

Relatório de Benchmark – TPC-H (Q1–Q3) em MySQL e MongoDB

CARLOS EDUARDO OBRISTO
GUILHERME EDUARDO CORREA RAPOSO

15/11/2025

1 Análise Comparativa das Consultas TPC-H (Q1, Q2 e Q3)

O objetivo deste relatório foi comparar o desempenho das três primeiras consultas do TPC-H executadas em dois bancos de dados: MySQL (modelo relacional) e MongoDB (modelo documental). Durante os testes, foi necessário criar índices mínimos nas tabelas, pois a segunda consulta chegou a exceder 30 minutos de execução sem otimizações. Os resultados obtidos pela nossa equipe foram:

Query	MySQL (s)	MongoDB (s)	Diferença
Q1	53.815	19.554	-34.261
Q2	0.470	0.154	-0.316
Q3	26.771	49.969	+23.197

Hardware utilizado no Benchmark

- Processador: Ryzen 5 5600X (Cache 35MB, 3.7GHz)
- Memória: 2x Memória Fury Beast 16GB DDR4 (Frequência: 3200MHz)
- SSD: SSD 480GB WD Green PC SN350 (Leitura: 545MB/s)

Query 1 (Q1) — Agregações sobre uma única tabela

A Q1 consiste em operações de: filtro por data, somas, médias e agrupamentos sobre a tabela (`lineitem`). Como não envolve relacionamentos, o MongoDB apresentou desempenho superior. O mecanismo de agregação do MongoDB é otimizado para leituras sequenciais e cálculos diretos sobre documentos, enquanto o MySQL executa a mesma operação com maior sobrecarga de manipulação relacional. Assim, o MongoDB foi aproximadamente três vezes mais rápido.

Query 2 (Q2) — Mínimo custo por fornecedor

A Q2 envolve encontrar o menor custo de fornecimento para peças específicas, exigindo filtragens e agrupamentos mais complexos. Após criar índices básicos nos dois bancos, ambos tiveram tempos de execução baixos, mas o MongoDB ainda foi ligeiramente mais rápido. Isso se deve ao fato de que parte dos dados já estava parcialmente denormalizada no MongoDB, reduzindo custos de junção. Já o MySQL executa uma subconsulta correlacionada, que tende a ser mais custosa.

Query 3 (Q3) — Junções e filtragem por datas

A Q3 foi a única em que o MySQL superou o MongoDB. Essa consulta envolve junções entre `customer`, `orders` e `lineitem`, além de filtros de data e agrupamento. O modelo relacional é mais eficiente nesse tipo de consulta, já que o MySQL otimiza joins clássicos com índices B-tree. No MongoDB, a mesma operação exige múltiplos `$lookup` e `$unwind`, resultando em maior custo computacional. Por isso, o tempo no MongoDB foi cerca do dobro.

Conclusão Geral

Os resultados mostram que cada banco possui vantagens em cenários distintos:

- O MongoDB foi mais rápido em consultas analíticas sem relacionamentos (Q1, Q2).
- O MySQL foi superior em consultas dependentes de múltiplas relações (Q3).

Assim, conclui-se que o desempenho depende diretamente do tipo de operação executada e da estrutura dos dados, e não existe um “vencedor absoluto” entre os dois sistemas.