

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).

2. Metodologia

Descrição do Ambiente Experimental

Os experimentos foram realizados em um ambiente computacional padronizado para garantir a reprodutibilidade. As configurações incluem:

- **Sistema Operacional:** Windows 11 Pro
- **Hardware:**
 - Processador Intel Core i5-10300f
 - Memória RAM de 16 GB
 - Conexão de internet de alta velocidade (200 Mbps)
- **Bibliotecas e Ferramentas:**
 - Python 3.10
 - Bibliotecas: `requests` para chamadas HTTP, `time` para medição de desempenho.
- **Token de Acesso à API:** Um token pessoal foi gerado para autenticação na API do GitHub. Ele permite acesso a endpoints REST e GraphQL.

Descrição do Experimento

O experimento consistiu em realizar consultas repetitivas à API do GitHub usando tanto o formato REST quanto o formato GraphQL, conforme descrito a seguir:

1. Tipos de Consultas

- **Consulta Simples:** Recupera informações básicas sobre repositórios populares.
 - REST: `/repositories?per_page=5`
 - GraphQL: Recupera o nome e proprietário de cinco repositórios mais estrelados.
- **Consulta Complexa:** Recupera informações detalhadas de um repositório específico.
 - REST: `/repos/octocat/hello-world`
 - GraphQL: Recupera detalhes como nome, descrição, número de estrelas, forks e issues.
- **Consulta de Agregação:** Recupera estatísticas e dados agregados.
 - REST:
`/search/repositories?q=stars:>100000&per_page=5`
 - GraphQL: Consulta o número de repositórios com mais de 100.000 estrelas e seus dados resumidos.

2. Execuções

Para cada tipo de consulta e cada abordagem (REST e GraphQL), foram realizadas **15 execuções consecutivas**, totalizando 90 chamadas à API:

- 15 consultas simples (REST e GraphQL).
- 15 consultas complexas (REST e GraphQL).
- 15 consultas de agregação (REST e GraphQL).

3. Medições

- **Tempo de Resposta:** Medido com a função `time.time()` antes e depois de cada conjunto de 15 consultas.
- **Tamanho das Respostas:** Calculado em bytes usando o tamanho do conteúdo retornado pela resposta HTTP (`response.content`).

4. Coleta e Armazenamento de Dados

- Os tempos totais e os tamanhos das respostas foram registrados separadamente para cada tipo de consulta e abordagem.
- As medições foram realizadas em intervalos de tempo próximos para minimizar a influência de variações externas, como flutuações de rede.

3. Resultados obtidos

REST

| | Tempo em segundos | Tamanho em bytes |
|-----------|-------------------|------------------|
| Simples | 4.86 | 7755 bytes |
| Complexa | 6.43 | 5190 bytes |
| Agregação | 4.95 | 7890 bytes |

GraphQL

| | Tempo em segundos | Tamanho em bytes |
|-----------|-------------------|------------------|
| Simples | 7.04 | 63076 bytes |
| Complexa | 5.50 | 76935 bytes |
| Agregação | 8.83 | 439545 bytes |

4. Análise de resultados

Nesta seção, apresentamos os resultados do experimento para as duas abordagens de consulta (GraphQL e REST) em relação às métricas de tempo de resposta e tamanho total das respostas para cada tipo de consulta.

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

Os tempos totais de execução para cada abordagem foram medidos para 15 execuções consecutivas por tipo de consulta. Os resultados obtidos são:

| Tipo de Consulta | GraphQL (s) | REST (s) | Diferença (%) |
|------------------|-------------|----------|---------------|
| Simples | 4.86 | 7.04 | -30.97% |
| Complexa | 6.43 | 5.50 | +16.91% |
| Agregação | 4.95 | 8.83 | -43.94% |

Análise:

- Nas consultas **simples** e **de agregação**, o GraphQL apresentou desempenho superior, com tempos de resposta 30,97% e 43,94% menores, respectivamente.
- Na consulta **complexa**, o REST foi 16,91% mais rápido. Esses resultados indicam que a vantagem do GraphQL em relação ao tempo de resposta pode depender da natureza da consulta, sendo mais eficiente em consultas que exigem dados estruturados ou agregados.

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

Os tamanhos totais das respostas em bytes foram calculados para 15 execuções consecutivas por tipo de consulta. Os resultados são apresentados abaixo:

| Tipo de Consulta | GraphQL (bytes) | REST (bytes) | Diferença |
|------------------|-----------------|--------------|-----------|
| Simples | 7,755 | 63,076 | 55,321 |
| Complexa | 5,190 | 76,935 | 71,745 |
| Agregação | 7,890 | 439,545 | 431,655 |

Análise:

- Em todas as consultas, as respostas da API GraphQL foram significativamente menores que as da API REST, com reduções que variaram de 55,321 (consulta simples) a 431,65 (consulta com agregação).
- Isso ocorre porque o GraphQL permite selecionar apenas os dados necessários para a consulta, enquanto o REST frequentemente retorna informações adicionais que podem não ser úteis.