

Pontifícia Universidade Católica de Minas Gerais
Laboratório de Experimentação de Software
GraphQL vs REST - Um experimento controlado

Giovanni Bogliolo Sirihal Duarte
Luiz Gustavo Mendes Santos
Pedro Ramos Vidigal

Belo Horizonte
2024

1. Desenho do Experimento

Hipóteses Nula e Alternativa

RQ1: "Respostas às consultas GraphQL são mais rápidas que respostas às consultas REST?"

- **H0 (Hipótese Nula):** Não há diferença significativa no tempo de resposta entre consultas GraphQL e REST.
- **H1 (Hipótese Alternativa):** Consultas GraphQL têm um tempo de resposta significativamente mais rápido do que consultas REST.

RQ2: "Respostas às consultas GraphQL têm tamanho menor que respostas às consultas REST?"

- **H0 (Hipótese Nula):** Não há diferença significativa no tamanho das respostas entre consultas GraphQL e REST.
- **H1 (Hipótese Alternativa):** Consultas GraphQL têm um tamanho de resposta significativamente menor do que consultas REST.

Variáveis Dependentes

- **Tempo de Resposta:** Medido em milissegundos (ms), corresponde ao tempo que cada API leva para retornar uma resposta completa ao cliente.
- **Tamanho da Resposta:** Medido em bytes (ou kilobytes), representa o volume de dados enviados do servidor ao cliente.

Variáveis Independentes

- **Tipo de API:** Este é o fator principal, variando entre dois níveis: API GraphQL e API REST.
- **Tipo de Consulta:** Pode incluir diferentes tipos de consulta (ex.: consultas simples, consultas com múltiplos relacionamentos e consultas de agregação).
- **Quantidade de Dados Solicitados:** Refere-se à quantidade de informação requerida em cada consulta, podendo variar de pequenas a grandes consultas, o que permite avaliar a eficiência em diferentes cenários de carga.

Tratamentos

- **Tratamento 1:** Realização de consultas utilizando GraphQL.
- **Tratamento 2:** Realização de consultas utilizando REST.

Cada tratamento envolve a realização de consultas idênticas em ambos os tipos de API para garantir a comparação equivalente de tempo e tamanho das respostas.

Objetos Experimentais

- **End Points:** Conjuntos de APIs REST e GraphQL que fornecem os mesmos dados.
- **Consultas (Requests):** A seleção de tipos específicos de consultas (por exemplo, consultas detalhadas contra agregadas) para avaliar desempenho e tamanho das respostas em situações comparáveis.

Tipo de Projeto Experimental

Este projeto é um experimento comparativo controlado, onde ambos os tratamentos (REST e GraphQL) são testados sob condições controladas e repetidas para reduzir a variabilidade e permitir uma comparação justa. As medições são tomadas para cada tratamento usando endpoints equivalentes, com o mesmo hardware e rede, para isolar as diferenças entre REST e GraphQL.

Quantidade de Medições

Para garantir a precisão dos resultados e minimizar o impacto de variações ocasionais nos dados, realizaremos 15 execuções para cada tipo de consulta (simples, complexa e de agregação) em ambas as APIs (GraphQL e REST). Essa quantidade de medições permite obter uma média confiável dos tempos e tamanhos das respostas, possibilitando uma análise estatística robusta para testar as hipóteses do experimento. Além disso, as 30 execuções ajudam a reduzir o efeito de fatores externos, como variações momentâneas na rede ou no servidor, aumentando a confiabilidade e a consistência dos resultados obtidos.

Ameaças à Validade

- **Validade Interna:** Flutuações no desempenho do servidor, variabilidade de rede e cachê de resposta podem impactar o tempo de resposta. Para mitigar isso, recomenda-se repetir o teste várias vezes e, se possível, em diferentes horários.
- **Validade Externa:** O experimento pode ser limitado a APIs e consultas específicas, e os resultados podem não ser generalizáveis para todos os contextos de uso. Ampliar o teste para diferentes tipos de dados e APIs ajudaria a aumentar a validade externa.
- **Validade de Construção:** Como tempo e tamanho são medidos deve ser consistente e precisa. Deve-se usar ferramentas confiáveis para medir o tempo de resposta (como logs detalhados de tempo de execução).