

# Relatório de Experimento Controlado: Análise de Desempenho GraphQL vs REST

**1. Introdução** O objetivo deste experimento foi avaliar quantitativamente os benefícios e desvantagens da adoção de uma API GraphQL em detrimento de uma API REST tradicional. O estudo focou em duas Perguntas de Pesquisa (RQs):

- **RQ1:** Respostas às consultas GraphQL são mais rápidas que respostas às consultas REST?
- **RQ2:** Respostas às consultas GraphQL têm tamanho menor que respostas às consultas REST?

As hipóteses levantadas foram: para RQ1, que haveria variação de tempo devido ao processamento da query; para RQ2, que o GraphQL apresentaria redução significativa de payload ao eliminar o *overfetching*.

**2. Metodologia** Para possibilitar a reprodução do experimento, o seguinte ambiente foi configurado:

- **Objetos Experimentais:** Utilizou-se a API pública "The Rick and Morty API", acessando os endpoints `/api/character` (REST) e `/graphql` (GraphQL).
- **Ferramentas:** Scripts desenvolvidos em linguagem Python (3.14), utilizando a biblioteca `requests` para chamadas HTTP e `pandas` para coleta de dados.
- **Tratamentos:** Comparou-se uma requisição REST padrão (trazendo todos os dados do personagem) contra uma query GraphQL otimizada (buscando apenas `name` e `status`).
- **Execução:** Foram realizadas **50 medições** (trials) sequenciais para cada tecnologia, intercaladas para minimizar interferências de oscilação de rede momentânea. O experimento foi executado em ambiente Windows, conectado via banda larga.

**3. Resultados Obtidos** A análise estatística dos dados coletados apresentou os seguintes resultados:

**3.1. Tempo de Resposta (RQ1)** O REST apresentou melhor desempenho em latência absoluta.

- **Média REST:** 286.01 ms
- **Média GraphQL:** 336.15 ms
- **Desvio Padrão:** O GraphQL apresentou maior instabilidade (62.59 ms) comparado ao REST (35.32 ms).
- **Conclusão RQ1:** O GraphQL foi, em média, **17.53% mais lento** que o REST neste cenário.

**3.2. Tamanho da Resposta (RQ2)** O GraphQL demonstrou eficiência superior na transferência de dados.

- **Média REST:** 19,496 bytes (~19 KB)
- **Média GraphQL:** 883 bytes (~0.8 KB)
- **Conclusão RQ2:** A resposta REST foi **22.1 vezes maior** que a do GraphQL.

**4. Discussão e Conclusão** Os resultados refutam a ideia de que o GraphQL é inherentemente "mais rápido" em tempo de processamento, mas confirmam sua eficiência em uso de banda.

A latência superior do REST (menor tempo) pode ser atribuída à simplicidade do endpoint, que entrega um recurso estático ou pré-montado, enquanto o servidor GraphQL precisa realizar o *parsing* e validação da query antes da execução. A maior variabilidade (Desvio Padrão) do GraphQL também sugere maior sensibilidade à complexidade do processamento no servidor.

Entretanto, para cenários de rede limitada (como dispositivos móveis em 4G/5G instável), a redução drástica de payload (de 19KB para 0.8KB) proporcionada pelo GraphQL justificaria o pequeno acréscimo no tempo de processamento, pois a economia de banda é o fator crítico.