



Fundação Getulio Vargas  
Escola de Matemática Aplicada  
Curso de Ciência de Dados e Inteligência  
Artificial

Bruno Kauan Lunardon  
Fabrício Dalvi Venturim  
Luís Felipe de Almeida Marques  
Otávio Augusto Matos Alves  
Vinícius Antunes de Sousa

---

## Exercício: RPC

---

Computação Escalável  
Professor: Thiago Pinheiro de Araújo

Rio de Janeiro  
2024

## Solução Arquitetural e Decisões de Projeto

Entre as duas abordagens propostas, optamos pela primeira, que consiste em desenvolver o stub do servidor em C++ e o stub do cliente em Python. Essa escolha foi motivada pela facilidade de integração com o processo de ETL que implementamos na primeira parte do curso. Uma vantagem significativa desta abordagem é a redução do overhead associado à leitura e ao carregamento de arquivos, pois elimina a necessidade de persistir os eventos em um banco de dados externo.

### Servidor

O servidor é iniciado e fica aguardando um stream de mensagens, quando um stream chega ele cria um **DataFrame** e vai preenchendo-o com as linhas que chegam das mensagens até que um número de linhas pré-determinado seja alcançado ou o fim do stream ocorra, quando uma das duas condições é atingida o servidor insere o **DataFrame** em uma **ConsumerProducerQueue**.

### Cliente

Na implementação do cliente, utilizamos um mock para gerar os dados que servirão como carga para o sistema. Além disso, definimos os tipos e as quantidades de mensagens que serão enviadas através do gRPC. O cliente opera enviando mensagens que aderem ao padrão estabelecido pelo contrato e espera receber respostas que também seguem esse padrão

## Apresentação e discussão dos resultados do experimento

A primeira etapa dos experimentos executar o servidor contendo a pipeline foi executada e usamos 300 **DataFrames** para fazer a carga. Algumas estatísticas descritivas sobre o tempo de processamento dos **DataFrames** são como se segue:

| Estatística | Valor (ms) |
|-------------|------------|
| Mean        | 3.261126   |
| Std         | 1.619103   |
| Min         | 0.970076   |
| 25%         | 1.924050   |
| Median      | 2.844985   |
| 75%         | 4.656610   |
| Max         | 9.256620   |

Tabela 1: Estatísticas Descritivas do Tempo de Processamento dos dataframes

Para o teste aumentando o número de processo medimos o tempo total de processamento de 20 `DataFrames` a tabela é como se segue:

| <b>Tempo (mm:ss:ms)</b> | <b>Nº Processos</b> | <b>Nº Dataframes</b> |
|-------------------------|---------------------|----------------------|
| 01:13:170               | 1                   | 20                   |
| 00:28:690               | 5                   | 20                   |
| 00:14:800               | 10                  | 20                   |
| 00:11:490               | 20                  | 20                   |

Tabela 2: Resumo dos Tempos de Execução