

**“Do conhecimento acadêmico à transformação sustentável: inovação com validação científica”****COMPARAÇÃO DE BANCO DE DADOS NOSQL PARA SÉRIES ESPAÇO-TEMPORAIS NA CAFEICULTURA**Luís Eduardo Damasceno<sup>1</sup> (IC), Melise Maria Veiga de Paula (PQ)<sup>1</sup><sup>1</sup>Universidade Federal de Itajubá**Palavras-chave:** Benchmarking. Brazil Data Cube. InfluxDB. MongoDB**Introdução**

O café está entre os produtos mais consumidos globalmente, e o Brasil lidera sua produção, com 38% do total. Apesar dos avanços da agricultura digital, a cafeicultura enfrenta pressões para aumentar a produtividade em meio a eventos climáticos extremos (QUEIROZ et al., 2022). Nesse contexto, a adoção de tecnologias ganha relevância, mas muitos produtores ainda encontram dificuldades para acessar informações agronômicas de qualidade, devido aos desafios na coleta, no armazenamento, no processamento e, principalmente, na interpretação dos dados (BASSOI et al., 2019)

Este estudo integra um projeto que busca reduzir essa lacuna tornando os dados mais acessíveis. Entre as tecnologias aplicadas, destacam-se os sensores remotos orbitais, capazes de fornecer informações frequentes e precisas para avaliar a saúde das plantas, mapear lavouras, identificar áreas mais produtivas e monitorar fatores agronômicos como perda foliar, estresse hídrico e níveis de nutrientes (FORMAGGIO et al., 2017)

Neste estudo, foi desenvolvida uma base de dados com informações espectrais do satélite Sentinel-2A/2B para o Sul de Minas Gerais, com foco em Três Pontas. Para isso, compararam-se dois SGBDs capazes de lidar com dados espaço-temporais: o MongoDB, que oferece suporte nativo a dados espaciais e séries temporais (HACHIMI et al., 2023), e o InfluxDB, projetado para séries temporais e com funcionalidades para consultas geográficas (CHOI et al., 2022). O objetivo foi avaliar o desempenho de ambos em operações de inserção e consulta de dados da cafeicultura em cenários simulados.

Os dados utilizados neste estudo foram obtidos por meio do Brazil Data Cube (BDC), projeto do Instituto Nacional de Pesquisas Espaciais que oferece séries históricas de sensoriamento remoto para todo o território brasileiro, abrangendo bandas espectrais e

índices de vegetação. As informações são estruturadas em matrizes multidimensionais, facilitando o acesso, a análise e a visualização de grandes volumes de dados geoespaciais (BATINA, 2023).

Este estudo se justifica pela importância de estruturar uma base de dados adequada e avaliar alternativas de armazenamento para dados de sensoriamento remoto voltados à cafeicultura. A pesquisa busca compreender os desafios técnicos relacionados à persistência e consulta de séries espaço-temporais, fornecendo subsídios para futuras aplicações e estudos na área.

**Metodologia**

A etapa inicial da pesquisa envolveu a escolha dos SGBDs, levando em conta a característica espaço-temporal dos dados e a necessidade de garantir escalabilidade e flexibilidade (MEHMOOD et al., 2017). Optou-se por soluções NoSQL com suporte a séries temporais e dados geoespaciais, entre as quais se destacam o MongoDB, que oferece coleções do tipo time series e consultas geoespaciais via GeoJSON (HACHIMI et al., 2023), e o InfluxDB, especializado em séries temporais, com alto desempenho na ingestão de dados e suporte espacial através do pacote geo (INFLUXDATA, 2025). Outras opções de SGBDs especializados em séries temporais, como o OpenTSDB e o CnosDB, também foram consideradas, mas o InfluxDB destacou-se por sua liderança no ranking da DB-Engines (DB-ENGINES, 2025).

Na segunda etapa, dedicada à aquisição e tratamento dos dados, utilizaram-se séries temporais obtidas do Brazil Data Cube (BDC) por meio da API STAC, implementada em R (v4.4.1). Foram extraídos dados da coleção S2-16D-2, formada por imagens dos satélites Sentinel-2A/2B, com resolução espacial de 10 m e temporal de 16 dias, no intervalo de 01/01/2017 a 16/03/2025. A base foi composta pelas bandas espectrais Red, Green e NIR, além do índice de vegetação NDVI.

**“Do conhecimento acadêmico à transformação sustentável: inovação com validação científica”**

O município de Três Pontas (MG) apresentou 6.398.277 píxeis disponíveis. Para adequar os testes à infraestrutura utilizada, aplicou-se uma amostragem sistemática, selecionando 1 a cada 500 píxeis, o que resultou em 12.784 amostras. Para cada píxel foram geradas quatro séries temporais (três bandas e o NDVI), com registros em intervalos regulares de 16 dias, totalizando 2.403.392 entradas no banco de dados. Mesmo após a redução, a amostra manteve representatividade suficiente para avaliar o desempenho dos SGBDs no armazenamento e consulta.

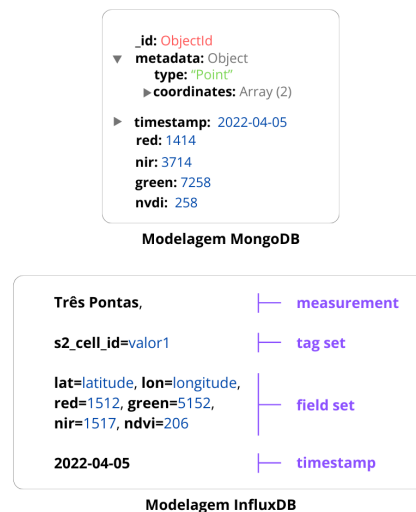
Na terceira etapa foi definida a modelagem dos dados, cada SGBD foi estruturado de acordo com suas particularidades. No MongoDB, utilizou-se coleções do tipo Time Series, em que cada documento representa um píxel em uma data específica, incluindo coordenadas geográficas, valores das bandas espectrais e o NDVI, representando pela Figura 1 - Modelagem MongoDB.

No InfluxDB, os dados foram estruturados em measurements, utilizando tags (pares chave-valor) para indexação e filtragem. O suporte espacial, necessário para o uso do pacote geo, foi garantido pelo identificador S2CellId, definido obrigatoriamente como tag e gerado pela biblioteca S2 em R a partir das coordenadas em WGS84. Já a latitude, a longitude e os valores espectrais foram armazenados como fields, requisito fundamental para a execução correta das consultas espaciais e temporais (ZEHR, 2017).

Na etapa de comparação, ambos os bancos foram implantados em containers Docker, garantindo ambientes isolados e padronizados para execução adequada dos testes (TRIPATHI et al., 2023). As métricas analisadas foram latência média e throughput, conforme descrito por KIM et al. (2023) e TRIPATHI et al. (2023). Essas métricas foram avaliadas com variação no número de usuários, visando identificar o impacto dessa variação no desempenho de cada banco de dados.

Os experimentos foram implementados em Python 3.13.3 e executados em um notebook VAIO (AMD Ryzen 5 5500U, 16GB RAM, 237GB SSD). Os códigos utilizados encontram-se disponíveis no repositório do projeto [GitHub](#).

Figura 1 - Modelagem dos dados espaço-temporais



O fluxo de execução dos testes foi dividido em três etapas principais. Na primeira, realizou-se uma análise preliminar dos containers Docker utilizados por cada SGBD, avaliando tamanho da imagem e consumo inicial de memória e CPU, definido como Teste 1.

Na segunda etapa, foram realizados testes de carga simulando duas inserções de dados: a primeira representando a passagem inicial do satélite sobre a região e a segunda, uma nova passagem com dados atualizados, caracterizando o comportamento contínuo das séries temporais. Para cada inserção, foi medida a latência de escrita com um único usuário, definido como Teste 2.

A terceira etapa consistiu nos testes de consulta, considerando informações típicas de monitoramento da cafeicultura. A primeira consulta recuperou toda a série temporal de bandas espectrais e NDVI para um píxel específico. No MongoDB, o acesso foi feito via coordenadas geográficas (`metadata.coordinates`), enquanto no InfluxDB foram testadas duas abordagens: filtragem direta por latitude e longitude (Abordagem 1) ou conversão para o identificador S2CellId, armazenado como tag (Abordagem 2), utilizando o pacote geo, definido como Teste 3A.

A segunda consulta buscou os valores das bandas em uma data específica para píxeis próximos a um ponto central em um raio de 1,5 km, chamado de Teste 3B.

Todos os testes de consulta foram repetidos 30 vezes com diferentes pontos de referência, variando o número de usuários simultâneos. Latência e throughput médios foram calculados, garantindo uma comparação

**“Do conhecimento acadêmico à transformação sustentável: inovação com validação científica”**

consistente do desempenho dos SGBDs em diferentes cenários de acesso e carga.

**Resultados e discussão**

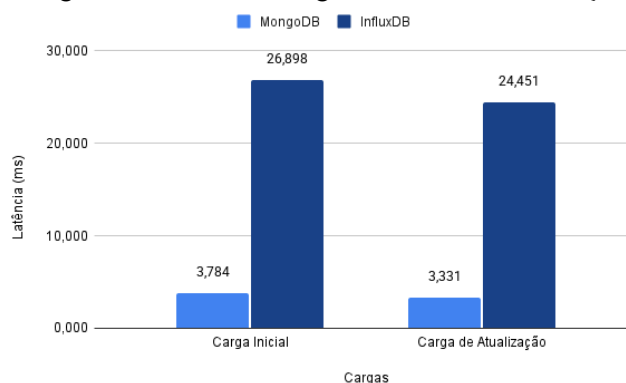
No Teste 1, que considerou os containers no estado inicial, o InfluxDB mostrou-se mais leve que o MongoDB, com imagem Docker cerca de 3 vezes menor, uso de memória 3,1 vezes inferior e consumo de CPU aproximadamente 11 vezes menor. Esses resultados evidenciam sua maior eficiência em termos de recursos, embora restritos ao estado inicial, sem operações de leitura ou escrita.

No Teste 2, representado pela Figura 2, foram avaliadas as cargas inicial e de atualização. O MongoDB apresentou desempenho cerca de sete vezes superior ao InfluxDB. Esse resultado sugere maior eficiência do MongoDB no processo de escrita, tanto na carga inicial quanto na atualização de dados.

O InfluxDB apresentou maior custo de escrita devido à sua arquitetura interna e o modelo de dados envolvido. O uso do identificador S2CellId como tag e a presença de múltiplos fields por registro aumentaram a cardinalidade, elevando o número de séries geradas e, conseqüentemente, a sobrecarga de indexação e armazenamento (INFLUXDATA, 2025). Em contraste, o MongoDB, ao organizar os dados em buckets de séries temporais agrupados por coordenadas (MONGODB, 2025), reduziu a complexidade do processo e obteve desempenho mais estável e eficiente.

No Teste 3A, que avaliou consultas por pixel, observou-se diferença significativa entre as abordagens de consulta no InfluxDB. A filtragem direta por latitude e longitude (Abordagem 1) apresentou o pior desempenho, chegando a sobrecarregar o banco em execuções com maior número de threads. Esse resultado está associado à necessidade de reestruturar os dados em memória por meio da função pivot() (INFLUXDATA, 2025), o que torna a consulta mais custosa. Já o MongoDB, ao utilizar a coordenada como metaField em coleções de séries temporais, obteve desempenho estável e superior à Abordagem 1, pois organiza os registros de um mesmo ponto em buckets temporais indexados (MONGODB, 2025), facilitando a recuperação.

Figura 2 – Teste de Carga Inicial e de Atualização



Por outro lado, a conversão das coordenadas para o identificador espacial S2CellId, armazenado como tag (Abordagem 2), resultou no melhor desempenho entre os modelos testados. Nessa configuração, o InfluxDB pôde explorar a indexação nativa de tags no mecanismo Time Series Index (TSI), recuperando de forma rápida as séries correspondentes a um pixel específico (INFLUXDATA, 2025). Em cenários com uma única thread, essa abordagem chegou a ser mais de 20 vezes mais rápida que o MongoDB e milhares de vezes mais eficiente que a filtragem direta (Abordagem 1), evidenciando a importância de modelagens baseadas em tags para consultas geoespaciais de alta cardinalidade.

No Teste 3B, que avaliou consultas espaciais por região, o MongoDB obteve melhor desempenho que o InfluxDB. Isso ocorreu porque o \$geoNear utiliza o índice geoespacial 2dsphere para calcular distâncias de forma direta e eficiente, combinando a filtragem espacial e temporal no acesso aos buckets de coordenadas (MONGODB, 2025).

Já o InfluxDB, ao usar a função geo.filterRows(), precisou converter a área consultada em células S2 e realizar verificações adicionais, o que aumentou o custo de processamento e leitura em disco (INFLUXDATA, 2025). Dessa forma, embora o InfluxDB seja otimizado para séries temporais, o MongoDB mostrou-se mais eficiente para consultas geoespaciais por região.

**Conclusões**

A pesquisa evidenciou diferenças claras de desempenho entre MongoDB e InfluxDB em cenários espaço-temporais aplicados à cafeicultura. O MongoDB apresentou maior eficiência na ingestão de grandes volumes de dados, tanto na carga inicial quanto nas atualizações incrementais, além de apresentar melhor

**“Do conhecimento acadêmico à transformação sustentável: inovação com validação científica”**

desempenho em consultas espaciais sobre regiões, especialmente em situações de alta concorrência.

Por outro lado, o InfluxDB mostrou vantagem em consultas pontuais, voltadas à recuperação de séries temporais de pixels específicos, principalmente quando foi utilizado o identificador espacial S2CellId como tag, evidenciando o impacto positivo de sua arquitetura orientada ao tempo e ao uso de índices internos para consultas geográficas localizadas.

Esses resultados reforçam a importância de alinhar a escolha do SGBD ao perfil de uso da aplicação: MongoDB é mais indicado para cenários de ingestão intensiva e análises espaciais regionais, enquanto InfluxDB é mais adequado para consultas temporais localizadas. Assim, a pesquisa oferece subsídios técnicos para a seleção de tecnologias em projetos de sensoriamento remoto voltados para a cafeicultura, contribuindo para decisões mais eficientes em termos de desempenho e gestão de dados espaço-temporais.

### Agradecimentos

Agradece-se à Universidade Federal de Itajubá (UNIFEI) pelo suporte institucional e pelas condições oferecidas para o desenvolvimento desta pesquisa. Também é reconhecido o apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) por meio da bolsa concedida, que possibilitou a realização deste trabalho.

### Referências

Queiroz, D. M. d., Valente, D. S. M., Pinto, F. d. A. d. C., and Borém, A. (2022). Agricultura digital. Oficina de Textos.

Bassoi, L. H., Inamasu, R. Y., Bernardi, A. C. d. C., Vaz, C. M. P., Speranza, E. A., and Cruvinel, P. E. (2019). Agricultura de precisão e agricultura digital. TECCOGS.

Formaggio, A. R. and Sanches, I. D. (2017). Sensoriamento remoto em Agricultura. Oficina de Textos

Hachimi, C. E., Belaqziz, S., Khabba, S., Sebbar, B., Dhiba, D., and Chehbouni, A. (2023). Smart Weather Data Management Based on Artificial Intelligence and BigData Analytics for Precision Agriculture. Agriculture.

Choi, W. G., Kim, S., Kim, J., Song, M.-H., and Lee, S.-S. (2022). Real-Time Data Processing Framework for Things with time-series and spatial features. In 2022 13th ICTC.

Batina, A. (2023). Data Cubes – A Modern Approach for Handling Earth Observation Data. In 2023 International Conference on Earth Observation and Geo-Spatial Information (ICEOGI), pages 1–6.

Mehmood, N. Q., Culmone, R., and Mostarda, L. (2017). Modeling temporal aspects of sensor data for MongoDB

NoSQL database. Journal of Big Data, 4(1):8.

InfluxDB Documentation. InfluxDB OSS v2 Documentation. DB-Engines. DB-Engines Ranking.

Zehra, S. N. (2017). Time Series Databases and InfluxDB.

Tripathi, P., Miraz, M. H., and Joshi, S. (2023). Comparative Analysis of MongoDB and InfluxDB for Time Series Data Management in IoT Environments: A Study on Performance, Scalability, and Concurrency. In 2023 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA).

Kim, S., Hoang, Y., Yu, T. T., and Kanwar, Y. S. (2023). GeoYCSB: A Benchmark Framework for the Performance and Scalability Evaluation of Geospatial NoSQL Databases. Big Data Research, 31:100368.

MongoDB Documentation. MongoDB Documentation.