BIG DATA

AULA 3

Prof. Armando Kolbe Júnior

CONVERSA INICIAL

O tema desta aula enfeixa todo um conjunto de programas e procedimentos que utilizam uma das mais democráticas formas de acesso a recursos computacionais, sua distribuição como *open source* – código aberto.

Não há custos de aquisição envolvidos e, após sua instalação, o produto está pronto para ser utilizado. O *software* cria uma estrutura flexível de modo que os usuários possam, sem necessidade de aprovação ou consulta, adicionar ou modificar o sistema instalado ou modificar o sistema de dados sempre que houver nova necessidade.

Aqui vale a ressalva feita sobre a instalação de novas versões, com mudanças sugeridas pela comunidade de usuários que trabalha com os programas instalados. Independentemente dessa ressalva, o sistema pode ser considerado um dos mais utilizados para atendimento da necessidade de criação de um *framework* (estrutura) diferenciado para operar atividades classificadas, como Big Data (mineração, armazenamento e recuperação de dados). Essa é a razão para seu tratamento diferenciado para fornecer armazenamento e processamento de dados.

CONTEXTUALIZANDO

O trabalho com esse conjunto de programas é coletivo e colaborativo entre usuários voluntários e outros que recebem incentivos financeiros, utilizando o produto para fins comerciais, o que não é bloqueado. Essa característica ativa de atuar como *software* aberto, podendo ser alterado de acordo com as necessidades dos usuários, é uma das razões de seu sucesso.

O sistema tem uma exigência, que é a utilização de um servidor Apache, e apresenta grande complexidade em seu desenvolvimento. Uma vez efetuada alguma alteração, não há obrigatoriedade da comunidade em prestar suporte. O ambiente criado é altamente flexível e recebe colaborações de uma comunidade altamente especializada, que conta com o apoio de doações para manutenção do sistema em seus provedores. Essa proposta é considerada um grande avanço na ainda temerária migração para o Big Data e para a computação em nuvem (atualmente quase que indissolúvel).

TEMA 1 – HADOOP

Em termos de uma definição padrão, adotamos a proposta por White (2015). O autor considera que esse sistema de código aberto coloca à disposição dos usuários uma ferramenta potente para o desenvolvimento das atividades de análise, secundando a criação das grandes bases de dados, que atuam como repositórios de dados, informações e arquivos de metadados[1] relacionados. A razão de esse sistema estar sendo estudado juntamente com o advento do fenômeno Big Data é a sua capacidade de processamento das altas cargas de trabalho que as atividades de pesquisa utilizam no seu afã de produção de informações estruturadas de acordo com a necessidade dos usuários.

Para tanto, aumenta-se a capacidade de processamento resultante de uma estrutura com crescimento e evolução não esperada por muitos, o que é configurado com a utilização de *clusters*[2]. Eles possibilitam que a estrutura montada seja aumentada com elevado grau de escalabilidade horizontal, que representa a adição do número de servidores reunidos, ao invés de utilização de escalabilidade vertical, representada pela compra e troca da estrutura atual por computadores mais rápidos e que suportam maior volume.

Os recursos de uso de CPU são igualmente atendidos. A estrutura criada apresenta algo grau de durabilidade e disponibilidade colocada via utilização de diversos níveis de acesso com a utilização de palavras-chave. A variedade de chaves utilizadas permite recuperar a informação de acordo com diferentes visões usuárias (por exemplo, se forem dados cadastrais de alguma pessoa, eles podem ser recuperados por estado de origem, idade, sexo, e assim por diante).

O modelo é desenvolvido, então, com a capacidade de desenvolvimento paralelo das mais diversas práticas analíticas computacionais. O volume de dados processados permite a utilização de um grande volume de dados com a criação dos armazéns de dados (*data warehouse*).

1.1 O que faz

As estruturas dos servidores Hadoop apresentam elevada velocidade e agilidade no tempo de resposta aos questionamentos analíticos. São como uma

[1] Metadados representam dados sobre outros dados que são utilizados para dar maior significado para a informação que está sendo trabalhada.
[2] *Cluster* é uma estrutura de computação escalável que cresce de acordo com necessidades contextuais, de forma que se obtenha a otimização nas atividades de manutenção de dados.

caixa preta para os usuários finais, devido à complexidade de seu projeto, mas não trazem dificuldades para o desenvolvimento das pesquisas e têm baixa complexidade administrativa. O produto pode ser criado em servidores internos ou em externos. Este segundo caso é o preferido, por conta dos contratos rigorosos estabelecidos entre a empresa e a indústria de *software*, garantindo a integridade e segurança dos dados, estruturação 24 x 7 x 365[3], que dá disponibilidade total aos dados.

O armazenamento espelhado[4] e sua utilização em tempo real dão a garantia de pequenas perdas, em caso de algum desastre acidental, considerando a lei de Moore[5], metáfora de utilização constante no mundo informático.

Vídeo

Confirme algumas das colocações deste tema acessando o *link*: <https://www.youtube.com/watch?v=qX5edsUWadE>.

Saiba mais

Em diversos pontos do material foi citado o tema *data warehouse*. Acesse o *link* e conheça o significado desse termo: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-530X2005000100006&lng=pt&nrm=iso>.

Pesquisa

Considere desenvolver uma pesquisa que lhe permita avaliar e justificar a diferença entre escalabilidade vertical e escalabilidade horizontal.

TEMA 2 – COMPONENTES DO HADOOP

Em uma visão externa, o Hadoop é um sistema de computação completo. Em seu interior, a estrutura se apresenta diferenciada e subdividida em quatro diferentes módulos, cada qual com a responsabilidade por uma tarefa essencial. A estrutura é similar à de outros sistemas computacionais que são desenhados para o desenvolvimento de atividades de análise.

[3] Essa estruturação se mantém ativa 24 horas por dia, 7 dias por semana e durante todos os dias do ano, e sua manutenção é em termo de contratos pactuados.
[4] No armazenamento espelhado, a mesma informação gravada em um dispositivo de armazenamento é replicada em outro dispositivo, como forma de garantia de preservação do dado que está sendo trabalhado em determinado momento.
[5] A lei de Moore é um tema de uso corrente em computação, utilizada com a conotação: se alguma atividade pode resultar em erro, certamente algum erro ocorrerá.

Veja quais são esses quatro módulos:

- módulo de distribuição do sistema de arquivos;
- módulo de desenvolvimento de atividades de Map Reduce;
- módulo Hadoop Common;
- módulo YARN.

Cada um deles desenvolve uma tarefa que será detalhada a seguir.

2.1 A distribuição do sistema de arquivos

O módulo de distribuição de arquivos tem ao seu encargo grandes responsabilidades. Ele permite que os dados sejam armazenados no formato mais simplificado e acessível possível, independentemente da plataforma ou sistema operacional em uso, pelo fato de o Hadoop criar o seu próprio sistema operacional, utilizado em suas tarefas. A camada do sistema Hadoop fica acima do sistema de arquivos do computador do usuário, e é isso o que libera o acesso utilizando qualquer *hardware* ou *software* presentes.

2.1.1 Map Reduce

O Map Reduce lê os dados do banco de dados e os coloca em um formato adequado para que a análise seja efetivada, no formato de um mapa, e desempenha as tarefas matemáticas. Tem a responsabilidade de exploração dos dados em qualquer formato.

2.1.2 Hadoop Common

Desenvolvido com a linguagem de programação Java, conversa com os sistemas computacionais que estão implantados (UNIX, Windows ou outros).

2.1.3 Módulo Yarn

O módulo final gerencia os recursos dos sistemas que armazenam os dados e que fazem a análise dos dados, devolvendo-os para que a informação estruturada seja estabelecida a partir dele.

Na bibliografia e na internet, poderão ser encontrados outros elementos que integram o *framework* do Hadoop, mas neste material nos atemos especificamente a esses quatro componentes.

Vídeo

Conheça um pouco da história do Hadoop neste vídeo: <https://www.youtube.com/watch?v=pcAdeZ5fLB8>.

Saiba mais

Leia um artigo com viés acadêmico sobre o Hadoop acessando o *link*: <https://www.cetax.com.br/blog/apache-hadoop/>.

Saiba mais sobre o Hadoop, tecnologia que sustenta as necessidades de armazenamento e recuperação de informações, acessando o *link*: <https://www.hostinger.com.br/tutoriais/o-que-e-apache>.

Pesquisa

Considere pesquisar e localizar um estudo de caso sobre o tema *componentes do Hadoop*, preparando uma síntese do que foi possível coletar contendo indicações de leituras complementares e endereços na grande rede.

TEMA 3 – MAP REDUCE

O Map Reduce é tratado aqui como um componente do Big Data e, em seu interior, um componente do Hadoop. A definição básica o considera um modelo de programação que permite que dados massivos (resultantes do Big Data) sejam recuperados em um algoritmo[6] paralelo e distribuído em um *cluster* de computadores interligados (veja a escalabilidade horizontal, já referenciada). Ele se tornou popular por ser a base para aplicações como o Facebook e Twitter, aumentando o número de programadores independentes ligados à plataforma.

É na escalabilidade horizontal (representada pela adição de nós complementares para facilitar o acesso às informações desejadas) que são resolvidos os tempos de acesso e escrita – ainda um problema sem solução para aumento de velocidade de gravação e leitura. O espelhamento, representado pela leitura e escrita paralela de dados, pode ser utilizado como uma das formas de diminuição do tempo de acesso.

A agilidade no tempo de obtenção de respostas é o objetivo a ser atingido. As perdas para regravação em caso de alguma pane no sistema são resolvidas pela replicação em diferentes pontos[7], garantindo a preservação digital.

[6] Algoritmos representam uma codificação que orienta para a execução de uma série de passos, que produz um ou mais resultados como saída de alguma rotina sendo executada.

[7] Essa atividade representa a redundância dos dados desenvolvida durante as atividades de manutenção da estrutura de banco de dados criada para armazenamento, com a finalidade de

3.1 Quando utilizar

A técnica é utilizada sempre que se mostrar necessário um *framework* para o processamento e armazenamento de dados, com a condição de um volume considerável e que justifique o investimento nessa técnica. O Map Reduce é considerado uma técnica de apoio e suporte ao processamento e armazenamento de dados massivos em *clusters* de programadores.

Vídeo

Assista ao vídeo <https://www.youtube.com/watch?v=rISmbGoO-cM> e aprenda um pouco mais sobre o Map Reduce.

Saiba mais

Conheça mais sobre o conceito de *clusters*, que foi citado diversas vezes durante as explicações do Big Data: <https://inmedia.com.br/projetos/cetax/apache-hadoop-tudo-o-que-voce-precisa-saber/>.

Pesquisa

Considere desenvolver uma pesquisa que lhe dê dados necessários para desenvolver um descritivo que apresente as vantagens da utilização do Map Reduce, com justificativa de suas colocações.

TEMA 4 – HADOOP: TIPOS DE DADOS

O estudo cria uma área denominada Hadoop HDFS – Hadoop Distribute File System[8]. O importante em obter detalhes sobre qual o melhor tipo de dado a ser utilizado está no fato de que a velocidade de processamento e devolução das informações pode ser afetada por essa escolha.

Para evitar tal problema, o Hadoop trabalha com diferentes tipos de armazenamento de dados, visando facilitar o trabalho dos operadores e usuários finais do sistema. Neste tema, veremos os tipos de dados com que a tecnologia Hadoop trabalha.

prevenção contra perdas resultantes de qualquer acidente de percurso que possa ocorrer durante o processamento das informações armazenadas.

[8] Sistema de arquivos distribuídos Hadoop.

4.1 Nomenclatura e significado

É possível demarcar os tipos montando uma tabela com a nomenclatura e significado:

Nome	Descritivo
Dados de texto estruturados	Existem formas mais sofisticadas de arquivos de texto com dados em alguns formulários padronizados, como arquivos CSV, TSV, XML ou JSON. No Hadoop, não há um formato de entrada construído para manipular arquivos XML ou JSON. Por não haver *tags* de início ou finalização em JSON, há dificuldade em trabalhar com arquivos JSON, pois é difícil dividir esses tipos de arquivos. São elementos que estão armazenados em grandes estruturas de bancos de dados de forma organizada, divididos em tabelas apresentadas em uma estrutura compreensível a "olho nu", devido à sua organização tabular, na qual o rótulo (*label* ou nome) de cada coluna pode indicar para que cada elemento está sendo utilizado (cuidado a ser tomado na montagem do dicionário de dados da empresa). Por exemplo, o registro de um cliente é dividido em colunas, com os rótulos *cidade*, *estado*, *país de origem* e tantos outros quanto forem necessários. Os arquivos binários são quaisquer arquivos cujos dados não estão armazenados de forma textual, tais como: fotografias, programas, vídeos, áudios etc.
Arquivos binários	Para a maioria dos casos, o armazenamento de arquivos binários, como imagens, vídeos, etc. no formato container como *sequence*[9] *files* é o preferido, mas para arquivos binários muito grandes, é preferível armazenar os arquivos como estão.
Sequence files – formatos específicos para Big Data	Existem muitos formatos de arquivo específicos para Big Data como estruturas de dados baseadas em arquivos, por exemplo, *sequence files*, formatos de serialização como Avro, formatos colunares como Parquet ou ORC.
Arquivos sequenciais	Esses arquivos contêm dados como pares binários de chave-valor. Existem mais 3 formatos: • Descompactado – sem compactação. • Registro compactado – os registros são compactados quando adicionados ao arquivo. • Bloco compactado – aguarda até que os dados cheguem ao tamanho do bloco para depois serem compactados. O bloco compactado fornece melhor compactação do que apenas registro compactado. Um bloco se refere a um grupo de registros compactados juntos dentro de um bloco HDFS – Hadhoop File System. Pode haver vários blocos de arquivos sequenciais dentro de um bloco HDFS.

Fonte: Adaptado de Eadline, 2015.

[9] Um arquivo de sequência (*sequence file*) utiliza um formato de arquivo que fornece estrutura de dados persistente para pares de valor de chaves binárias, que identificam seu início e final. Um exemplo clássico são textos transferidos entre diferentes componentes de algum sistema computacional, delimitados por caracteres especiais, que não podem fazer parte integrante do texto.

Vídeo

Acesse o vídeo a seguir, sobre o tema desta aula: <https://www.youtube.com/watch?v=I80l0X760ao>.

Pesquisa

Considere desenvolver uma pesquisa sobre os tipos de dados, escolha um modelo específico e pesquise vantagens e desvantagens sobre ele.

TEMA 5 – EXEMPLO DE USO DO HADOOP

Cada vez mais é gerado um volume gigantesco de dados pelas organizações, e em alguns casos, como da Google, chegam a gerar mais do que 20 petabytes. Como visto, a maioria de soluções de Big Data é criada para de alguma forma acelerar o processo de tomada de decisão, otimizando os processos para que haja redução de custo apoiado pela agilidade do tempo de processamento dos servidores com propósitos de entendimento de como acontece o comportamento dos consumidores e as necessidades das empresas.

A resolução de alguns problemas complexos, como o volume de dados cada vez maior, sejam eles estruturados ou não, torna-se um grande desafio, que requer ferramentas e técnicas de análises adequadas na extração da informação para as organizações. Nesse cenário, o Hadoop proporciona uma forma segura e ágil para a manipulação de bases de dados que são utilizadas em aplicações de Big Data.

Saiba mais

O *site* Meu Artigo cita exemplos de uso do Hadoop. Algumas organizações que se apropriaram dessa plataforma são NASA, Twitter e Netflix: <https://meuartigo.brasilescola.uol.com.br/administracao/hadoop-sua-importancia-no-mundo-atual.htm>.

5.1 O problema

O Hadoop não é apenas um meio de armazenamento, e sim um conjunto de ferramentas responsável por revolucionar a maneira de armazenar grandes volumes de dados. Mesmo sendo muito elogiado, ele apresenta inúmeros problemas, entre eles a fiabilidade[10] e o monitoramento de tarefas.

[10] A fiabilidade se refere ao nível de confiança que a estrutura deve apresentar e é pactuada em cláusulas de contratos de nível de serviço (SLA – Service Level Agreement – acordo de nível de

Uma das propriedades interessantes no Hadoop é a computação em que o dado está. A transição de dados entre computadores diferentes em computação distribuída torna-se um dos maiores gargalos. Outro problema do Hadoop que é que o modelo de programação Map Reduce não é a solução ideal para tudo, e com isso surgiram diversos sistemas especializados.

Apesar de ser um dos genéricos mais utilizados com Hadoop, o Spark pode apresentar problemas quando o banco de dados ficar muito grande, pois no caso de ser necessário reconstruir os RDD em alguma falha, irá levar muito tempo. Uma maneira de tentar evitar esse problema é a opção de *checkpoint*, que salva os dados diretamente no HDFS, eliminando toda a linhagem armazenada e tendo ainda tolerância a falhas.

5.2 A solução

O Spark[11] foi uma das ferramentas que substituiu muitos desses sistemas genéricos criados em cima do Hadoop. Para a transição entre computadores diferentes, o Hadoop é baseado no modelo de programação Map Reduce. No Map Reduce, os dados são particionados e armazenados em diferentes nós, processados por duas funções, a função *mapper* e a função *reducer*.

> A função Mapper irá ser executada em cada um dos pedaços e irá emitir várias tuplas de chave - valor. Após a conclusão do processo de Mapper, há o processo de Shuffle, responsável por ordenar as tuplas de todos os processos que executaram os mappers e distribuir todas as chaves idênticas para o mesmo reducer, que irá receber estes valores e executar uma outra função nestes dados. A ideia aqui é apenas dar uma breve descrição do modelo de Map Reduce, sem entrar em muitos detalhes. Vale ressaltar que o Hadoop utiliza apenas programação Map Reduce (Relvas, 2016, p. 2).

O Spark é um *framework* para computação em *cluster*, elaborado em Scala, e surgiu para resolver dois problemas do Hadoop. O primeiro é a necessidade de persistir em disco após cada *job*. O segundo é que é de difícil utilização para tarefas interativas. É um *dataset* distribuído nos nós do *cluster* e suporta várias operações, além de *map* e *reduce*. É um serviço utilizado para testar experimentos acadêmicos, executado em nuvem, no Hadoop, Apache Mesos, Kubernetes ou autônomo, podendo acessar diversas fontes de dados.

serviço). Já o monitoramento de tarefas é função de auditoria de dados, incorporada como necessidade de validação das estruturas. Ambas são condições inerentes ao uso do Big Data pelas empresas.

[11] Disponível em: <http://spark.apache.org/>. Acesso em: 3 out. 2019.

Utilizamos o modelo de programação em Map Reduce por ser acíclico, o que faz com que seja muito útil no gerenciamento de falhas, pois se ocorrer a falha de uma função *mapper* ou *reducer*, necessitamos executar novamente esse pedaço. Não é aconselhado o uso desse modelo para reuso de dados, como os processos iterativos, muitos comuns em análise de dados. Esses processos iterativos usando Map Reduce necessitam do encadeamento de diversos processos, mas não se consegue aproveitar a saída de um *reducer* para a entrada em um *mapper* de forma automática. É necessário salvar a saída do reducer no HDFS para logo depois o próximo *mapper* conseguir acessar os dados, enfatizando o problema de alta latência presente no Hadoop.

Assim como o Spark, o Hadoop também foi pensado para ser escalável. Resumindo, quando o problema for muito grande, aumentar o número de computadores necessários para conseguir processar e reduzir o tempo será o suficiente. O ideal é escolher um parceiro para capacitar sua equipe, tanto na implantação do ambiente como no desenvolvimento do projeto.

Vídeo

Acesse o vídeo a seguir, que explica o que é o Hadoop: <https://www.youtube.com/watch?v=qX5edsUWadE>.

Entenda as diferenças entre Hadoop e Spark assistindo ao vídeo: <https://www.youtube.com/watch?v=fyEC24OCtBM>.

Saiba mais

Acesse o artigo a seguir e saiba mais sobre Hadoop e Spark. Qual se aplica melhor para sua empresa? <https://itforum.com.br/noticias/hadoop-ou-spark-veja-qual-se-aplica-melhor-para-sua-empresa/>.

Pesquisa

Encontre um estudo de caso com a aplicação de técnicas Hadoop.

TROCANDO IDEIAS

Discuta aspectos nos quais encontrou dificuldade nesta aula. Procure instigar seus colegas a desenvolver uma participação ativa para efetivar a aprendizagem colaborativa e cooperativa.

O tratamento deste assunto pode ter tanta complexidade quanto desejado. Durante o desenvolvimento da aula, buscamos, dentro do possível, esconder as complexidades específicas de uma área com alto teor tecnológico.

As pesquisas se refinam cada vez mais e, rondando à espera de uma oportunidade para entrar em ação, está a IA – Inteligência Artificial.

NA PRÁTICA

Faça uma pesquisa sobre um exemplo de uso do Hadoop, levantando as complexidades que o estudo escolhido pode trazer.

FINALIZANDO

As interfaces gráficas desenhadas se encarregam de diminuir o nível de complexidade. Para os usuários finais, é como se estivessem desenvolvendo uma pesquisa comum, uma Data Query, que por mais complexa que possa ser, se comunica com o usuário por meio de interfaces amigas (GUI – Graphical User Interface).

O tratamento do tema não se esgota nesta aula. Na continuidade dos estudos, conheceremos aspectos que envolvem a criatividade, isto é, como as empresas utilizam a complexidade de programação para atender a diferentes solicitações, relacionadas à necessidade de informações em um ritmo crescente.

REFERÊNCIAS

EADLINE, D. **Hadoop 2 quick-start guide**: learn the essentials of Big Data computing in the Apache Hadoop 2 ecosystem. Boston: Addison-Wesley, 2015.

RELVAS, C. E. M. Apache Spark. **USP**, 2015. Disponível em: <https://www.ime.usp.br/~gold/cursos/2015/MAC5742/reports/ApacheSpark.pdf >. Acesso em: 3 out. 2019.

WHITE, T. **Hadoop**: the definitive guide: storage and analysis at internet scale. Nova York: O'Reilly Media, 2015.