

Trabalhos	Questões de pesquisa de Protocolo	Resultados	Autores	Ano
Testing Spatial Data Deliverance in SQL and NoSQL Database using NodeJS Fullstack Web App.	O artigo não especifica explicitamente as questões de pesquisa do protocolo que guiaram o estudo.	Os resultados mostram que o banco de dados NoSQL MongoDB supera significativamente o banco de dados relacional SQL PostGIS em termos de carregamento de dados geoespaciais, à medida que a quantidade de dados aumenta. O autor conclui que, para aplicativos web GIS que precisam carregar grandes quantidades de dados espaciais, um banco de dados NoSQL como o MongoDB oferece melhor desempenho.	Laksono, Dany	2018
Research on High Availability Architecture of SQL and NoSQL	O artigo não especifica explicitamente as questões de pesquisa do protocolo que guiaram o estudo.	É concluído que, embora os bancos de dados NoSQL tenham vantagens significativas em termos de escalabilidade e flexibilidade, eles ainda não podem substituir completamente os bancos de dados SQL quando se trata de lidar com dados estruturados. A arquitetura híbrida proposta busca lidar de forma eficiente com diferentes tipos de dados, oferecendo uma abordagem universal para o gerenciamento de dados. No entanto, é mencionado que mais estudos são necessários para aprofundar as questões relacionadas aos middlewares e às políticas de agendamento.	Wang Z.; Wei Z.; Liu H.	2017
Performance optimization of applications based on non-relational databases	O artigo não especifica explicitamente as questões de pesquisa do protocolo que guiaram o estudo.	A conclusão do estudo é que a implementação e combinação dos dois sistemas de banco de dados não relacionais, MongoDB e Elasticsearch, resultaram em uma melhoria significativa na velocidade de leitura enquanto ainda mantêm a integridade dos dados. As vantagens principais dessa abordagem incluem a boa integridade dos dados, alta velocidade de leitura, facilidade de implementação e uso, além da documentação abrangente para ambas as tecnologias, o que facilita trabalhos de futuras expansões.	Vokorokos L.; Uchnar M.; Leščišin L.	2016
Performance evaluation of SQL and MongoDB databases for big e-commerce data	Os autores projetaram experimentos com várias operações, incluindo leitura, escrita, exclusão e seleção, em ambos os bancos de dados. Eles usaram um esquema típico de comércio eletrônico e realizaram operações em uma quantidade significativa de dados. As operações foram realizadas nas mesmas condições para ambos os bancos de dados. Eles usaram a linguagem Java e a interface JDBC para interagir com os bancos de dados.	Inserção: MongoDB superou o SQL Server em operações de inserção em todos os tamanhos de dados testados. Exclusão: O desempenho do MongoDB foi melhor do que o SQL Server em operações de exclusão. Atualização: MongoDB superou significativamente o SQL Server em operações de atualização, especialmente com grandes quantidades de dados. Seleção: MongoDB também superou o SQL Server em operações de seleção em diferentes escalas de dados.	Aboutorabi S.H.; Rezapour M.; Moradi M.; Ghadiri N.	2015
Performance Evaluation for MongoDB and Redis: A Comparative Study	Os autores utilizaram um conjunto de dados composto por informações de funcionários de empresas, com diferentes tamanhos de registros (1000, 100000, 1000000) e 12 campos. Eles conduziram testes para as operações de inserção, seleção, atualização e exclusão em ambas as bases de dados, utilizando scripts escritos em PHP.	Os resultados dos testes mostraram que o Redis teve um desempenho melhor do que o MongoDB na operação de inserção quando o conjunto de dados era grande. No entanto, ambas as bases de dados tiveram desempenho semelhante na operação de inserção quando o conjunto de dados era pequeno. Para a operação de atualização, o Redis também apresentou um desempenho superior em diferentes cenários.	Alraddadi S.; Almotairi S.	2019
Optimizing the Research of DNA Sequences in a NoSQL Document Database: A Preliminary Study	O estudo se concentra na otimização da pesquisa de sequências de DNA em um banco de dados NoSQL orientado a documentos. Os autores investigam a busca por sequências nucleotídicas em regiões genômicas, considerando o uso de funções de hash para mapear sequências de nucleotídeos de tamanho variável para dados de tamanho fixo. Eles comparam a eficiência do banco de dados MongoDB, um sistema de gerenciamento de banco de dados NoSQL, com o MySQL, um sistema de gerenciamento de banco de dados SQL, para realizar essas pesquisas.	Os resultados dos experimentos mostraram que o MongoDB, especialmente quando instalado em um único servidor, proporcionou melhores desempenhos em comparação com o MySQL para a pesquisa de sequências nucleotídicas em genomas. No entanto, o MySQL apresentou melhores tempos de resposta ao buscar sequências específicas em comparação com o uso de expressões regulares no MongoDB. Os autores sugerem que, em trabalhos futuros, podem explorar modelos de dados alternativos para melhorar a pesquisa de sequências em diferentes regiões genômicas e também investigar a aplicação de aprendizado profundo em grandes conjuntos de dados genômicos.	Celesti F.; Celesti A.; Galletta A.; Fazio M.; Villari M.	2019
Comparing NoSQL MongoDB to an SQL DB	<p>Questão de Pesquisa 1: Qual é o desempenho das operações de inserção entre o MongoDB e o SQL Server em um banco de dados estruturado de tamanho moderado?</p> <p>Resultados: O desempenho de inserção foi comparado entre MongoDB e SQL Server. Não houve um claro vencedor; o desempenho variou dependendo do caso de teste.</p> <p>Questão de Pesquisa 2: Como o desempenho das operações de atualização difere entre o MongoDB e o SQL Server?</p> <p>Resultados: O desempenho das atualizações foi analisado para diferentes tipos de atualizações. MongoDB superou o SQL Server quando as atualizações eram baseadas nas chaves primárias.</p> <p>Questão de Pesquisa 3: Qual é o desempenho das operações de seleção em termos de consultas simples e complexas entre o MongoDB e o SQL Server?</p> <p>Resultados: Foram comparados os tempos de execução das consultas simples e complexas entre MongoDB e SQL Server. MongoDB teve desempenho superior, exceto para consultas de agregação, nas quais o SQL Server superou o MongoDB.</p>	<p>Os resultados das operações de inserção mostraram que o desempenho variou entre os dois sistemas, sem um vencedor claro em todos os casos.</p> <p>Em termos de atualizações, o MongoDB teve melhor desempenho para atualizações baseadas em chaves primárias, enquanto o SQL Server teve vantagem em atualizações baseadas em outros atributos.</p> <p>As consultas simples (seleção por chave primária ou atributo específico) mostraram que o MongoDB teve um desempenho geral melhor do que o SQL Server.</p> <p>As consultas complexas que envolviam joins e funções de agregação tiveram um melhor desempenho no SQL Server em comparação com o MongoDB.</p> <p>A conclusão geral foi que o MongoDB pode ser mais adequado para bancos de dados maiores com esquemas em constante mudança, enquanto o SQL Server pode ser mais apropriado para bancos de dados estruturados com consultas complexas envolvendo funções de agregação.</p>	Parker Z.; Poe S.; Vrbsky S.V.	2013

Adapting TPC-C Benchmark to Measure Performance of Multi-Document Transactions in MongoDB	<p>O estudo abordou as seguintes questões de pesquisa:</p> <p>Como adaptar o benchmark TPC-C para o MongoDB: O artigo detalha as modificações feitas no código do benchmark PyTPCC para que ele funcione com as características e melhores práticas do MongoDB, incluindo suporte a transações multi-documento e otimizações de esquema e consultas.</p> <p>Qual é o impacto das adaptações no desempenho: O estudo investigou como as modificações feitas no benchmark afetaram o desempenho das transações multi-documento no MongoDB, comparando diferentes cenários e configurações.</p> <p>Quais são as melhores práticas para otimizar o desempenho: O estudo explora as práticas que levaram a melhorias de desempenho, como otimizações de esquema, índices, operações em lote e outras otimizações de código.</p>	<p>Os principais resultados e conclusões do estudo foram os seguintes:</p> <p>Adaptação do Benchmark: Foi possível adaptar o benchmark TPC-C para o MongoDB, mas foram necessárias modificações significativas para alinhar o benchmark com as melhores práticas e recursos do MongoDB.</p> <p>Desempenho do MongoDB: Os testes demonstraram que o MongoDB é capaz de lidar com transações multi-documento e obter um bom desempenho, mesmo quando comparado a bancos de dados tradicionais.</p> <p>Impacto das Modificações: As modificações feitas no esquema, índices e operações de escrita tiveram um impacto positivo no desempenho das transações multi-documento, resultando em melhorias significativas.</p> <p>Escolha do Esquema: O estudo mostrou que, ao adotar práticas de design de esquema de documentos, o desempenho foi melhor do que ao usar um esquema normalizado, contrariando a expectativa de que a normalização sempre proporcionaria melhor desempenho.</p> <p>Uso de Operações Otimizadas: A otimização das operações, como a combinação de seleção e atualização em uma única operação, levou a reduções no número de operações e melhorias no desempenho.</p> <p>Escalabilidade e Custo: Os testes realizados em diferentes tamanhos de clusters mostraram que o aumento no desempenho nem sempre estava diretamente relacionado ao aumento no custo, indicando que é importante considerar a relação custo-benefício ao escolher a configuração do cluster.</p>	Kamsky A.	2018
A qualitative analysis of the performance of MongoDB vs MySQL Database based on insertion and retrieval operations using a web/android application to explore Load Balancing – Sharding in MongoDB and its advantages	<p>As questões de pesquisa que foram abordadas incluem:</p> <p>Necessidade de Bancos de Dados NoSQL: A pesquisa explora a necessidade de bancos de dados NoSQL, como o MongoDB, devido ao crescimento explosivo de dados não estruturados e à inadequação dos bancos de dados relacionais tradicionais, como o MySQL, para lidar com grandes volumes de dados.</p> <p>Comparação de Desempenho: A pesquisa compara o desempenho do MongoDB e do MySQL em termos de tempos de inserção e recuperação. A hipótese é que o MongoDB será mais eficiente em lidar com grandes cargas de dados devido à sua capacidade de balanceamento de carga.</p> <p>Características do MongoDB: A pesquisa explora as características distintas do MongoDB, como seu modelo de documento, sharding automático e capacidade de balanceamento de carga, que o tornam uma escolha preferida em aplicações com cargas de dados intensas.</p>	<p>Em resumo, a pesquisa demonstrou que o MongoDB apresenta vantagens significativas em termos de desempenho para operações de inserção e recuperação em comparação com o MySQL, especialmente em cenários com grande quantidade de dados. A capacidade de balanceamento de carga e o modelo de documento flexível do MongoDB contribuem para sua eficácia em lidar com as demandas atuais de gerenciamento de dados.</p>	Patil M.M.; Hanni A.; Tejeshwar C.H.; Patil P.	2017