RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

Avaliação das Ferramentas do Big Data – Caso de Estudo: Spark Vs Flink

Big Data Tools Review – Case Study: Spark Vs Flink

David Francisco Cudijinguissa [1]
Instituto Superior Politécnico de Privado do Kilamba,
davicudj@gmail.com

Resumo

Este artigo teve como objectivo comparar o desempenho das ferramentas de Big Data, Spark vs Flink, considerando cinco atributos que tornam esses sistemas altamente complexos e exigentes em termos de processamento. Como resultado, as ferramentas utilizadas para trabalhar com esses dados tendem a ser significativamente mais robustas do que as convencionais, sendo muitas vezes também mais caras. Sistemas de código aberto (open source) oferecem acesso ao código-fonte, facilitando a compreensão dos sistemas e algoritmos pelos colaboradores e permitindo que os adaptem conforme as necessidades de seus projetos. A metodologia adotada neste estudo baseia-se em pesquisas descritivas para relacionar as variáveis, com uma abordagem exploratória e explicativa de caráter qualitativo. Um estudo de caso é apresentado, comparando as plataformas Spark e Flink e levando em consideração factores como escalabilidade, armazenamento de dados, complexidade e opções de implementação.
Palavras-chave: Análise de dados; Big data; Open source; Spark; Flink.

Abstract

This article aimed to compare the performance of Big Data tools, Spark and Flink, considering five attributes that make these systems highly complex and demanding in terms of processing. As a result, the tools used to work with this data tend to be significantly more robust than conventional ones, often being more expensive as well. Open-source systems provide access to the source code, making it easier for collaborators to understand the systems and algorithms, and allowing them to adapt them according to their project needs. The methodology adopted in this study is based on descriptive research to relate the variables, with an exploratory and explanatory approach of a qualitative nature. A case study is presented, comparing the Spark and Flink platforms, considering factors such as scalability, data storage, complexity, and implementation options.
Keywords: Data analysis; Big data; Open source; Spark; Flink.

1 INTRODUÇÃO

Quanto mais ampla e profunda a difusão da tecnologia da informação avançada em fábricas e escritórios, maior a necessidade de um trabalhador instruído (Castells, 1999, p. 306). Com a popularização da informática, internet das coisas e sensores, uma grande quantidade de dados tem expandido a necessidade de analisar grandes volumes de dados em lotes ou em tempo real. Esta competência se tornou uma base fundamental da competição que aumenta e sustenta novas ondas de crescimento da produtividade, inovação e superatividade de consumidores. Empresas de todos os sectores terão que lidar com as implicações dos grandes dados, e não apenas com alguns gerentes de dados orientados ou algum sector específico.

As técnicas do Big data reúnem métodos de várias áreas que, ao longo dos anos, se têm desenvolvido, como a estatística, inteligência artificial e machine learning (Torsten, 2018). Estas técnicas permitem a transformação de informação em conhecimento potencialmente útil.

Devido à complexidade no processamento desses dados, é necessário contar com uma infraestrutura e

[1] David Francisco Cudijinguissa, MSc. em Engenharia Informatica especialidade em gestão de Redes de Computadores e Sistemas de Comunicação.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.
Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

técnicas mais robustas do que as utilizadas pelos sistemas tradicionais. Além do desempenho, fatores como escalabilidade e resiliência precisam ser considerados, uma vez que é difícil garantir esses aspectos com os recursos convencionais. A aquisição dessas tecnologias pode representar um grande obstáculo na execução das atividades, pois muitas vezes requer investimentos elevados. A crise financeira atual pressiona as empresas a economizarem ao máximo seus recursos. Com orçamentos extremamente controlados, elas se veem obrigadas a continuar entregando resultados e a manter uma posição competitiva no mercado.

2 REVISÃO DA LITERATURA

2.1 Big Data

O termo "Big Data" foi introduzido pelo chefe da SGI2 , o cientista John R.Mashey (1998), podemos descrever como uma coleção massiva de dados estruturados e não estruturados, disponibilidade e processamento de grandes volumes de dados de streaming em tempo real, em 2001 o analista Doug Laney definiu esse termo como 3V: volume, velocidade e variedade. O volume está ligado à grande quantidade de dados, a velocidade se refere ao processamento de dados e a variedade se refere ao aumento da complexidade das análises. Porém, a literatura mostra que esse conceito está mais relacionado a 5Vs e são acrescentadas mais duas características. A veracidade que está directamente ligada ao quanto uma informação é verdadeira e ao valor que está relacionado com o valor obtido desses dados, ou seja, informação útil. O termo Big Data sempre foi um conceito relativo, pois seu tamanho depende de quem está usando os dados.

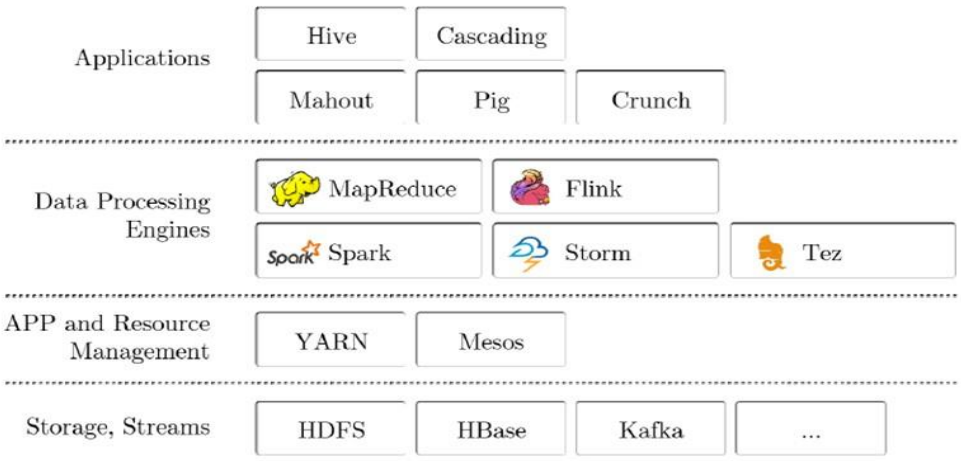

Figura 1 - Estrutura actual do Big Data
Fonte: Fayyad, Piatetsky-Shapiro, & Smyth, 1996.

Big Data é o conjunto de grandes informações em que gerada todos os dias – vastos zetabytes de dados que fluem de nossos computadores, dispositivos móveis e sensores de máquina.
Segundo Miranda (2023, p.4), o sistema tradicional utiliza os famosos SGBDs, ou sistemas gerenciais de base

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

de dados, que guardam informações de forma estruturada, no formato de tabelas, com linhas e colunas. Utilizam máquinas com grande capacidade de armazenamento e processamento. Quando há a necessidade de expandir a capacidade dessas máquinas, é necessário introduzir novos componentes de hardware, para que tenham mais memória e processamento.

2.2 As características do Big Data (cinco Vs)

Marques (2017) afirma que o volume, as grandes escalas das informações processadas podem ter ordens de magnitude maiores que os conjuntos de dados tradicionais, como os requisitos de trabalho excedem os recursos de um único computador, isso se torna um desafio de agrupar, alocar e coordenar recursos de grupos de computadores.

Velocidade, os dados estão a fluir frequentemente para o sistema de Big data a partir de várias fontes e cada vez mais se deseja que sejam processados em tempo real para obter percepções e actualizar o entendimento actual do sistema.

Variedade, os problemas de Big data geralmente são únicos devido à grande variedade de fontes processadas e sua qualidade relativa.

Vários indivíduos e organizações sugeriram expandir os três Vs originais, embora essas propostas tendam a descrever os desafios em vez de qualidades de Big data. Algumas adições comuns são:

Veracidade: A variedade de fontes e a complexidade do processamento podem levar a desafios na avaliação da qualidade dos dados.

Valor: O desafio final do Big Data é entregar valor. Às vezes, os sistemas e processos implementados são complexos o suficiente para que a utilização dos dados e a extração do valor real possam se tornar difíceis.

2.3 Classificação do Processamento Analítico

Em relação ao componente Processamento Analítico, existem diversas arquiteturas no contexto de Big Data.

Classificamos as abordagens de acordo com seu suporte de armazenamento de dados: (i) aquelas baseadas em armazenamento de dados em disco, que dividimos em dois grupos de acordo com o modelo de dados, bancos de dados NoSQL. (classe A na Fig. 2) e relacionais (classe B na Fig. 2); e aquelas que suportam bancos de dados em memória (classe C na Fig. 2), nas quais os dados são mantidos em RAM reduzindo as operações de I/O (Lee *et al*., 2012).

Detalhamos cada classe da seguinte forma:

A- Arquitetura baseada em NoSQL: É baseada em bancos de dados NoSQL como modelo de armazenamento, contando ou não com DFS (Distributed File System). Ele também se baseia em modelos de processamento paralelo, como MapReduce, nos quais a carga de trabalho de processamento é espalhada por muitas CPUs em nós de computação comuns. Os dados são particionados entre os nós de computação em tempo de execução e a estrutura sublinhada lida com a comunicação entre máquinas e falhas de máquina. urais.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.
Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

A personificação mais famosa de um cluster MapReduce é o Hadoop. Ele foi projetado para rodar em muitas máquinas que não compartilham memória ou discos (o modelo de nada compartilhado). Este tipo de arquitetura é projectado por Ain Fig.2.

B- Arquitetura de banco de dados relacional paralelo: Baseia-se, como bancos de dados clássicos, em tabelas relacionais armazenadas em disco. Ele implementa recursos como indexação, compactação, visualizações materializadas, compartilhamento de entrada ou saída e armazenamento em cache de resultados.

Entre as arquiteturas de pesquisa estão: nada compartilhado (múltiplos nós autônomos, cada um possuindo seus próprios dispositivos de armazenamento persistente e executando cópias separadas do SGBD), memória compartilhada ou qualquer coisa compartilhada (um espaço de endereço de memória global é compartilhado e um SGBD está presente) e disco compartilhado (baseado em vários nós de processamento fracamente acoplados semelhantes ao nada compartilhado, mas um subsistema de disco global é acessível ao SGBD de qualquer nó de processamento).

Nesta arquitetura, a solução analítica é baseada na conjunção de bancos de dados paralelos (sem compartilhamento) com um modelo de programação paralela. Podemos ver este tipo de arquitetura projetada por Bin Fig. 2.

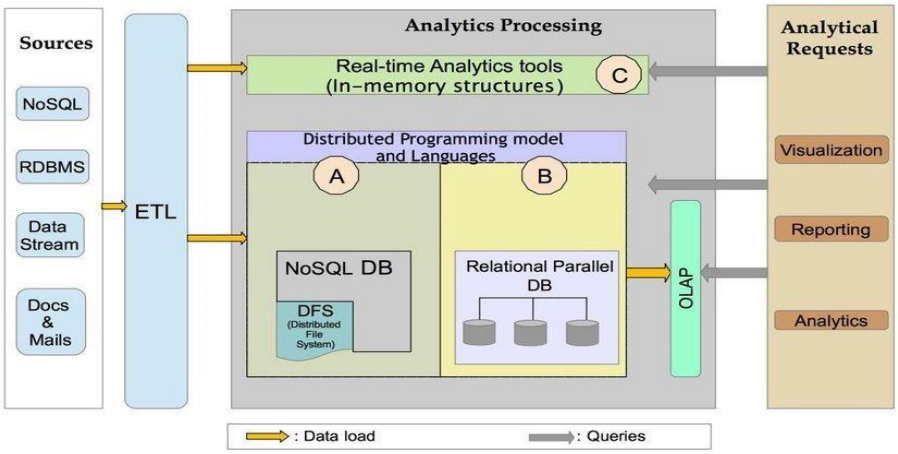

Figura 2 - Arquitetura generia para uma abordagem big data
Fonte: Research Gate. (2021).

2.4 Tipos de processamentos

O processamento de Big Data ou big data analytics trata-se de um conjunto de práticas que analisam um volume imenso de dados estruturados e não estruturados. Cujo objectivo principal, é a obtenção de respostas para enfrentar desafios e identificar oportunidades para o seu negócio.

"A quantidade de dispositivos somada aos diversos formatos de arquivos, e a necessidade da extrair de valor dos mesmos, mostrou a limitação dos modelos relacionais, que serviam bem para o tratamento de dados estruturados, mas não possibilitavam o tratamento de dados semiestruturados ou não estruturados.

Ainda no modelo NOSQL, os dados oriundos de diversos dispositivos desde aparelhos móbiles até servidores, são replicados em clusters onde são processados através de ferramentas Analytics, e posteriormente

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

visualizados através de gráficos, dashboards, entre outras ferramentas de análise, tal qual no modelo ETL, também usado nos modelos relacionais.

Informações estruturadas são aquelas informações dispostas de forma rígida em um ambiente já pensado para análise de dados e análises quantitativas, como planilhas de Excel, sistemas legados ou arquivos de texto. Se você cria uma célula no Excel que aceita apenas números, não adianta colocar texto, entendeu? Tipos de dados estruturados:

- Planilhas eletrônicas como as do
- As próprias bases de
- Arquivos
- Arquivos
- Arquivos JSON

Informações não estruturadas. Já as informações não estruturadas também podem ser utilizadas em sua base de dados, no entanto, essas possuem uma análise um pouco mais complexa, uma vez que não possuem formato para padronização da leitura, como: Páginas da internet; Vídeos, Áudios, Gravações telefónicas, Documentos do Word ou Google Docs.

2.5 Processamento em Batch (Lote)

A tradução dada pelo Google da palavra “batch” é lote. O que reflete bem a definição de processamento em batch, que consiste em processar uma grande quantidade de dados de uma só vez. Microsoft (2025).

Processamento em Batch é um método de processamento de dados semelhantes em lotes e as executa em sequência. Ele é bastante oportuno para a execução de tarefas que são repetitivas e demoradas ou que requerem recursos computacionais significativos.

Processamento em Batch é comumente utilizado quando diante de grandes quantidades de dados e em situações que não exigem resultados de análise em tempo real. Alguns exemplos de utilização são situações do financeiro como folha de pagamento ou cobranças.

De acordo com Wampler, (2018, p. 12), os três componentes essenciais de uma arquitetura em batch (Hadoop) são, o HDFS para armazenamento, o MapReduce e/ou o Spark para processos de computação e o YARN para controlo, enquanto na arquitetura de streaming o Kafka é utilizado para armazenamento, o Spark, Flink, Akka Streams e Kafka Streams são usados para computação, e o controlo pode ser feito pelo Kerberos, Mesos ou pelo YARN (com algumas limitações).

Pois esse período trouxe novos desafios para o cerne da discussão e a comunicação teve um papel central para o desenrolar de todo o processo. As organizações viram-se forçados alterar processos e formas de trabalhar, praticamente de um dia para outro, sendo que o trabalho a distância se tornou inevitavel para a continuidades das actividades.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

2.6 Processamento em Streaming (fluxo)

Os mecanismos de streaming com latências muito baixas como o Kafka permitem funcionalidades similares às do MapReduce realizado em ambientes Hadoop, com a diferença de que podem processar terabytes em microssegundos em vez das altas latências dos sistemas de batch (Narkhede, Shapira, & Palino, 2017).

Uma arquitetura de Big Data deve configurar as duas soluções de processamento, uma vez que nem todos os processos exigem uma resposta imediata. O processamento em batch tem grande eficiência, é altamente escalável, de baixo custo e processa dados em repouso, enquanto o streaming permite dar respostas imediatas à medida que os eventos acontecem continuamente nas organizações, melhorando a capacidade de resposta a situações como deteção de fraude, ajustamento de preços em tempo real, alertas de situações clínicas entre outras (Narkhede, Shapira, & Palino, 2017).

2.7 Software open source

Nesta secção abordamos o conceito de software open source e as suas origens, e são ainda referidas algumas das principais vantagens e desvantagens do uso deste tipo de modelo relativamente a outros.

Os softwares open source constituem hoje uma fonte, ainda pouco explorada, de ferramentas úteis para o desenvolvimento dessas novas tarefas. Actualmente, existe um conjunto de softwares open source que permite desenvolver o processo de ensino e aprendizagem não só de uma forma mais atraente, mas, sobretudo, mais eficaz.

O software open source, software de código aberto ou também designado software livre, surge como alternativa ao software proprietário ou comercial. É distribuído mediante um conjunto de licenças entre as quais se destacam a GPL (General Public Licence) e a BSD (Berkeley Software Distribution).
Pushkarev et al. (2010) avaliam sete ferramentas open-source ou com períodos gratuitos de teste utilizando critérios como conectividade, gerenciamento, interface e funcionalidades. Gao et al. (2016) analisam oito ferramentas comerciais em relação às suas funcionalidades.

3 - METODOLOGIA DA PESQUISA

Nesta secção é descrito os aspectos metodológicos baseados na pesquisa quanto à classificação geral, a abordagem da pesquisa, procedimentos técnicos, técnica de recolha de dados.

“Método é o caminho para se realizar alguma coisa e quando se tem o caminho, tornasse mais fácil realizar viagens sabendo onde se está e aonde se quer chegar e como faze-lo”(Pereira Et Al., 2018, p. 67).

A metodologia utilizada na elaboração deste estudo baseia-se em pesquisas descritivas com o objetivo de relacionar as variáveis envolvidas, além de uma abordagem exploratória e explicativa. Em termos de metodologia, adotou-se uma abordagem qualitativa. Esse método foi escolhido para facilitar a compreensão das ferramentas de Big Data, especificamente nas plataformas Spark vs Flink.

Foram empregados diversos métodos, procedimentos teóricos, técnicas e tipos de pesquisa para

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

alcançar o objetivo geral, destacando-se os seguintes:

1. Método Teórico:
    - Pesquisa Bibliográfica: Utilizada para a análise de elementos bibliográficos já publicados, como livros, revistas, publicações e artigos científicos. Esse método teve como finalidade atualizar o conhecimento sobre as temáticas a serem solucionadas, garantindo a identificação de informações existentes e inexistentes na validação final da proposta.
2. Métodos Empíricos:

    a) Observação: Aplicada no processo de verificação das ferramentas de Big Data em open source. Além disso, a técnica de entrevista foi utilizada na fase de levantamento de requisitos para avaliar o nível de conhecimento sobre essas ferramentas.

    b) Experimentação: Empregada na validação das funcionalidades de algumas ferramentas, como Spark e Flink.

    c) Método Hipotético-Dedutivo: Utilizado na formulação de hipóteses, partindo da premissa de que a avaliação das ferramentas de processamento de Big Data em open source possibilita uma melhor escolha quando confrontada com a necessidade específica de processamento de Big Data em um determinado caso.

Esta pesquisa tem carácter indutivo, conforme apontam Marconi e Lakatos (2007), diante da observação sistemática e da classificação dos fenómenos seleccionados. Ela está fundamentada nas circunstâncias e frequências em que as publicações sobre esse tema ocorreram e na medição de suas diferentes intensidades.

4 - APRESENTAÇÃO DOS RESULTADOS

4.1 Descrição

Os testes com a aplicação Word Count foram realizados repetidamente até alcançar um nível de confiança de 95%, utilizando uma distribuição t-Student, com no mínimo 90 execuções. O erro máximo permitido foi de 5%. Da mesma forma, os testes com a aplicação word count foram repetidos até atingir 90% de confiança, também com uma distribuição t-Student e pelo menos 90 execuções, permitindo um erro máximo de 10%.

O ambiente de virtualização foi configurado no Hyper-V, contendo duas máquinas virtuais, ambas executando o sistema operacional Linux - Ubuntu.

- Primeira máquina virtual: configurada com o Apache Spark para processamento distribuído de grandes volumes de dados.
- Segunda máquina virtual: configurada com o Apache Flink, focado em processamento de fluxo de dados em tempo real.

Essa infraestrutura permite a execução e comparação de workloads em diferentes frameworks de Big Data, avaliando o desempenho e a eficiência de cada um conforme os cenários de teste.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

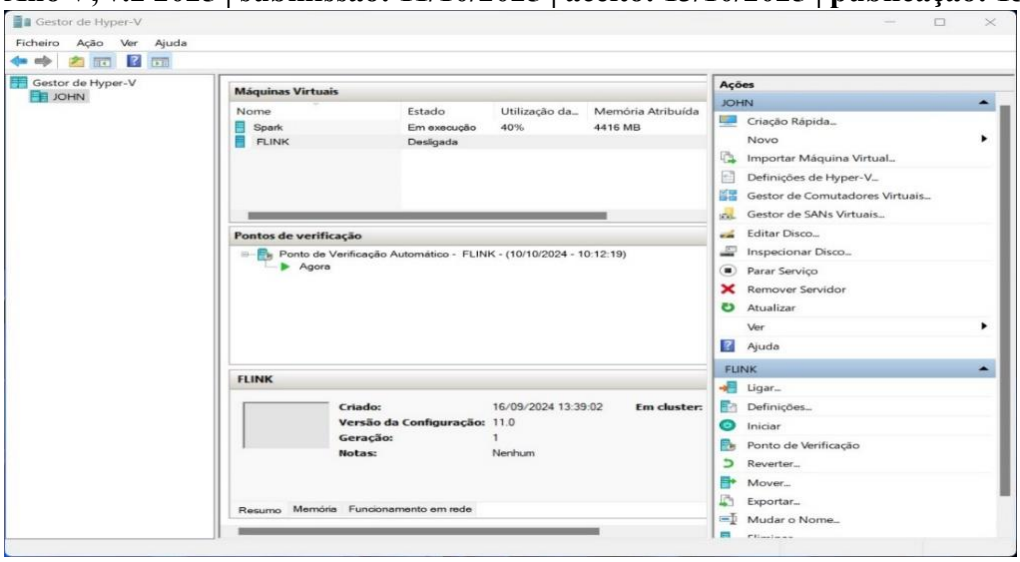

Figura 3 - Ambiente Hyper-V com Spark vs Flink

Fonte: Autor.

A Figura 4 apresenta a interface gráfica do Apache Spark, exibindo informações sobre Jobs e Tasks, juntamente com seus respectivos tempos de duração. Essa visualização permite monitorar a execução das tarefas, avaliar o desempenho do processamento e identificar possíveis gargalos na distribuição da carga de trabalho.

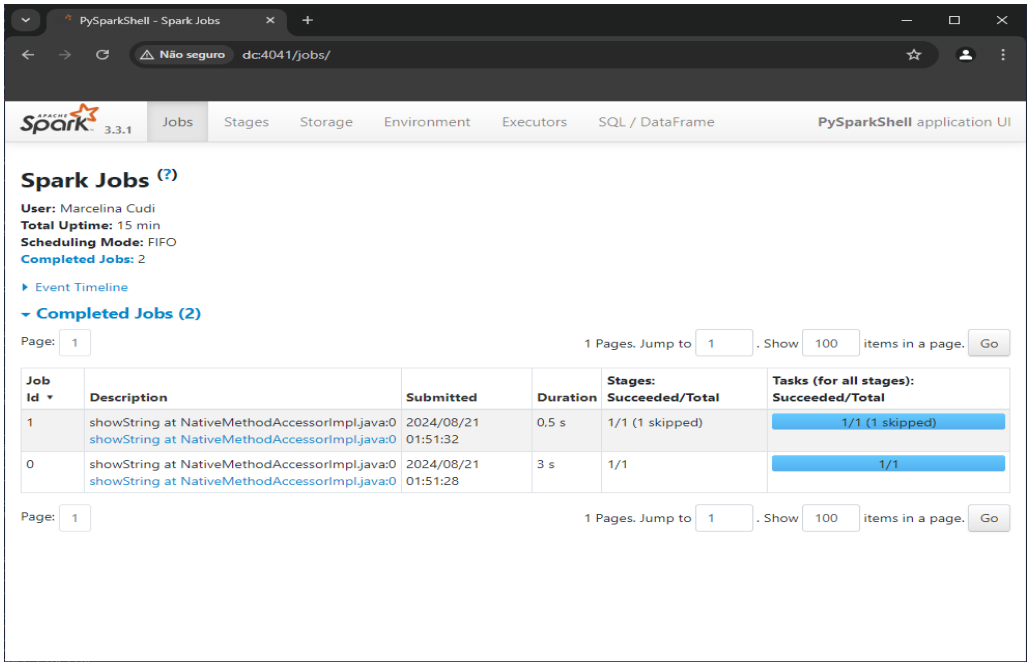

Figura 4 - Ambiente GUI Pysparkshel – Spark jobs

Fonte: Autor.

A Figura 5 apresenta a interface gráfica do Apache Flink, exibindo as métricas do Job Manager. Essas métricas fornecem informações detalhadas sobre o desempenho e o estado das execuções, incluindo estatísticas de utilização de recursos, tempo de execução e status das tarefas em andamento.

RCMOS - Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

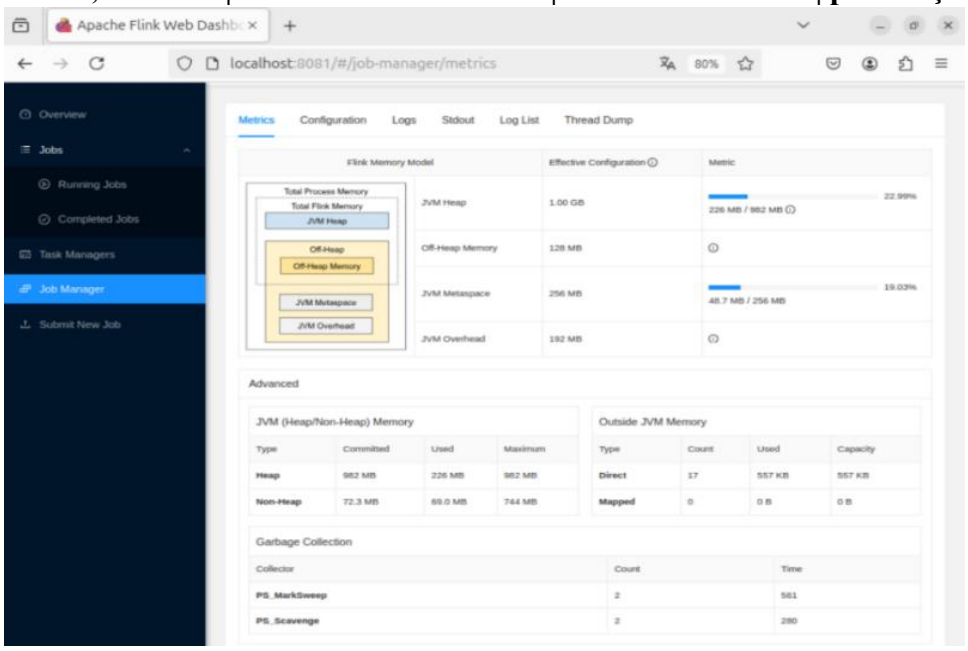

Figura 5 - Interface gráfica do Apache Flink Job Manager.

Fonte: Autor.

4.2 Descrição dos resultados

$$S = \frac{T1}{Tn}$$

A aceleração (speedup) de um programa paralelo é definida como:

Onde:

- T1 é o tempo de execução do programa em um único processador.
- Tn é o tempo de execução do programa em nn processadores.

A estimativa para o speedup é dada por:

$$Tn = Ts + \frac{Tp}{n}$$

Sendo:

- Ts o tempo de CPU para a parte serial do programa.
- Tp o tempo de CPU para a parte executada em paralelo.

Portanto, o speedup pode ser expresso como:

$$S = \frac{T1}{Tn} + \frac{Ts + Tp}{Ts + \frac{Tp}{n}}$$

Para $n \to \infty$, a equação se aproxima de:

$$Sma'x = \frac{Ts + Tp}{Ts}$$

4.3 Resultados experimentais:

- Tempo total (real): 4,472 s (inclui outros processos no sistema).

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

- Tempo de usuário (user): 7,243 s (tempo gasto pelo processador executando o programa).
- Tempo de sistema (sys): 0,360s (tempo gasto pelo kernel do sistema operacional para o programa).

Esses valores indicam que o tempo total de execução inclui sobrecargas do sistema operacional e outros processos concorrentes. Para uma análise mais precisa do speedup, deve-se considerar o tempo de usuário como referência para a execução do programa.

5 - CONSIDERAÇÕES FINAIS

5.1 Conclusões

Embora ainda sejam plataformas relativamente novas, o Flink apresentou resultados impressionantes quando comparado ao Apache Spark. Por exemplo, em uma aplicação que lida com grandes volumes de entrada e saída, como o WordCount, o ganho de velocidade (speedup) alcançou até 1:86. Em uma aplicação como a Estimativa de PI, o tempo de execução do Flink permaneceu quase constante mesmo com o aumento do número de amostras, ao contrário do observado com o Spark, onde o ganho máximo de velocidade foi de 7:30.

No entanto, a instalação do Flink enfrentou grandes dificuldades devido à falta de memória, o que impediu a coleta do tempo de execução. Com base na pesquisa bibliográfica, pode-se inferir que o Flink é otimizado para o processamento de dados em fluxo contínuo, fazendo uso intensivo da memória. Isso resulta em uma redução significativa no tempo de execução quando se trata de grandes fluxos de dados.

5.2 Recomendações para Investigação Futura

Sugerimos aos futuros investigadores que continuem este estudo comparativo de ferramentas de *Big Data open source* preparadas para projetos em tempo real, visando oferecer uma alternativa às ferramentas comerciais.

Além disso, seria enriquecedor acompanhar o desenvolvimento das ferramentas open source Spark e Flink, comparando seus resultados de performance com ferramentas comerciais de mineração de dados. Dessa forma, será possível estabelecer uma comparação direta entre as duas realidades e compreender os custos envolvidos na obtenção desses resultados.

REFERÊNCIAS

1. Castells, M. A. (1999), *Sociedade em rede*. São Paulo: Paz e Terra. v. 1.
2. Fayyad, U. M.; Piatetsky-Shapiro, Gregory; Uthurusamy, R. (1996). *Advances in Knowledge Discovery and Data Mining*. USA: MIT Press.

Ano V, v.2 2025 | submissão: 11/10/2025 | aceito: 13/10/2025 | publicação: 15/10/2025

3. Gao, J. Z., Xie, C., and Tao, C. (2016). *Big data validation and quality assurance issues, challenges, and needs*. In SOSE, pages 433-441. IEEE Computer Society.
4. John, R. M. (1998), Big Data, 1.3 X/year CAGR: historical trendline 1.6 X/year since 1990 2.0 X/year leap 1998/1999.
5. Lee, Y. W. et al. (2012). *A methodology for information quality assessment. Information & Management:* Aimq: 40(2):133-
6. Marconi, M. de A., & Lakatos, E. M. (2017). Metodologia Científica: (7ª ed. Atualização João Bosco Medeiros). Atlas.
7. Marques A. V. (2017), *importância dos big data no sector*.
8. Microsoft. (2025). Big Data Architectures. Microsoft Learn. Disponível em: https://learn.microsoft.com/en-us/azure/architecture/databases/guide/big-data-architectures Miranda, J. V. (2023). *Big data*: https://www.alura.com.br/artigos/big-data.
9. Narkhede, N., Shapira, G., & Palino, T. (2017). *Kafka: The Definitive Guide. Gravenstein Highway North, Sebastopol, CA*: O'Reilly Media, Inc.
10. Pereira, A. S. et al. (2018). *Metodologia da pesquisa científica*: Santa Maria: UAB / NTE / UFSM, Doi: http://repositorio.ufsm.br/handle/1/15824.
11. Pushkarev, V. et al. (2010). An overview of open source data quality tools. In IKE, pages 370-376. CSREA Press.
12. ResearchGate. (2021). *Generic-Architecture-for-a-Big-Data-Analytical*. https://www.researchgate.net/publication/318870641/figure/fig1/.
13. Torsten H. (2018). *Machine Learning & Statistical Learning.* Disponível em: https://www.pt.m.wikipedia.org/ wiki/Big_data.
14. Wampler, D. (2018). *Fast Data Architectures for Streaming Applications* - Second Edition. Gravenstein Highway North, Sebastopol: O'Reilly Media, Inc.
15. Witten, I. H., Frank, E., & Hall, M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques, 3 edn, Morgan Kaufmann Publishers.