# **HADOOP**

O Hadoop é um framework, ou seja, um conjunto de softwares ou plataforma de desenvolvimento mantido pela Fundação Apache. Ele contém um grupo de ferramentais que o auxiliam no seu objetivo de armazenar e processar arquivos de maneira distribuída em larga escala.

Ao se fazer uma busca sobre tecnologias de Big Data, o principal termo retornado é o apache Hadoop e suas ferramantas relacionadas como Hbase, Flume, Hive entre outros.

É considerado uma plataforma altamente escalável de processamento distribuído que tem como proposta armazenar e processar grandes quantidades de dados entre vários clusters de computadores que operam paralelamente. Ele consegue detectar falhas automaticamente e uma das grandes vantagens de usa-lo é por ele ser open source. Um dos exemplos que podem ser feitos com Hadoop é um sistema de recomendação em um site de e-commerce.

[complemento] O que é Hadoop part 1 <https://www.youtube.com/watch?v=qX5edsUWadE>

## História

O Hadoop foi criado em 2002 por Doug Cutting, com o projeto Apache Nutch. Durante os anos de 2002 e 2004, o projeto Nutch demonstrou grande eficácia ao usar 4 nós e vasculhar 100 milhões de páginas web, mas a arquitetura usada não conseguiria suportar bilhões de páginas. Foi com o artigo publicado pelo Google em 2003 que surgiu a arquitetura distribuída de um sistema de arquivos denominado GFS (Google File System) e com essa arquitetura, o problema das bilhões de páginas desaparecia.

[saiba mais] Google File System < http://www.lisha.ufsc.br/teaching/os/ine5412-2008-2/work/gfs.pdf>

Entre os anos de 2004 e 2006, a versão open source do Apache Nutch começou a ser criada e ganhou forma, juntamente com a implementação do MapReduce, que surgiu em 2004 a partir de um artigo publicado pelo Google.

Em 2006, um sub-projeto do Apache Nutch foi criado e batizado de Hadoop, que é a implementação que temos hoje atualmente.

## Estrutura do Hadoop

Dentro do Hadoop, há dois componentes principais: MapReduce e HDFS (Hadoop File .

O MapReduce é um paradigma de programação criado pelo Google para analisar e armazenar grandes conjuntos de dados. Nele, cada tarefa é processada paralelamente no cluster e possui duas funções principais: Map e Reduce. O Map recebe um o conjunto de dados do arquivo em qualquer estrutura e o transforma em um conjunto de dados onde os elementos são quebrados em conjuntos organizados por chave/valor. Já o Reduce recebe a saída do Map e combina essas tuplas em um conjunto menor de tuplas. A Figura 1 ilustra a estrutura do Hadoop, com foco em MapReduce e HDFS. Note que o foco do uso desta técnica está na organização dos dados de uma forma que facilite o processamento e geração do resultado final.

Uma tupla é um agregado de dados que possuem uma quantidade pré-estabelecida de componentes. Exemplos:

- Datas: dia, mês e ano. (20, 01, 2018)

- Uma pessoa: nome, estado civil, idade, altura. (“Maria Rosa”, “solteira”, 24, 1.60)

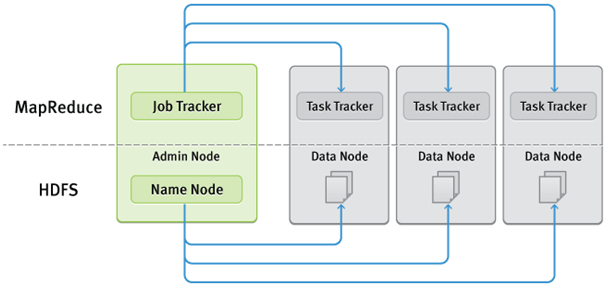


Figura 1 Estrutura do Apache Hadoop.. Fonte: https://imasters.com.br/tecnologia/redes-e-servidores/big-data-e-hadoop-o-que-e-tudo-isso/?trace=1519021197

O HDFS é o sistema de armazenamento do Hadoop. Cada nó no cluster é conectado entre si, o que permite que os arquivos sejam distribuídos, e acessados de forma contínua. Ele é tolerante a falhas, o acesso é de alto rendimento.

Frisando que o HDFS armazena arquivos, ele não é um sistema gerenciador de banco de dados. A Figura 2 ilustra toda a estrutura do HDFS, com seus componentes.

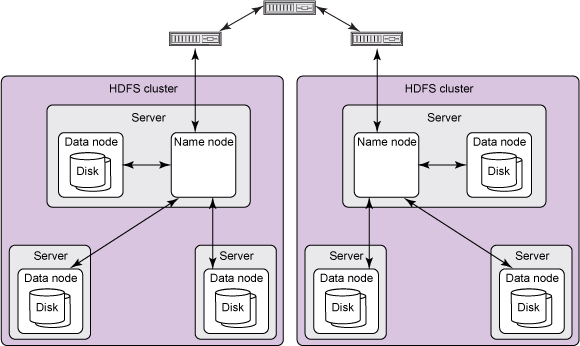


Figura 2 Estrutura do HDFS. Fonte: https://www.ibm.com/developerworks/br/library/wa-introhdfs/index.html#fig1

Exemplo de comando do HDFS**:**

$ Hadoop fs –put /path of the required file /path in HDFS where to save the file

O comando acima especifica onde o arquivo deve ser salvo.

[saiba mais] o que é hadoop parte 2

< https://www.youtube.com/watch?v=HiM8ALNEDPM>

## Benefícios do uso do Hadoop

Usar o Hadoop tem grandes vantagens no armazenamento e gerenciamento de grandes quantidades de dados. A seguir seguem alguns benefícios:

* Baixo custo: É uma plataforma open source.
* Flexibilidade: os conjuntos de dados podem estar em qualquer formato, não é necessário formatar os dados em uma estrutura específica.
* Confiabilidade: Se algum nó dos clusters falha, o Hadoop automaticamente redireciona esse nó para outro cluster.
* Desempenho: Processamento paralelo e distribuído.

## Desvantagens

* Segurança: o framework foi escrito em sua totalidade em Java, uma linguagem de programação que tem algumas brechas na segurança.
* Sem capacidade para suportar bases de dados muito pequenas: o Hadoop não consegue lidar com bases de dados muito pequenas, na hora de ler e abrir os arquivos.
* Estabilidade: por ser uma plataforma open-source, o Hadoop pode apresentar alguma instabilidade durante sua execução.

## HBASE

O Apache HBase também faz parte do Hadoop, é um banco de dados NoSQL (não relacional). Ele junta a escabilidade do Hadoop rodando em cima do HDFS e precisa de uma abordagem diferente para modelar os dados. O HBase define um modelo com quatro dimensões, como pode ser visto na figura abaixo.

* Row Key: cada linha tem uma chave única da linha.
* Column Family: os dados dentro da linha são organizados em famílias de colunas.
* Column Qualifier: as famílias de colunas definem as colunas atuais, que são chamadas de column qualifiers.
* Version: cada coluna tem um número de versões configurável.

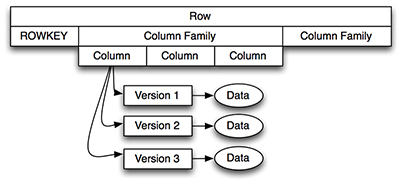


Figura 3 Estrutura do Apache HBase. Fonte: http://www.informit.com/articles/article.aspx?p=2253412

## APACHE FLUME

O Apache Flume é uma das ferramentas padrão pertencentes ao Apache Hadoop. Com ele é possível coletar logs e transportar quantidades grandes de dados para o armazenamento dentro do HDFS, que foi apresentado anteriormente. O Flume também é altamente escalável e possui um mecanismo de tolerância a falhas.

Um bom exemplo é o da figura abaixo, onde os geradores de dados geram dados que são coletados pelos agentes Flume. Então, um coletor de de dados coleta os dados dos agentes, que são armazenados dentro do HDFS ou HBase.

Agente 1

Agente 3

HDFS ou HBase

Coletor de Dados

Gerador de dados

Agente 2

**Flume**

Figura 4 Estrutura do Apache Flume. Fonte: Traduzido e adaptado de https://www.tutorialspoint.com/apache\_flume/apache\_flume\_architecture.htm

## Apache Hive

O Hive é o sistema de armazém de dados do Hadoop, ele permite o resumo dos dados, consultas e analises, para tanto utiliza-se a linguagem HiveQL, que é semelhante ao SQL.

Com o Hive pode-se projetar uma grande estrutura de dados e depois utiliza-la sem necessitar ter o conhecimento da linguagem de programa Java ou do MapReduce.

## Apache Kafka

O Kafka é um sistema de envio de mensagens que permite a criação de aplicativos em tempo real usando os dados em streaming, como por exemplo transações financeiras, logs de aplicativos. Ele recebe os dados e o envia para o aplicativo de processamento como por exemplo o Spark.

[saiba mais] Big data com Apache Spark – Part 1: Introdução < https://www.infoq.com/br/articles/apache-spark-introduction>

## Consideração Finais

O Apache Hadoop não é um banco de dados em si, é um sistema de gerenciamento de arquivos distribuídos, contudo é a principal ferramenta tecnológica utilizada para processar e armazenar grandes volumes de dados. Por isso merecem serem referenciados lembrado quando se trada de tecnologias de armazenamento de dados não convencionais.

Contudo seu uso não é indicado usar o Hadoodp em poucos dados, por seria algo como usar uma espingarda de matar elefante em uma formiga, ou seja, o processamento se tornaria custoso e teria o mesmo tempo de resposta caso estivesse trabalhando com um grande volume de dados, além de não ser recomendado ficar manipulando os arquivos armazenados a qualquer instante de tempo.

Nesses casos onde os dados sofreram constante atualizações ou começarão em uma escala menor indica-se o uso de Banco de dados tanto os tradicionais quanto os NoSQL, dependendo muito da natureza do problema a ser tratado.

A popularização do Hadoop se deve ao fato de sua escalabilidade horizontal ser facilmente configurável e nó principal e realizar a tarefa de replicação com os nós secundários mantendo a consistência dos arquivos armazenados. Note que aqui se fala consistência dos arquivos e não dos dados, pois mesmo que o arquivo esteja particionado nos nós da rede, após o processamento do MapReduce ele conterá todos os dados originais.

# Material Complementar

Inserir os materiais complementares para aprofundamento dos conceitos abordados na unidade.

**Importante**: Mínimo 04 indicações — livros, sites, vídeos, artigos, repertagens

<https://www.ibm.com/developerworks/br/library/wa-introhdfs/index.html#fig1>

<https://www.tutorialspoint.com/hadoop/hadoop_mapreduce.htm>

<https://www.ibm.com/analytics/hadoop>

<https://www.cetax.com.br/blog/apache-hadoop/>

<https://www.cetax.com.br/blog/o-que-e-hadoop/>

<https://imasters.com.br/tecnologia/redes-e-servidores/big-data-e-hadoop-o-que-e-tudo-isso/?trace=1519021197>

<https://www.sas.com/en_th/insights/big-data/hadoop.html>

<http://www.balasubramanyamlanka.com/history-of-hadoop/>

<https://www.tutorialspoint.com/apache_flume/apache_flume_architecture.htm>

<http://www.informit.com/articles/article.aspx?p=2253412>

<https://br.hortonworks.com/apache/flume/>

**Referencias**

Ahmed, F. D., Jaber, A. N., Majid, M. B. A. and Ahmad, M. S. "Agent-based Big Data Analytics in retailing: A case study." Software Engineering and Computer Systems (ICSECS), 2015 4th International Conference on. IEEE, 2015.

Essa, Youssef M., Gamal Attiya, and Ayman El-Sayed. “Mobile agent based new framework for improving big data analysis”. Cloud Computing and Big Data (CloudCom-Asia), 2013 International Conference on. IEEE, 2013.

FLUME, Apache Flume, site: https://flume.apache.org. Acessado em janeiro/2018

HADOOP, Apache Hadoop, site: http://hadoop.apache.org/. Acessado em janeiro/2018

HBASE, Apache HBase, site: https://hbase.apache.org. Acesado em janeiro 2018

HIVE, Apache Hive, site: https://hive.apache.org. Acessado em janeiro 2018

HIVE2, Saiba mais sobre o Apache Hive e o HiveQL no Azure HDInsight , https://docs.microsoft.com/pt-br/azure/hdinsight/hadoop/hdinsight-use-hive> acessado em janeiro 2018.

KAFKA, Apache Kafka site: <https://kafka.apache.org>. Acessado em janeiro 2018

Kamala, V. R., and L. MaryGladence. “An optimal approach for social data analysis in Big Data”. In: Computation of Power, Energy Information and Commuincation (ICCPEIC), 2015 International Conference on. IEEE, 2015