

# Cluster de Big Data com Apache Hadoop: Um Mapeamento Sistemático da Literatura

João Victor Tabosa de Souza<sup>1</sup>, Paulo do Amaral Costa<sup>1</sup>

<sup>1</sup>Coordenadoria de Análise e Desenvolvimento de Sistemas  
Instituto Federal de Sergipe (IFS) – Aracaju, SE – Brasil

joao.souza077@academico.ifs.edu.br, paulo.amaral@ifs.edu.br

**Abstract.** *The continuous and growing demand for large-scale data processing solutions, due to the exponential increase in digital data from increasingly ubiquitous computing, requires efficient and already consolidated technologies to handle large volumes of data, such as the Apache Hadoop framework. This article consists of a Systematic Literature Mapping (SLM) of works published in the last 10 years on Big Data clusters that used Hadoop, involving the use of Hard Disk Drives (HDDs) or Solid State Drives (SSDs), in physical or virtualized environments, with the aim of answering five research questions. This search in the scientific literature found that there were few documents, especially focused on these environmental scenarios, that established any comparative performance relationship. There was a balance of experiments carried out in virtualized and physical environments; however, the main application used in the performance tests was Terasort with nine mentions, followed by WordCount with only four.*

**Resumo.** *A contínua e crescente demanda por soluções de processamento de dados em larga escala, devido ao aumento exponencial de dados digitais provenientes de uma computação cada vez mais ubíqua, exige tecnologias eficientes e já consolidadas, para lidar com grandes volumes de dados, como é o caso do framework Apache Hadoop. Este artigo consiste em um Mapeamento Sistemático da Literatura (MSL) de trabalhos publicados nos últimos 10 anos, sobre clusters de Big Data que utilizaram o Hadoop, envolvendo o uso de Hard Disk Drives (HDDs) ou de Solid State Drives (SSDs), em ambientes físicos ou virtualizados, com o intuito de responder a cinco questões de pesquisa. Essa busca na literatura científica constatou a existência de poucos documentos, especialmente focados nesses cenários ambientais que estabeleceram alguma relação comparativa de desempenho. Houve um equilíbrio de experimentos realizados em ambientes virtualizados e físicos, no entanto, o principal aplicativo utilizado nos testes de desempenho foi o Terasort com nove menções, seguido do WordCount com apenas quatro.*

## 1. Introdução

Em 2017, [Reinsel et al. 2017] já previam uma geração e processamento de novos dados crescentes de forma exponencial, principalmente por causa das centenas de milhares de dispositivos conectados à Internet, gerando uma enxurrada de novas informações todos os dias, estimando-se que, até o ano que vem, 2025, o volume global de dados digitais atingiria a impressionante marca de 175 zettabytes. O termo “Big Data”, portanto, face à

irrefreável produção de dados, continua causando desafios para infraestruturas de Tecnologia de Informação, visto que os métodos tradicionais de processamento e análise de dados não conseguem lidar, de maneira eficiente, com um volume massivo de informações. O conceito de Big Data, apesar de ser um tanto quanto abstrato, caracteriza-se pela forma de gerenciar e analisar dados em uma escala quase incomensurável.

Vários autores, no entanto, representam essa caracterização atribuindo-lhes, normalmente, à satisfação de 4 ou 5 V's. Como é o caso de [Saxena and Kumar 2014], que apresenta uma definição de Big Data com cinco V's (Volume, Velocidade, Variedade, Veracidade e Valor) e de [Gupta et al. 2022] que apresenta uma definição com 4 V's (Volume, Velocidade, Variedade e Veracidade). O volume se refere à quantidade de dados produzidos. A velocidade, à rapidez do processamento de dados; a variedade, aos diversos formatos em que os dados aparecem; à veracidade da qualidade e confiabilidade desses dados e o seu valor, no qual [Saxena and Kumar 2014] considera como o mais importante, porque diz respeito à capacidade de transformar dados brutos em conhecimento aplicável. Por sua vez, [Gupta et al. 2022] não considera o quinto V, já que, para ele, valor é tido como uma característica implícita.

Na era das tecnologias de virtualização a pleno vapor, os ambientes virtuais também têm-se tornado mais viáveis, exigindo também a comparação entre HDDs e SSDs em suas versões virtuais. Já que as características de cada tipo de armazenamento influenciam diretamente a performance das operações, como o tempo de resposta e a capacidade de processamento, é fundamental entender esses impactos em diferentes cenários.

A relevância desse mapeamento decorre da constante demanda por otimização de recursos computacionais, onde a eficiência e a escalabilidade podem gerar significativas reduções nos custos operacionais e afetar negativamente a competitividade das empresas. Portanto, o principal problema motivador dessa pesquisa foi o de investigar, na literatura, se há registros que apontem diferenças significativas de performance entre clusters de Big Data formados por HDDs e por SSDs, por máquinas físicas e por máquinas virtuais, entender o porquê da diferença e como foram quantificadas.

O restante do artigo está organizado da seguinte forma: a Seção 2 discute o framework Hadoop, bem como a estrutura e funcionamento dos componentes HDFS e MapReduce. A Seção 3 discorre sobre o método do mapeamento usado neste estudo. Por meio de gráficos e tabelas, a Seção 4 resume os resultados obtidos da análise dos artigos selecionados, em relação às respostas às questões de pesquisa. A Seção 5 aborda ameaças à validade dessa pesquisa. Finalmente, a Seção 6 apresenta as conclusões sobre este trabalho.

## **2. Fundamentação Teórica**

### **2.1. Framework Apache Hadoop**

A depender da forma que se pretende trabalhar com Big Data, existem várias tecnologias especializadas para isso. O Apache Hadoop, segundo [Akhtar et al. 2017], é um framework de código aberto que oferece armazenamento massivo de dados e processamento distribuído de grandes volumes de dados através do MapReduce, além do quê, por ser programado em Java, permite que desenvolvedores implantem programas personalizados para processar dados de maneira paralela, através de centenas ou milhares de nós servidores de clusters. Para [Valova 2023], o Apache Hadoop é um ecossistema que fornece

ferramentas muito poderosas para se trabalhar com Big Data em um ambiente gratuito, o qual, com a disponibilidade de hardware adequado para sua instalação, possui módulos essenciais para o seu funcionamento que foram e ainda são atualizados ao longo dos anos, a fim de atender às demandas.



Figura 1. Arquitetura do Hadoop [Sanches 2021].

## 2.2. Sistema de Arquivos Distribuídos do Hadoop (HDFS)

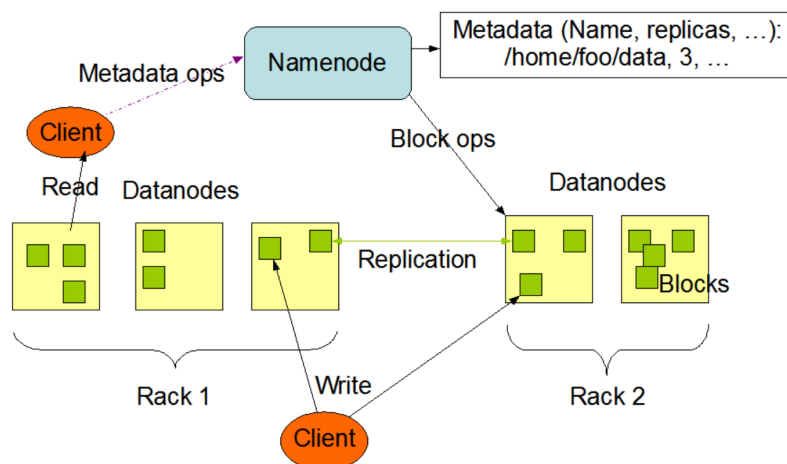
Segundo [Wu et al. 2018], o HDFS é o sistema de arquivos distribuídos subjacente que sustenta a execução do Hadoop. O HDFS pode ser implantado tanto em servidores dedicados quanto em máquinas comuns de baixo custo (commodity hardware) e é capaz de armazenar arquivos de dados muito grandes. Os usuários podem acessar o HDFS através de suas interfaces abstratas, como se estivessem operando em um sistema de arquivos local.

### 2.2.1. Replicação de dados

De acordo com a documentação oficial do [Apache Hadoop 2023], o HDFS tem uma arquitetura mestre-escravo. Adota o modelo “WORM” de “gravação única e leitura múltipla”, no qual os aplicativos gravam os arquivos de dados apenas uma vez (exceto para acréscimos e truncamentos), mas os leem uma ou mais vezes, o que simplifica a integração dos dados e aumenta a taxa de transferência. Conforme Figura 2, um cluster HDFS consiste em um único “NameNode” (um servidor mestre que gerencia o namespace do sistema de arquivos) que regula o acesso aos arquivos pelos clientes. Além disso, há vários “DataNodes”, geralmente um por cada computador (nó) do cluster, que gerenciam o armazenamento conectado aos nós físicos em que são executados.

O HDFS possui, portanto, dois softwares *daemons* que rodam em segundo plano: O “NameNode” e o “DataNode”.

- **NameNode:** É executado em um nó físico mestre. Mantém os metadados (nome do arquivo, número de blocos, número de réplicas, localização dos blocos, IDs dos blocos) na memória RAM (para acesso rápido) e também no disco (persistência). Quanto maior a memória RAM do nó mestre, melhor o desempenho. O NameNode, deve residir, portanto, em um nó físico (hardware) confiável. Grandes clusters podem exigir a participação de NameNodes secundários (Secondary NameNode).
- **Datanode:** É executado em um nó físico escravo. Também denominado de “worker”. É quem, de fato, executa tarefa de armazenamento distribuído dos dados, atendendo às solicitações de leitura e gravação dos clientes do sistema de arquivo. O Datanode pode residir em um hardware de commodities (nó físico não muito confiável). Portanto, cada nó físico em um cluster HDFS tem um DataNode.



**Figura 2. Arquitetura do HDFS [Apache Hadoop 2023].**

O mecanismo de replicação do HDFS permite que um bloco de dados tenha várias réplicas distribuídas em diferentes máquinas. Assim sendo, quando um grande arquivo de dados é gravado no HDFS, internamente, ele é dividido em um ou mais blocos e esses blocos são armazenados e referenciados por um conjunto de DataNodes. Por padrão, o fator de replicação do HDFS é 3.

Para garantir a tolerância a falhas, as atualizações de metadados são registradas no NameNode que gerencia a distribuição e localização desses blocos. Esse sistema de replicação proporciona alta confiabilidade, disponibilidade e tolerância a falhas, pois o seu principal objetivo é justamente armazenar grandes arquivos de dados (gigabytes e terabytes de dados) de forma distribuída e confiável.

### 2.3. Dispositivos de Armazenamento HDDs e SSDs

No cenário digital de larga escala, a gestão de grandes volumes de dados se torna uma decisão crítica, já que a forma na qual será trabalhada afeta fatores como escalabilidade, manutenção, eficiência e consequentemente custo. A escolha entre os tipos dispositivos de armazenamento, seja HDD ou SSD, impacta diretamente no desempenho dos sistemas. Tradicionalmente, os HDDs têm sido amplamente mais utilizados devido ao seu baixo custo por gigabyte. No entanto, a crescente demanda por processamentos mais rápidos e eficientes levantou questionamentos a respeito das vantagens dos SSDs, que utilizam memória flash e oferecem velocidades superiores, aliado ao fato da sua progressiva queda de preço.

Outro fator relevante no processo de escolha é o consumo de energia. Os SSDs, por não possuírem partes móveis, tendem a consumir menos energia e gerar menos calor que os HDDs, o que pode ser especialmente importante em data centers de grande escala. A eficiência energética se torna um fator decisivo na escolha, uma vez que pode reduzir significativamente os custos operacionais e contribuir para as metas de sustentabilidade das empresas. Em estudos comparativos, como o de [Moon et al. 2014], essas diferenças apresentaram impactos distintos no desempenho, com os HDDs sendo 15% menos eficazes em desempenho que os SSDs, mas que, em uma visão geral de custo e eficácia, o uso de HDD, é até 35% mais eficiente.

Além disso, os aspectos relacionados à durabilidade e manutenção também são

importantes. Segundo [Saxena and Kumar 2014], enquanto os HDDs, por terem partes mecânicas, podem requerer mais manutenção e ter menor durabilidade, os SSDs apresentam maior resistência a danos e menor taxa de falha ao longo do tempo.

## 2.4. Modelo de Programação MapReduce

De acordo com [Wu et al. 2018], MapReduce é um paradigma de computação paralela proposto pelo Google, que ganhou ampla adoção devido à simplicidade na programação. Esse é um dos vários pontos que estão a favor do MapReduce como principal meio de programação com Big Data no Apache Hadoop. A respeito de seu funcionamento, [Zhu and Miao 2014] divide o modelo de programação em duas etapas:

- Map (Mapear): Essa etapa é responsável por dividir os dados em partes menores e processá-los de forma paralela. Os resultados parciais são organizados e preparados para serem enviados à fase de Reduce. Ela é composta por três subfases: read, buffering e write.
- Reduce (Reduzir): “A fase de Reduce é mais complexa que a fase de Map, com um alto grau de paralelização e sobreposição nas operações de uma única tarefa de Reduce” [Zhu and Miao 2014]. Nessa Fase, os resultados do Map são agrupados e processados para gerar a saída do resultado final combinando e organizando os dados, em uma única resposta. Ela inclui três subetapas: shuffle/copy, sort e reduce.

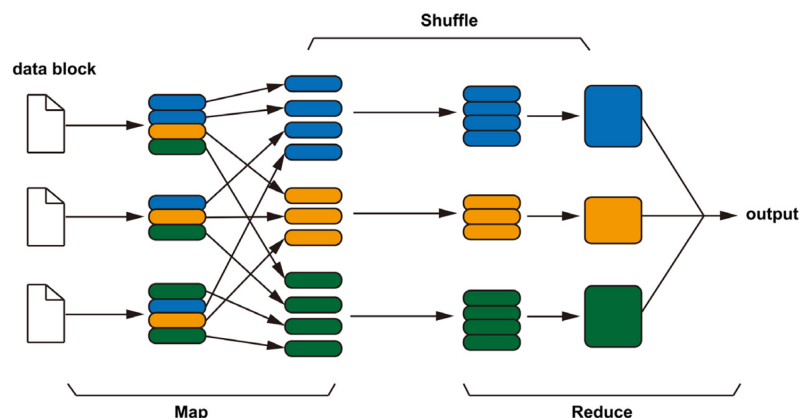


Figura 3. O processamento de uma tarefa do MapReduce [Wu et al. 2018].

## 3. Metodologia

O Mapeamento Sistemático da Literatura para [Kitchenham and Charters 2007] organiza as evidências de um determinado campo de pesquisa de forma ampla. Isso ajuda a identificar onde carecem informações e onde há espaço para novos estudos. O foco deste mapeamento foram clusters de big data com Apache Hadoop, envolvendo o uso de HDDs ou de SSDs em máquinas físicas ou virtualizadas, contemplando os últimos dez anos, com o objetivo de responder às seguintes questões de pesquisa (QPs):

- (QP1) Quais os tipos de clusters utilizados nos experimentos?
- (QP2) Quais os algoritmos ou aplicativos de testes usados nos experimentos?
- (QP3) Qual a linguagem dos algoritmos ou aplicativos de testes usados nos experimentos?
- (QP4) Quais os componentes da pilha de softwares do Hadoop presentes no ambiente do cluster?
- (QP5) Qual o Sistema operacional utilizado no ambiente do cluster?

### 3.1. Estratégia para Recuperação de Documentos nas Bases Científicas

#### 3.1.1. Análise PICO e Palavras-Chave

O protocolo PICO (Population, Intervention, Comparison, Outcome) foi usado para guiar a geração dos termos descritores e da string de busca aplicada nas fontes secundárias. A busca foi direcionada para clusters implementados com Hadoop, com foco em tecnologias de armazenamento HDDs e SSDs, quer seja em ambientes físicos ou virtualizados. Os seguintes termos descritores foram assim considerados:

- Population (População): cluster, clusters, clustering.
- Intervention (Intervenção): hadoop, hdfs, mapreduce, big data.
- Comparison (Comparação): HDD, HDDs, SSD, SSDs.
- Outcome (Resultado): wordcount, word, count, terasort, pi.

#### 3.1.2. String de Busca

Com base na análise PICO, a string de busca foi formulada com sentenças booleanas, a partir das palavras-chave advindas de cada um dos componentes do acrônimo PICO (P and I and C and O).

```
("cluster" OR "clusters" OR "clustering")  
AND  
("HDD" OR "HDDs" OR "SSD" OR "SSDs")  
AND  
("hadoop" OR "hdfs" OR "mapreduce" OR "big data")  
AND  
("wordcount" OR "word" OR "count" OR "terasort" OR "pi")
```

**Figura 4. String de busca aplicada nas bases.**

#### 3.1.3. Seleção dos Artigos

Critérios de inclusão de artigos:

1. Publicações que abordem a implementação de clusters Hadoop.
2. Trabalhos que envolvam experimentos com HDFS e MapReduce.
3. Artigos que realizem discussões teóricas sobre os componentes Hadoop.

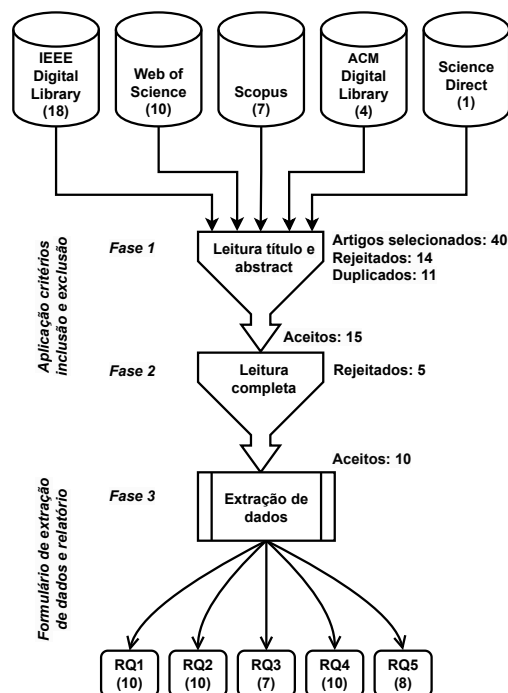
Critérios de exclusão de artigos:

1. Publicações que não abordam clusters de Big Data com Apache Hadoop.
2. Estudos que tratem apenas de computação em memória (ex.: Spark).
3. Publicações que não estejam em inglês ou português.
4. Textos inacessíveis ou versões mais antigas de estudos já incluídos.

## 4. Resultados e Discussão

### 4.1. Resultados Preliminares

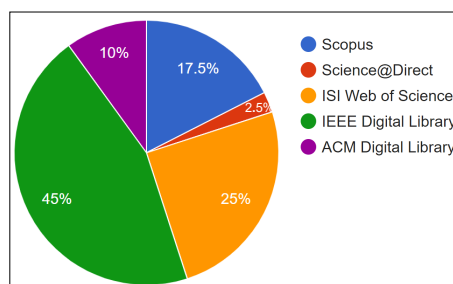
A pesquisa com a *string* de busca resultou em 40 artigos inicialmente recuperados de 5 bases de pesquisas: Scopus; IEEE; Web of Science; ACM e Science Direct (Figura 5). Na primeira triagem, com a leitura do título e *abstract*, foram retirados 11 artigos duplicados e 14 rejeitados por não atenderem aos critérios de inclusão. Dos 15 artigos aceitos na



**Figura 5. Processo de seleção de documentos e extração de dados.**

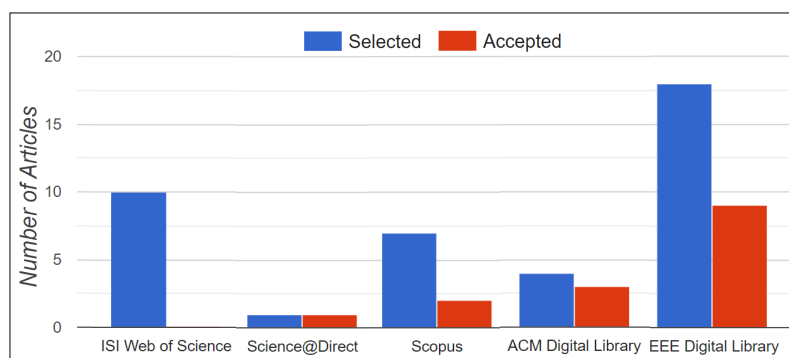
primeira fase de análise, após uma leitura completa dos seus conteúdos, apenas 10 foram finalmente aceitos. Sendo 2 da ACM, 6 da IEEE, 1 da Science Direct e 1 da Scopus.

A Figura 6 mostra a representação gráfica o percentual de artigos encontrados em cada uma das fontes de pesquisas utilizadas. Este gráfico apresenta uma visão clara da proporção de dados obtidos de diferentes bases científicas, permitindo uma análise detalhada da contribuição de cada fonte para o conjunto total de informações coletadas. Através desta ilustração, é possível avaliar a relevância e a cobertura proporcionadas por cada base de artigos na composição do estudo.



**Figura 6. Artigos recuperados por fonte.**

Na Figura 7, é possível observar o gráfico de barras que apresenta o aproveitamento dos documentos aceitos *versus* selecionados. Este gráfico detalha a eficácia de cada base na contribuição de informações relevantes para o estudo, evidenciando a proporção de material aproveitado em relação ao total recuperado.



**Figura 7. Artigos aceitos por fonte.**

## 4.2. Resultados das Questões de Pesquisa

### 4.2.1. QP1 - Quais os tipos de clusters utilizados nos experimentos?

Todos os artigos responderam diretamente a essa questão, sendo três trabalhando apenas com clusters em máquinas físicas, três apenas com clusters em máquinas virtuais e quatro com ambos os ambientes físicos e virtuais.

Ambiente de Cluster	Publicações
Máquinas Físicas	[Saxena and Kumar 2014, Hong et al. 2016, Lim and Park 2024]
Máquinas Virtuais	[Gugnani et al. 2016, Tang et al. 2017, Auradkar et al. 2020]
Ambas	[Moon et al. 2014, Issa 2015, Islam et al. 2016, Lee and Fox 2019]

**Tabela 1. Ambientes de clusters dos experimentos.**

### 4.2.2. QP2 - Quais os algoritmos ou aplicativos de testes usados nos experimentos?

Os 10 artigos responderam a essa questão, sendo que seis utilizaram somente o algoritmo TeraSort, dois utilizaram TeraSort e WordCount, um usou somente WordCount e um usou TeraSort juntamente com WordCount e DFSIO.

Algoritmo ou Aplicativo	Publicações
Terasort	[Saxena and Kumar 2014], [Hong et al. 2016], [Lim and Park 2024], [Gugnani et al. 2016], [Tang et al. 2017], [Auradkar et al. 2020], [Moon et al. 2014], [Islam et al. 2016], [Lee and Fox 2019]
Wordcount	[Issa 2015], [Tang et al. 2017], [Islam et al. 2016], [Lee and Fox 2019], [Auradkar et al. 2020], [Lim and Park 2024]
DFSIO	[Islam et al. 2016], [Lee and Fox 2019], [Auradkar et al. 2020]

**Tabela 2. Algoritmos ou aplicativos usados nos experimentos.**

### 4.2.3. QP3 - Qual a linguagem dos algoritmos ou aplicativos de testes usados nos experimentos?

Sete dos 10 artigos especificaram Java como linguagem utilizada, o que era esperado, visto que o Hadoop é desenvolvido em Java. Isso facilita o uso de Java para implementar



algoritmos e aplicativos que interajam diretamente com essa infraestrutura, porém não informaram versão. enquanto os outros 3 artigos não especificaram.

#### **4.2.4. QP4 - Quais os componentes da pilha de softwares do Hadoop presentes no ambiente do cluster?**

Todos os artigos utilizaram o HDFS e MapReduce, discorrendo que a otimização dos seus parâmetros afeta diretamente o desempenho dos clusters.

#### **4.2.5. QP5 - Qual o Sistema operacional utilizado no ambiente do cluster?**

Oito artigos informaram que o sistema operacional Linux foi usado no ambiente dos clusters dos experimentos. Dois artigos não informaram sobre o sistema operacional usado.

### **5. Ameaças à Validade**

Apesar do protocolo sistemático ser estabelecido de forma a evitar vieses, é impossível atestar a qualidade dos artigos que foram incluídos neste estudo, pois foram selecionados sem atribuição escores. Apenas 5 bases da literatura científica foram consultadas, fato este que limitou o escopo da busca, deixando fora documentos relevantes possivelmente existentes em outras bases. É plenamente possível que algum critério de exclusão tenha eliminado artigos importantes, o que também constitui uma ameaça à validade do presente artigo. O rigor do método, contudo, possibilita que outros pesquisadores possam replicar a presente pesquisa, ratificando resultados idênticos ou equivalentes.

### **6. Conclusões**

O levantamento de trabalhos da literatura científica publicados nos últimos 10 anos, sobre clusters de Big Data implementados com o Apache Hadoop, envolvendo o uso de HDDs ou de SSDs, em ambientes físicos ou virtualizados, constatou a existência de poucos documentos focados nesses cenários ambientais que procuraram estabelecer alguma relação comparativa de desempenho. Essa lacuna detectada contribui para que outros pesquisadores sejam instigados a direcionarem esforços de pesquisas relacionadas à avaliação comparativa de desempenho de clusters de Big Data, quanto ao custo, eficácia e eficiência.

### **Referências**

- Akhtar, N., Parwej, F., and Perwej, Y. (2017). A perusal of big data classification and hadoop technology. *International Transaction of Electrical and Computer Engineers System (ITECES), USA*, 4(1):26–38.
- Apache Hadoop (2023). Hdfs architecture. Apache Software Foundation (ASF).
- Auradkar, P., Prashanth, T., Aralihalli, S., Kumar, S. P., and Sitaram, D. (2020). Performance tuning analysis of spatial operations on spatial hadoop cluster with ssd. volume 167, page 2253 – 2266. All Open Access, Gold Open Access.
- Gugnani, S., Lu, X., and Panda, D. K. (2016). Performance characterization of hadoop workloads on sr-iov-enabled virtualized infiniband clusters. In *Proceedings of the 3rd IEEE/ACM International Conference on Big Data Computing, Applications and Technologies*, BDCAT '16, page 36–45, New York, NY, USA. Association for Computing Machinery.

- Gupta, M. K., Pandey, S. K., and Gupta, A. (2022). Hadoop- an open source framework for big data. In *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*, pages 708–711.
- Hong, J., Li, L., Han, C., Jin, B., Yang, Q., and Yang, Z. (2016). Optimizing hadoop framework for solid state drives. In *2016 IEEE International Congress on Big Data (BigData Congress)*, pages 9–17.
- Islam, N. S., Wasi-ur Rahman, M., Lu, X., and Panda, D. K. D. K. (2016). Efficient data access strategies for hadoop and spark on hpc cluster with heterogeneous storage. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 223–232.
- Issa, J. A. (2015). Performance evaluation and estimation model using regression method for hadoop wordcount. *IEEE Access*, 3:2784–2793.
- Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. 2.
- Lee, H. and Fox, G. (2019). Big data benchmarks of high-performance storage systems on commercial bare metal clouds. In *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*, pages 1–8.
- Lim, S. and Park, D. (2024). Improving hadoop mapreduce performance on heterogeneous single board computer clusters. *Future Generation Computer Systems*, 160:752–766.
- Moon, S., Lee, J., and Kee, Y. S. (2014). Introducing ssds to the hadoop mapreduce framework. In *2014 IEEE 7th International Conference on Cloud Computing*, pages 272–279.
- Reinsel, D., Gantz, J., and Rydning, J. (2017). Data age 2025: The evolution of data to life-critical. An IDC White Paper, Sponsored by Seagate.
- Sanches, R. (2021). Tudo programado blog: Introdução a arquitetura hadoop.
- Saxena, P. and Kumar, P. (2014). Performance evaluation of hdd and ssd on 10gige, ipoib rdma-ib with hadoop cluster performance benchmarking system. In *2014 5th International Conference - Confluence The Next Generation Information Technology Summit (Confluence)*, pages 30–35.
- Tang, Z., Wang, W., Huang, Y., Wu, H., Wei, J., and Huang, T. (2017). Application-centric ssd cache allocation for hadoop applications. In *Proceedings of the 9th Asia-Pacific Symposium on Internetware*, Internetware '17, New York, NY, USA. Association for Computing Machinery.
- Valova, I. (2023). Using big data and hadoop in the student learning process - enhancing the educational process through real experience. page 470 – 475. Cited by: 0.
- Wu, W., Lin, W., Hsu, C.-H., and He, L. (2018). Energy-efficient hadoop for big data analytics and computing: A systematic review and research insights. *Future Generation Computer Systems*, 86:1351–1367.
- Zhu, Q. and Miao, L. (2014). The realization of green storage in hadoop. In *Proceedings of 2014 International Conference on Cloud Computing and Internet of Things*, pages 91–95.