



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS E INFORMÁTICA
UNIDADE EDUCACIONAL PRAÇA DA LIBERDADE
Bacharelado em Engenharia de Software

Carla d'Abreu Martins Vieira
Daniel Lyncon Gonçalves de Souza

GRAPHQL VS REST: UM EXPERIMENTO CONTROLADO
Laboratório de Experimentação de Software

Belo Horizonte

2021

Carla d'Abreu Martins Vieira
Daniel Lyncon Gonçalves de Souza

GRAPHQL VS REST: UM EXPERIMENTO CONTROLADO

Laboratório de Experimentação de Software

Trabalho prático de estudo da atividade de experimentação a fim de trazer uma análise comparativa entre APIs GraphQL e REST.

Professor José Laerte Pires Xavier Junior

Belo Horizonte

2021

SUMÁRIO

1. Introdução	4
2. Metodologia	4
3. Resultados	5
3.1 Testes de Hipótese	7
4. Conclusão	7

1. Introdução

A linguagem de consulta GraphQL proposta pelo Facebook, como método de implementação da API Web, representa uma alternativa à popular API REST. A linguagem é baseada em grafos e permite aos usuários consultar o banco de dados na forma de *schemas* para que o banco de dados possa ser exportado e consultado em um formato definido pelo provedor de API. Por outro lado, APIs criadas com base em métodos REST são baseadas em *endpoints*: operações predefinidas podem ser chamadas por clientes que desejam consultar, excluir, atualizar ou gravar dados na base. Desde o seu início, vários sistemas realizaram migrações entre as duas soluções, mantendo uma solução REST compatível, mas proporcionando os benefícios da nova linguagem de consulta proposta. No entanto, não está claro quais são os benefícios reais de adotar a API GraphQL em vez da API REST. Neste caso, o objetivo deste trabalho é realizar experimentos controlados para avaliar quantitativamente os benefícios na adoção do modelo API GraphQL e, para isso, foram propostas as seguintes questões de pesquisa(RQ):

RQ1. Respostas à consultas GraphQL são mais rápidas que respostas à consultas REST?

RQ2. Respostas à consultas GraphQL tem tamanho menor que respostas à consultas REST?

Como hipótese nula foi definido que consultas REST são mais rápidas e menores que consultas GraphQL. Nesse sentido, a hipótese alternativa é que consultas GraphQL são mais rápidas e menores que consultas REST.

2. Metodologia

Para responder às questões de pesquisa e então descartar ou não a hipótese nula, foi definido como medida de **tempo** a velocidade com que a API responde em milissegundos e, por **tamanho**, a quantidade de caracteres na resposta. Além do tipo de API analisado também foi realizado a comparação das requisições para **um** e **todos** atributos, e com ou sem **paginação** nas consultas.

Como objeto de estudo foi definido o GitHub por este prover ambas as APIs GraphQL e REST e possuir dados públicos de fácil acesso para realização das consultas. Dentro da API, definiu-se que seriam pesquisados 10% dos usuários que é possível encontrar em sua localização o termo “US” de *United States*. Em 02/06/2021, a quantidade encontrada era de 13999 usuários. Logo, foram buscados 1399 usuários, utilizando a ordenação padrão da API(quantidade de *followers*).

Para realização da coleta foi utilizado um script escrito em Python que utiliza a biblioteca **requests** para realizar as requisições para as APIs, a biblioteca **time** para

calcular o tempo que cada requisição levou. Também foi utilizado a biblioteca **json** para gerenciar os **TOKENs** de autenticação da API do GitHub, dado que as requisições REST possuem uma restrição de 60 por hora por token.

Foram realizadas 20 iterações por tipo de API, sendo elas 10 requisições buscando **um** atributo e outras 10 buscando **todos** atributos. Das 10 requisições por quantidade de atributos, 5 delas foram utilizadas de paginação da consulta e 5 delas não.

A cada iteração, foi utilizado o terminal para exibir os resultados do tempo e tamanho gastos naquela iteração. Posteriormente, os dados foram passados para um arquivo csv para realização dos testes de hipótese.

Para os testes de hipótese foi desenvolvido um script em R, utilizando-se dos testes de **Shapiro Wilk** e **teste da soma dos postos de Wilcoxon** para determinar se as amostras foram selecionadas a partir de populações que têm a mesma distribuição. A princípio foi cogitado o teste ANOVA, porém como não foi possível constatar a homogeneidade da variância, este foi descartado.

3. Resultados

A Tabela 1 mostra o tempo médio gasto em cada modalidade de requisição, considerando um atributo, todos atributos, sem paginação e com paginação para cada tipo de API. A princípio, a API rest se mostra mais rápida na média em todas as modalidades, com exceção da busca por apenas um atributo.

Tempo		Paginação			
		Sem Paginação		Com Paginação	
		GraphQL	REST	GraphQL	REST
Atributos	Um atributo	0,2910	0,3245	3,0824	3,0678
	Todos os Atributos	0,4368	0,2664	3,7271	3,0440

Tabela 1: Média dos Tempos em segundos

Na tabela 2, temos o desvio padrão do tempo gasto nas requisições. O desvio padrão se mostrou pequeno em comparação com as médias, indicando que os resultados da média estão próximos da realidade.

Tempo		Paginação			
		Sem Paginação		Com Paginação	
		GraphQL	REST	GraphQL	REST
Atributos	Um atributo	0,0595	0,0759	0,4213	0,4129
	Todos os Atributos	0,0428	0,0519	0,4344	0,3753

Tabela 2: Desvio Padrão do Tempo

Na tabela 3, temos a quantidade de caracteres dos retornos das APIs. Não há diferença entre as iterações para o tamanho, por isso não há cálculo de média.

Tamanho		Paginação			
		Sem Paginação		Com Paginação	
		GraphQL	REST	GraphQL	REST
Atributos	Um atributo	445	9.672	4.896	97.211
	Todos os Atributos	2.527	9.672	25.827	97.211

Tabela 3: Número de caracteres por resposta

3.1 Testes de Hipótese

Variáveis Independentes	P-Value	
	Tempo	Tamanho
Tipo de API GraphQL	0.001089	8.1e-06
Tipo de API REST	0.0002605	4.467e-05
Um Atributo	0.0002673	3.716e-06
Todos Atributos	0.0007967	4.467e-05
Com Paginação	0.01267	6.678e-05
Sem Paginação	0.003377	6.619e-05

Tabela 4: P valor teste Shapiro Wilk

A Tabela 4 mostra que, para um alfa de 0,05, a hipótese nula deve ser rejeitada, ou seja, não existem evidências de que alguma das amostras sigam uma distribuição normal.

Variáveis Independentes	P-Value	
	Tempo	Tamanho
Tipo de API	0.2766	3.722e-05
Quantidade de atributos	0.2423	0.1721
Uso de paginação	1.451e-11	3.722e-05

Tabela 5: P valor testes de postos Wilcoxon

Conforme a Tabela 5 e considerando o tempo e alfa igual a 0.05, não foi rejeitada a hipótese nula para os tipos de API e quantidade de atributos, ou seja, existem evidências que estes gastam a mesma mediana de tempo. Quanto à paginação, se rejeitou a hipótese nula, pois há evidências que esta interfere na mediana do tempo. Quanto ao tamanho, não foi rejeitada a hipótese nula para os tipos de API e para a paginação, ou seja, haveria evidências que os tamanhos são, de fato, diferentes. Contudo, para a quantidade de atributos não se rejeitou a hipótese nula. Nesse caso há indícios que a mediana de ambos não seja estatisticamente diferente. Com

auxílio do Gráfico 1 boxplot, notamos que a api REST possui a maior mediana de tamanho.

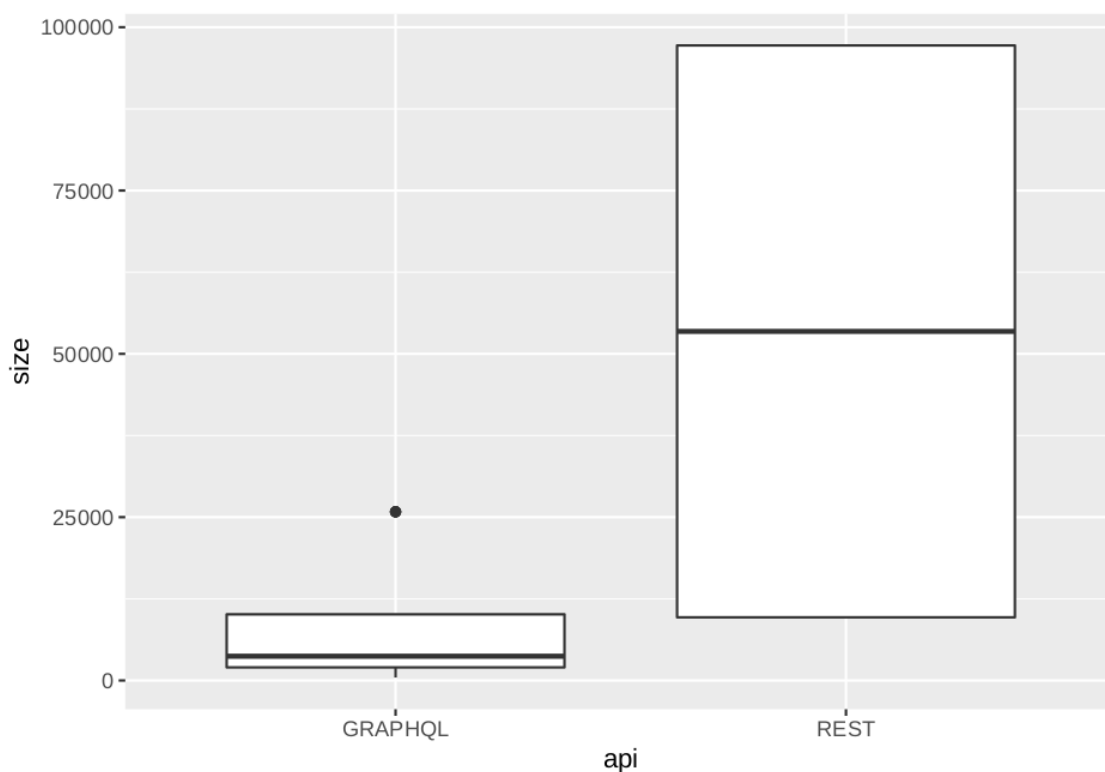


Gráfico 1: Tamanho do retorno X Tipo de API

4. Conclusão

Ao comparar o resultado dos experimentos com as hipóteses estabelecidas inicialmente juntamente com os testes de hipótese realizados, foi possível descartar a hipótese nula quando consideramos o tamanho do retorno nas diferentes APIs e também quanto ao uso de paginação. Foi confirmado então que as consultas GraphQL são menores, com exceção da quantidade de atributos, que não foi possível encontrar uma diferença estatística entre os dados da amostra.

Sobre a velocidade da resposta, a API rest se mostra superior quanto ao retorno na maioria dos casos. Porém o teste de hipótese indicaria que não haveria diferença estatística entre o tempo das consultas.