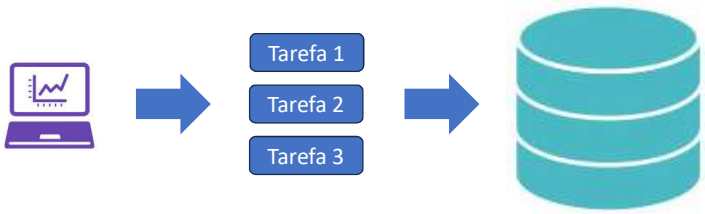
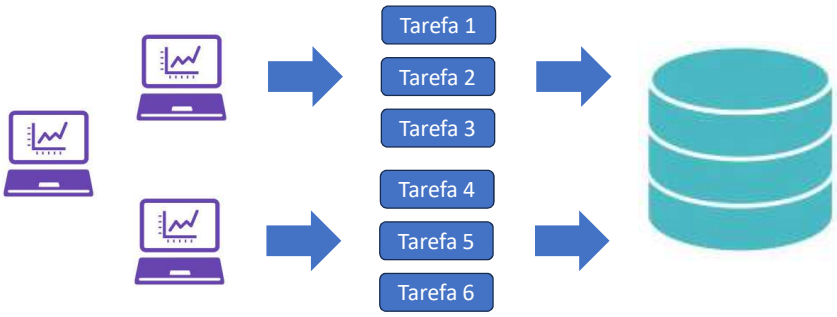
Tecnologias e Ferramentas de Big Data

Prof. Edson Pedro Ferlin

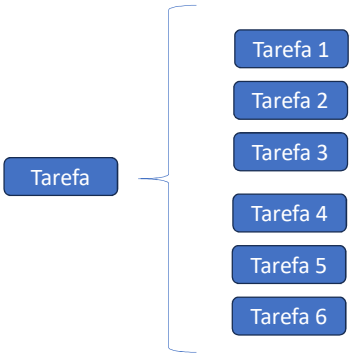
Processamento Tradicional



Processamento Distribuído



Processamento Paralelo



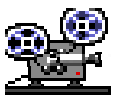
Esse processamento é dividido em vários nós ou clusters, para maximizar o poder computacional. Para simplificar, cluster é o conjunto de hardwares que trabalham sincronizadamente para funcionar como se fosse um único computador. Assim, diversas máquinas atuam de forma organizada como se fossem uma só.

O que é Processamento Paralelo?

Na publicação (**O que é Processamento Paralelo?**)
(link: http://professorferlin.blogspot.com/2011/08/o-que-e-processamento-paralelo.html)
temos um resumo sobre processamento paralelo.

Processamento Paralelo

Assista o vídeo sobre Processamento Paralelo
(link: https://youtu.be/lQZz0IJRmsg).



Hadoop

Hadoop

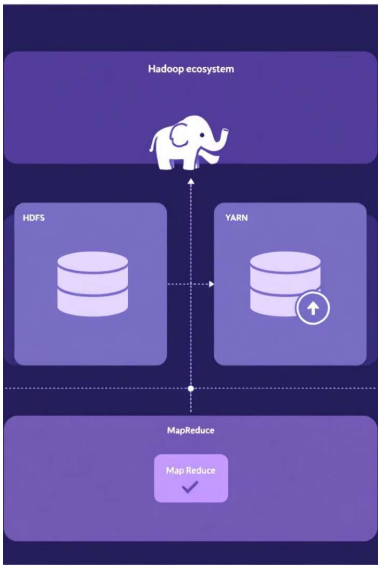
O projeto Apache™ Hadoop® desenvolveu software de código aberto para computação distribuída confiável, escalável.

A biblioteca de software Apache Hadoop é uma estrutura que possibilita o processamento distribuído de grandes conjuntos de dados em clusters de computadores utilizando modelos de programação simples.

Dividir para Conquistar

O Hadoop deixava que problemas grandes fossem divididos em elementos pequenos, para que a análise fosse feita rapidamente e com baixo custo.

Dividindo o problema de Big Data em partes menores que podem ser processadas em paralelo, com isso pode-se analisar a informação e reagrupar as pequenas partes para apresentar resultados.



Hadoop: Arquitetura e Ecossistema

1 **Framework Open-Source**

Processamento e armazenamento de grandes volumes.

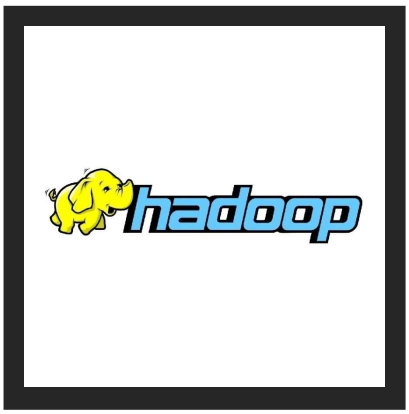
2 **Componentes Principais**

HDFS, YARN, MapReduce.

3 **Ecossistema**

Hive, Pig, HBase, Flume, Sqoop.

Ecossitema Hadoop



O ecossistema Hadoop fornece uma coleção cada vez maior de ferramentas e tecnologias, criadas especificamente para suavizar o desenvolvimento, a implantação e o suporte de soluções Big Data.

A plataforma acaba sendo confundida com um banco de dados, mas vai muito além. O também chamado Apache Software Hadoop é um ecossistema completo para computação para comportar o processamento de muitos dados em alta velocidade.

Componentes Principais do Hadoop

HDFS

Armazena dados em blocos distribuídos.

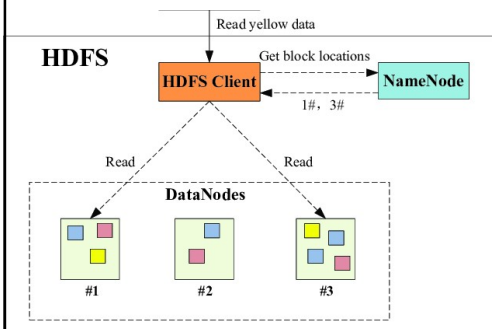
YARN

Gerencia recursos para aplicativos distribuídos.

MapReduce

Processamento paralelo de grandes volumes.

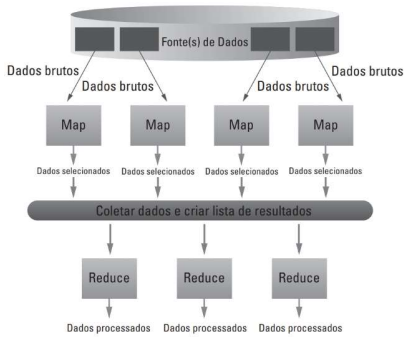
Hadoop Componentes



Hadoop File System (HDFS)

O HDFS é responsável pelo armazenamento distribuído e pela clusterização de computadores que suportarão a guarda dos dados, utilizando grandes blocos de memória. Esse sistema gerencia o disco das máquinas que formam o cluster, além de servir para a leitura e a gravação dos dados. Tem como características: escalabilidade para armazenar volumes expressivos de dados; tolerância a falhas e redirecionamento automático de dados; confiabilidade fornecida pela geração de cópias de dados; portabilidade entre hardwares e sistemas similares.

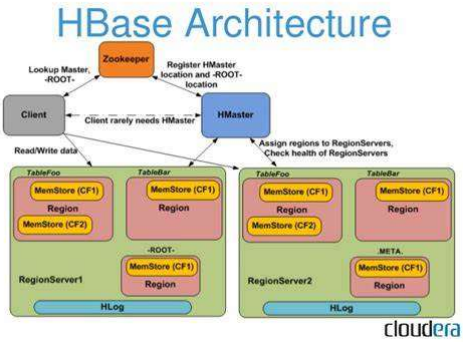
Hadoop Componentes



MapReduce

Gerencia o processamento dos dados em ambiente de cluster. É um modelo de programação desenvolvido para processar em larga escala, tendo como bases o mapeamento (map) e a redução (reduce). É caracterizado pela: flexibilidade para processar dados de diferentes tipos e formatos; acessibilidade para suportar diversas linguagens de programação; confiabilidade para permitir a execução de jobs em paralelo, sem perda de desempenho.

Hadoop Componentes



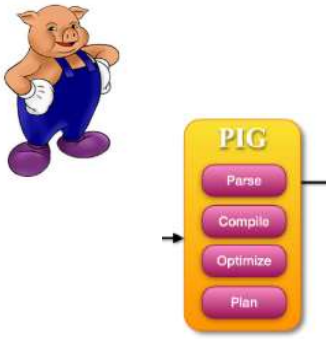
HBase

É um banco de dados não relacional e orientado a colunas, projetado para lidar com grandes conjuntos de dados. É conhecido como a base de dados oficial do Hadoop.

Oozie

Trata-se de um sistema de agendamento de workflow utilizado para organizar e gerenciar os jobs simultâneos enviados pelo Map Reduce.

Hadoop Componentes



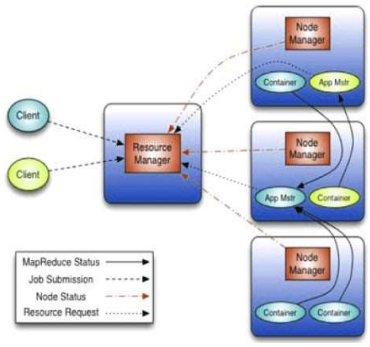
Pig

É uma linguagem de procedimentos de alto nível para consultar conjuntos de dados que, além de grandes, são semiestruturados — duas qualidades que tornam a análise de dados muito mais complexa.

Sqooq

A responsabilidade desse projeto é importar e exportar dados de bases relacionais.

Hadoop Componentes



ZooKeeper

É um serviço que gerencia conjuntos de clusters por meio de uma coordenação distribuída.

Yarn

Ao se conectar com o HDFS, o Yarn gerencia recursos por meio de clusters, além de realizar agendamentos de recursos.

Hadoop Componentes

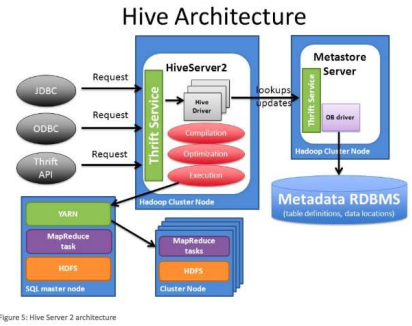


Figure 5: Hive Server 2 architecture

Hive

É um conversor SQL em MapReduces. Possibilita que usuários de negócio e analistas de dados usem análises preferenciais, relatórios e ferramentas de exposição de dados.

Spark

Essa ferramenta de Big Data é capaz de processar grandes conjuntos de dados. Em relação ao MapReduce, o ganho de velocidade de processamento é 100 vezes maior.

MapReduce

MapReduce: Conceito e Funcionamento

Modelo de Programação

Processa grandes conjuntos de dados em paralelo.

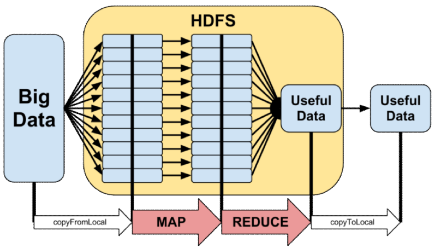
Map (Mapeamento)

Divide o problema em pequenas tarefas.

Reduce (Redução)

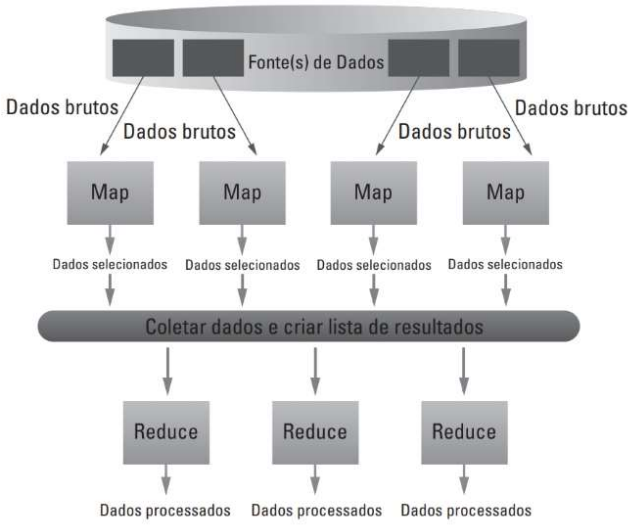
Junta resultados parciais e agrupa.

MapReduce

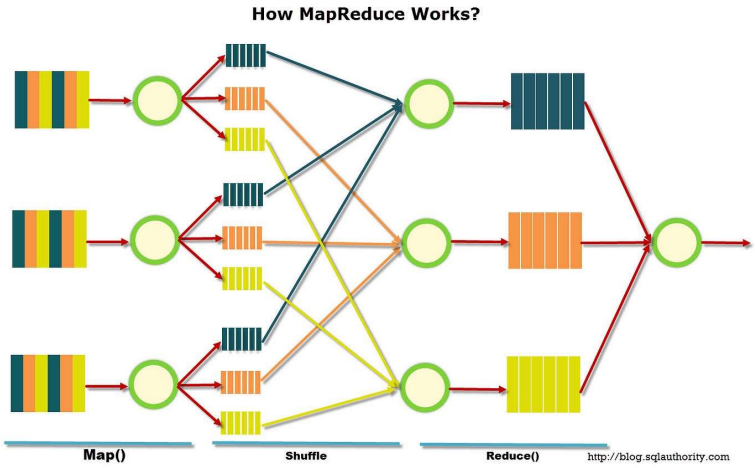


É uma estrutura de software que possibilita que desenvolvedores escrevam programas que possam processar quantidades massivas de dados desestruturados em paralelo, através de um grupo distribuído de processadores. Foi desenvolvido nos anos 2000 por engenheiros do Google pensando nas necessidades do Big Data. Esses engenheiros determinaram que, se o trabalho pudesse ser distribuído através de computadores baratos e, então, conectados à rede em forma de um "agrupamento", poderiam resolver o problema.

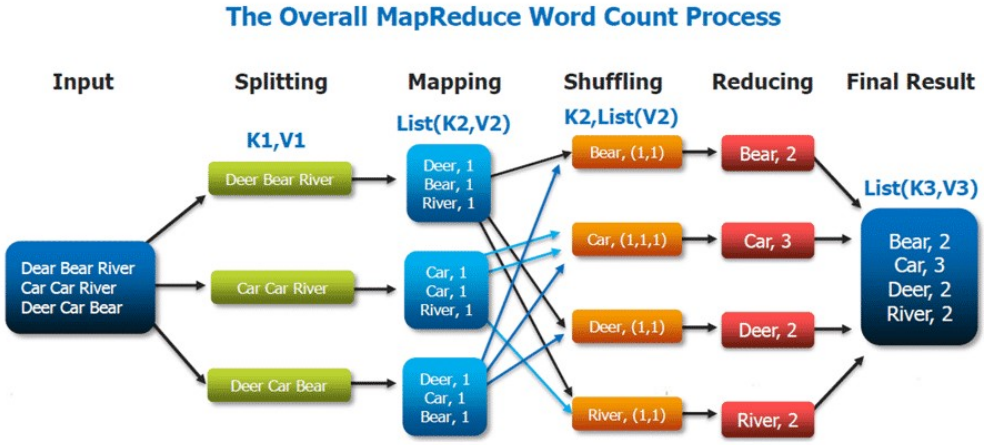
Estrutura do MapReduce



Funcionamento do MapReduce



Exemplo do Processo do MapReduce



Apache Spark

Apache Spark: Características e Aplicações

Características

Processamento em memória, resiliência, APIs múltiplas.

Vantagens

Mais rápido que MapReduce, tolerância a falhas.

Aplicações

Análise em tempo real, machine learning, SQL.

Origem do Apache Spark

Foi desenvolvido pelo grupo de pesquisa AMP-Lab, da Universidade da Califórnia, em Berkeley, é um framework de código aberto mantido pela Apache Foundation, para processamento distribuído de dados em larga escala, rápido e robusto, além de servir para propósitos gerais, que roda tanto de maneira independente quanto acessando um cluster Hadoop.

Objetivo do Apache Spark

Ele foi projetado para cobrir uma grande variedade de carga de trabalho, o que inclui aplicações com processamento em lotes, algoritmos interativos, consultas interativas e streaming, tornando mais simples e barato combinar diferentes tipos de processamento em um único mecanismo, o que geralmente é necessário nas estruturas de análise de dados em produção, sem precisar da oneração de gerenciar e manter ferramentas separadas.

Recursos do Apache Spark

SparkSQL é a ferramenta que possibilita a utilização de SQL para consultas e processamento de dados sobre o Spark.

Spark Streaming é a ferramenta que possibilita o processamento de fluxos de dados em tempo real.

MLlib é a biblioteca de aprendizagem de máquina que possui diferentes algoritmos para diferentes atividades, como a de clustering ou análise de agrupamento de dados.

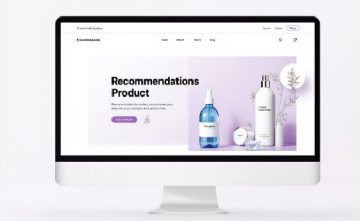
GraphX é a ferramenta que realiza o processamento sobre os grafos.

Vantagens do Apache Spark

Uma das vantagens do Spark em relação ao Hadoop é que seus módulos funcionam integrados na própria ferramenta, tendo em vista que o Hadoop possui módulos integrados, mas também pode fazer uso de ferramentas acopladas para compor o ecossistema.

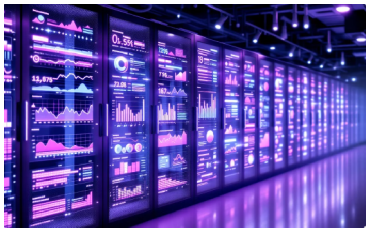
Spark podem ter interface com as linguagens: Scala, Python, R e Java.

Exemplo de Aplicação do Apache Spark



Recomendação de Produtos

Processa dados de transações para recomendações personalizadas.



Análise de Logs

Detecta padrões de uso e problemas em tempo real.

Processamento de Dados

Transformação de dados brutos em insights acionáveis.

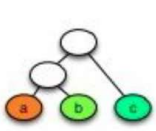
Parquet

Armazenamento Colunar

PARQUET - Estrutura de dados otimizada para retorno COLUNAR.

Otimização que comprime os dados (dados semelhantes na mesma coluna/atributos são considerados redundantes e não precisam ter toda representação repetida). Com isso otimiza o tempo de leitura dos dados armazenados.

Estrutura do Armazenamento Colunar



Nested schema

Logical table representation

a	b	c
a1	b1	c1
a2	b2	c2
a3	b3	c3
a4	b4	c4
a5	b5	c5

Row layout

a1	b1	c1	a2	b2	c2	a3	b3	c3	a4	b4	c4	a5	b5	c5

Column layout

a1	a2	a3	a4	a5	b1	b2	b3	b4	b5	c1	c2	c3	c4	c5

encoding

encoded chunk	encoded chunk	encoded chunk

Linear

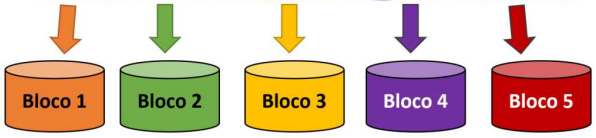
A1	B1	C1	A2	B2	C2	A3	B3	C3

Colunar

A1	A2	A3	B1	B2	B3	C1	C2	C3

Armazenamento Colunar

ID	NOME	IDADE	CIDADE	PAÍS
100	Maria	45	New York	EUA
101	José	32	São Paulo	Brasil
102	Paulo	78	Maringá	Brasil
103	Beto	17	Santos	Brasil
104	Julia	22	Lisboa	Portugal
105	Amanda	25	Paris	França



Contato

eferlin@live.com

(BLOG) professorferlin.blogspot.com

(SITE) professorferlin.com.br

(YOUTUBE) ProfEdsonPedroFerlin