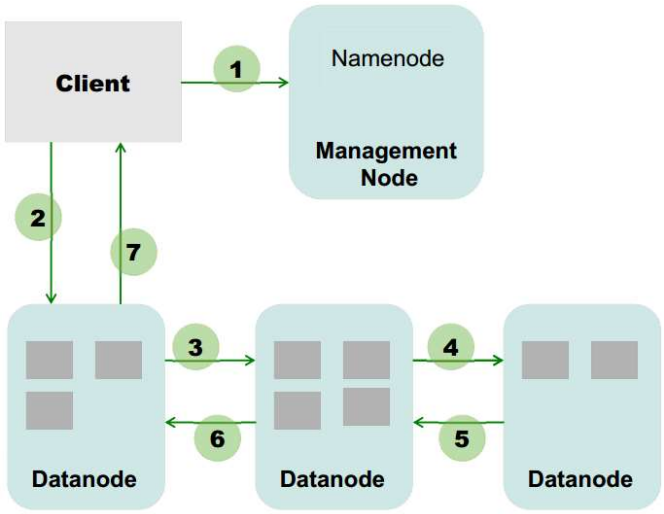


https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsDesign.html

Namenode e Datanodes

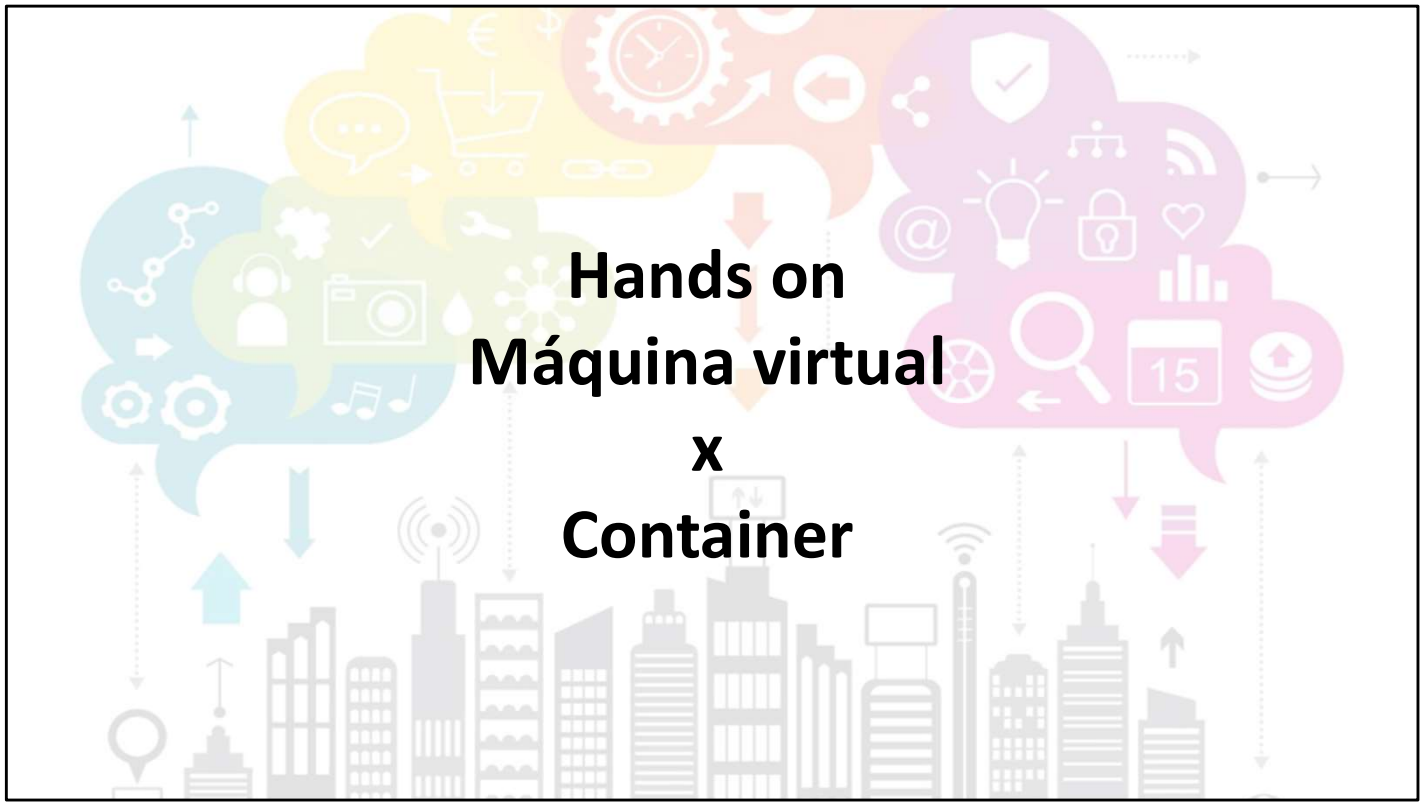
O HDFS possui uma arquitetura mestre/escravo. Um cluster HDFS consiste em um único NameNode, um servidor mestre que gerencia o namespace do sistema de arquivos e regula o acesso aos arquivos pelos clientes. Além disso, há vários DataNodes, geralmente um por nó no cluster, que gerenciam o armazenamento nos nós em que são executados. O HDFS expõe um namespace do sistema de arquivos e permite que os dados do usuário sejam armazenados em arquivos. Internamente, um arquivo é dividido em um ou mais blocos e esses blocos são armazenados em um conjunto de DataNodes. O NameNode executa operações no namespace do sistema de arquivos, tais como abrir, fechar e renomear arquivos e diretórios. Ele também determina o mapeamento de blocos para DataNodes. Os DataNodes são responsáveis por atender solicitações de leitura e gravação dos clientes do sistema de arquivos. Os DataNodes também executam a criação, exclusão e replicação de blocos mediante instruções do NameNode.

HDFS – Escrita de um arquivo



1. Cria o novo arquivo no namespace do Namenode; calcula a topologia do bloco;
2. Faz streaming dos dados para o primeiro nó;
3. Faz streaming dos dados para o segundo nó;
4. Faz streaming dos dados para o terceiro nó;
5. ACK sucesso/falha
6. ACK sucesso/falha
7. ACK sucesso/falha

Fonte: “Hadoop Distributed File System (HDFS) Overview” (http://www.coreservlets.com/hadoop-tutorial/)

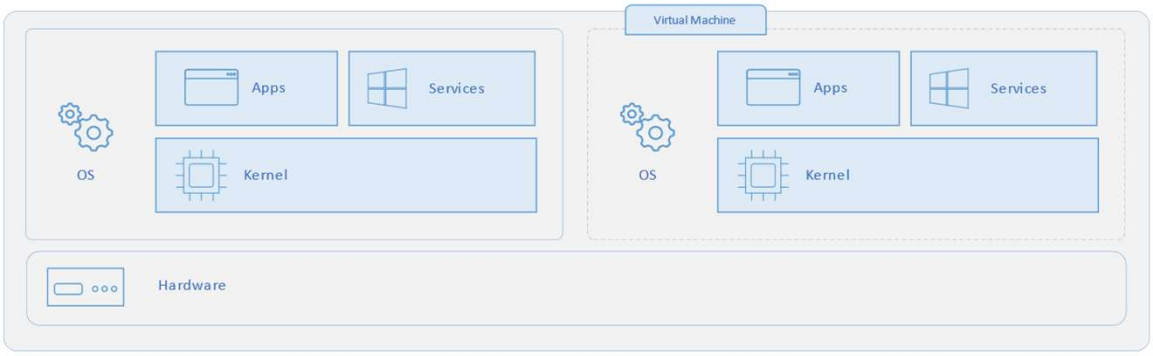

Hands on
Máquina virtual
x
Container

Problema

- Sua máquina local **não possui os mesmos programas** que você precisa para estudar e trabalhar
- Sua máquina local nem **mesmo possui o mesmo Sistema Operacional** que você encontrará em ambientes corporativos

Máquina Virtual

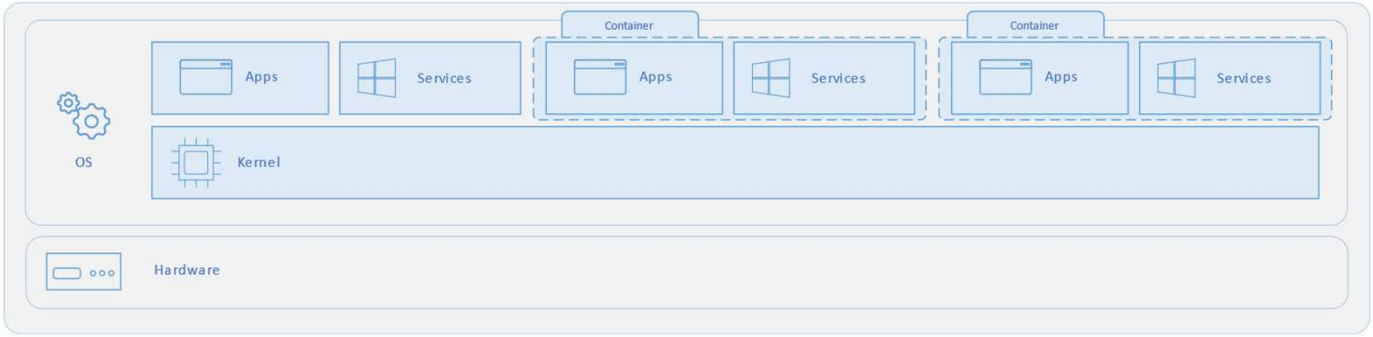
- Executa o sistema operacional completo
- Usa o hardware da sua máquina local
- Requer um virtualizador



Fonte: https://learn.microsoft.com/pt-br/virtualization/windowscontainers/about/containers-vs-vm

Container

- Usa o sistema operacional da máquina local
- Usa o hardware da sua máquina local
- Requer uma plataforma de containerização



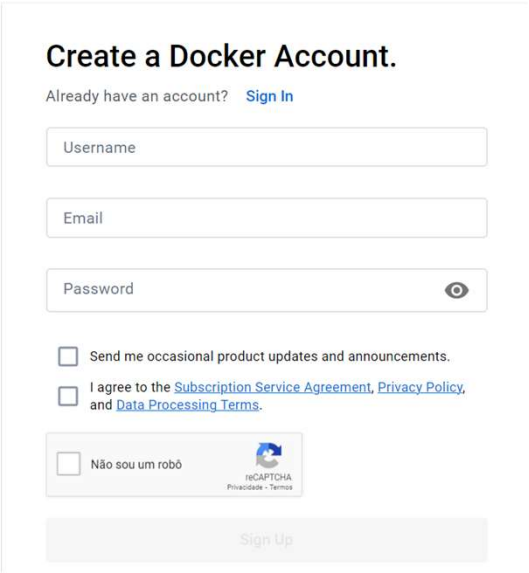
Fonte: https://learn.microsoft.com/pt-br/virtualization/windowscontainers/about/containers-vs-vm

Play With Docker

- Executa containers em servers remotos
- Não requer infraestrutura
- Requer cadastro prévio (https://hub.docker.com/signup)
- (https://www.docker.com/play-with-docker)

Play With Docker - Cadastro

- Acesse https://hub.docker.com/signup
- Crie um username e uma senha
  - Prefira um username igual ao seu username de aluno da FIAP
- Informe seu email da FIAP

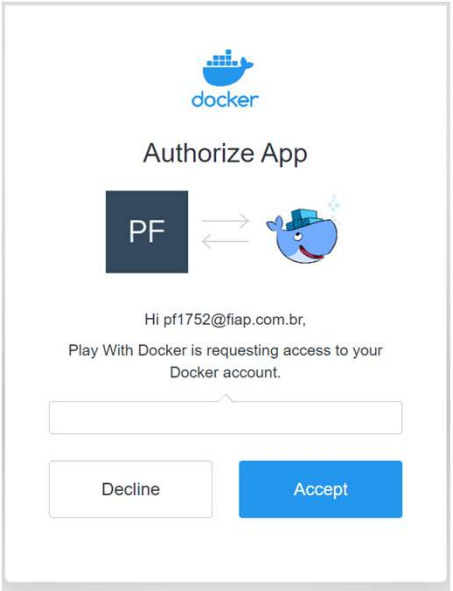


Play With Docker - Cadastro

- Faça seu primeiro login
- Escolha o tipo de plano
  - Escolha o gratuito
- Acesse seu email da FIAP e clique no link para verificar sua conta

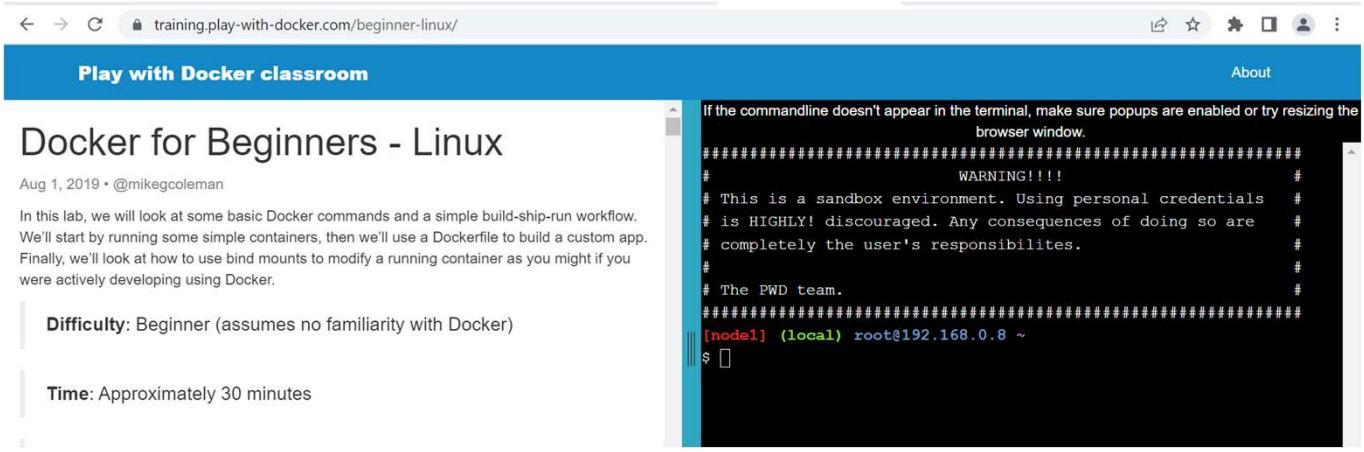
Play With Docker - Cadastro

- Autorize o usuário a acessar o aplicativo



Play With Docker - Linux

- Acesse o container para treinamento de Linux
- https://training.play-with-docker.com/beginner-linux/



Linux - Comandos

- Acesse o container para treinamento de Linux
- https://training.play-with-docker.com/beginner-linux/

FIAP

Linux - Comandos

- ls – lista o diretório
  - ls
  - ls > teste.txt
- cat – lista o conteúdo de um arquivo
  - cat teste.txt
- mkdir – cria diretório
  - mkdir teste

FIAP

Linux - Comandos

- cp – copia arquivo
  - cp teste.txt teste1.txt
- mv – move um arquivo
  - mv teste1.txt /etc/teste2.txt
- rm – apaga arquivo
  - rm /etc/teste2.txt