

Jackes da Fonseca

RA: 190030291

# GRPC E PROTOS

Como solução de mensageria

10 de fevereiro de 2022

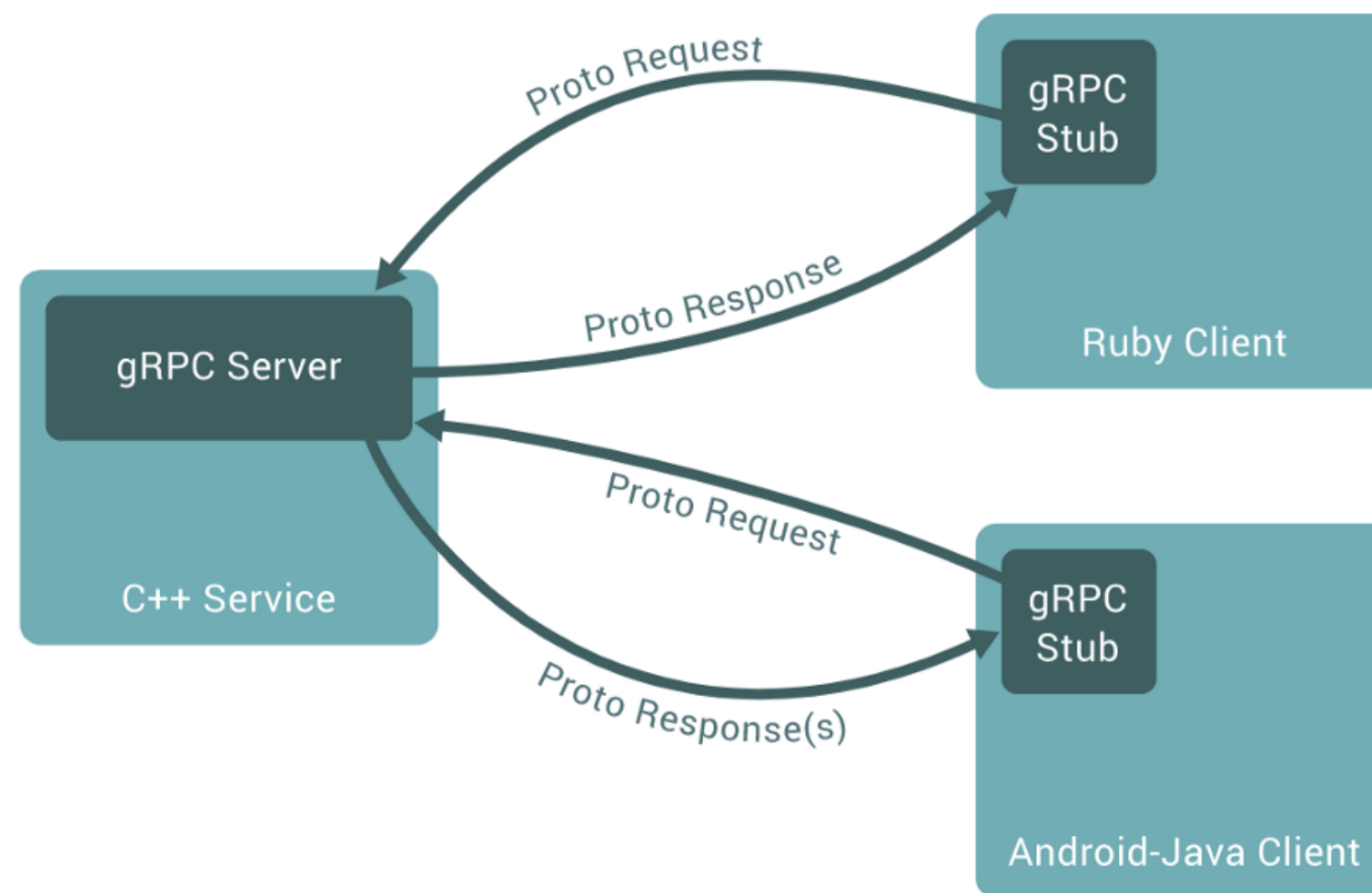
UnB - FGA

# Introdução ao gRPC

Uma introdução ao gRPC e buffers de protocolo

# O que é, afinal?

É uma estrutura de código aberto e alto desempenho criada pelo Google. O gRPC segue a semântica HTTP sobre HTTP/2, possibilitando comunicação entre diferentes sistemas via conexão de rede. No gRPC, um aplicativo cliente pode chamar diretamente um método em um aplicativo servidor em uma máquina diferente como se fosse um objeto local, facilitando a criação de aplicativos distribuídos.



# Visão Geral

A proposta do gRPC é que o cliente interaja com o servidor por meio de chamadas de funções simples, ou seja, de interfaces de códigos geradas automaticamente pela própria aplicação do gRPC. Isso significa que você precisa apenas implementar sua lógica de programação, o que facilita muito a adoção desse recurso.

# Vantagens no desenvolvimento de microserviços:

- Fácil contrato entre cliente e servidor;
- Melhor desempenho dos serviços;
- Features nativas do HTTP/2, como streaming de dados, load balance, monitoramento;

# Linguagens compatíveis

- C#
- C++
- Dart
- Go
- Java
- Kotlin
- Node
- Objective-C
- PHP
- Python
- Ruby

# gRPC x REST

O que mais diverge do REST – o protocolo de comunicação mais comum em arquitetura de microsserviços – é que, com gRPC, são usados caminhos estáticos para um melhor desempenho durante o despacho de chamadas, já que a análise de path params, query params e payload adiciona latência e complexidade à comunicação entre serviços.



# Alguns cenários uso de gRPC

- Sistemas multilíngues: com seu suporte de geração de código nativo para uma ampla gama de linguagens de desenvolvimento;
- Streaming em tempo real: Streaming bidirecional que permite que seu sistema envie e receba mensagens em tempo real sem esperar pela comunicação de resposta do cliente;
- Redes de baixa largura de banda e de baixa potência: o uso de mensagens Protobuf serializadas pelo gRPC permite mensagens leves, o que contribui para maior eficiência e velocidade em redes de baixa potência

# Introdução ao Protobuf

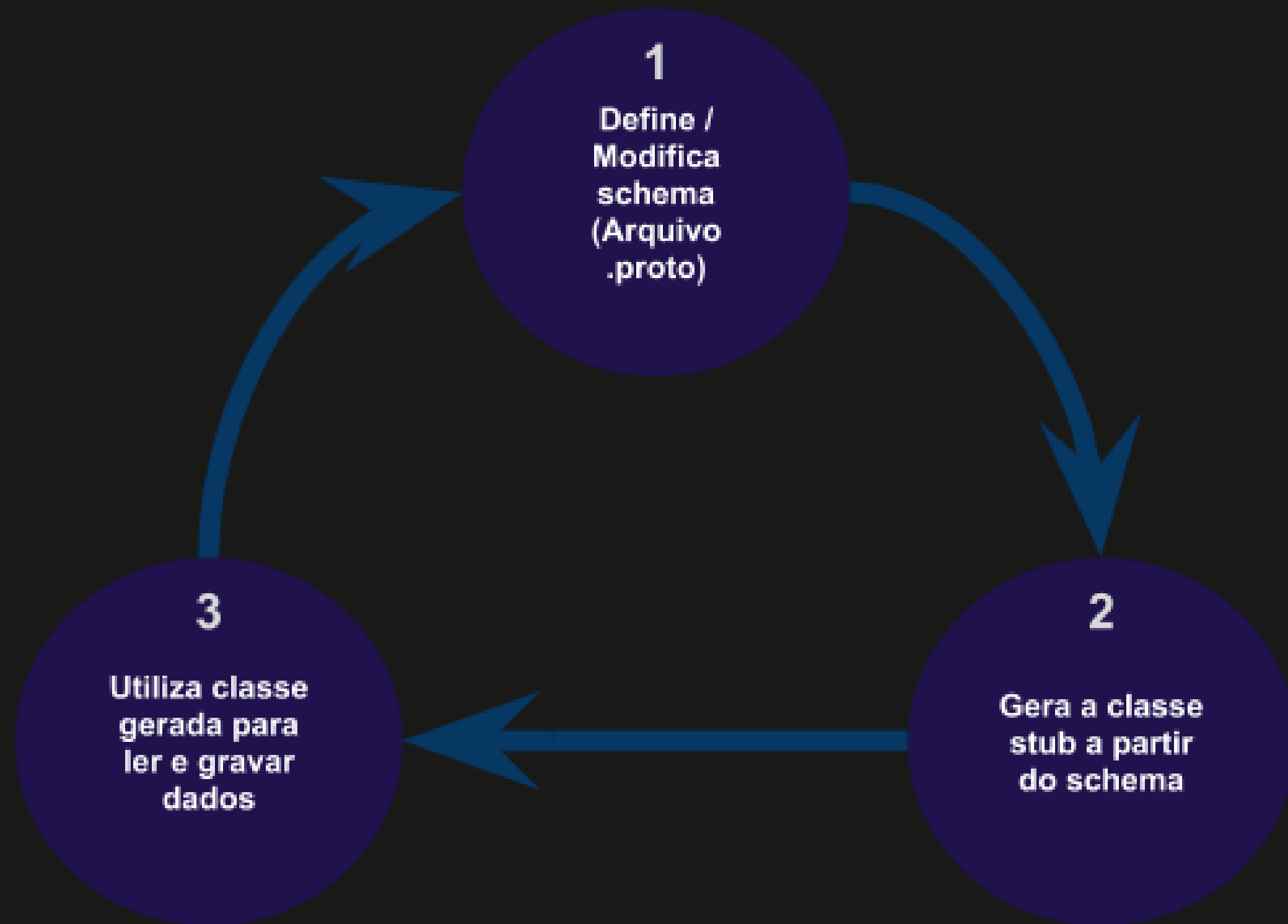
Uma alternativa ao JSON e XML

# Visão Geral

Normalmente, a comunicação destes serviços que utilizam a arquitetura SOA(Service-Oriented Architecture) ocorre através de HTTP usando o formato JSON. Embora o JSON possua muitas vantagens óbvias como formato de intercâmbio de dados — ele é legível para humanos, bem compreendido e normalmente funciona bem — ele também tem seus problemas. Para os casos no qual os dados não são consumidos diretamente por Javascript em navegadores, como por exemplo em serviços internos, pode ser que formatos estruturados — como o Protobuf — seja uma melhor opção para codificar dados.

# O que é Protobuf?

Protobuf (sigla de Protocol buffers) é um mecanismo criado e usado pelo Google para serializar dados estruturados. É independente de linguagem ou plataforma.



# Funcionamento

- Primeiro é definido como deseja que os dados sejam estruturados — em um arquivo de extensão `.proto`.
- Em seguida, esta definição é compilada e o resultado é um código-fonte automaticamente gerado na linguagem desejada — no momento que escrevo este post, as linguagens compatíveis são C++, C#, Go, Java e Python.
- Finalmente, código-fonte gerado é utilizado para gravar e ler os dados estruturados.
- Sempre que houver mudança na estrutura dos dados, o ciclo se repetirá

# Quando utilizar?

JSON, que significa JavaScript Object Notation, é simplesmente um formato de mensagem que surgiu em subconjunto da linguagem de programação Javascript. As mensagens JSON são trocadas em formato de texto e, hoje em dia, são completamente independentes e suportadas por praticamente todas as linguagens de programação. Protobuf, por outro lado, é mais que um formato de mensagem, é também um conjunto de regras e ferramentas para definir e trocar essas mensagens. Além disso, o Protobuf possui mais tipos de dados que o JSON, como enumerados e métodos, e também é muito usado em RPCs

# Referências

- 1 gRPC: O que é e como utilizá-lo na sua próxima aplicação. ZUP. Disponível em: <<https://www.zup.com.br/blog/grpc-o-que-e-beagle>>. Acesso em: 10 de fevereiro de 2022
- 2 proto-jason. Github. Disponível em: <<https://github.com/galvanidev/proto-json>>. Acesso em: 10 de fevereiro de 2022
- 3 Introdução ao HTTP/2. Web. Disponível em: <<https://developers.google.com/web/fundamentals/performance/http2>>. Acesso em: 10 de fevereiro de 2022

- 4 Introduction to gRPC. gRPC. Disponível em: <<https://grpc.io/docs/what-is-grpc/introduction/>>. Acesso em: 10 de fevereiro de 2022
- 5 IProtobuf - Uma alternativa ao JSON e XML. Medium. Disponível em: <<https://medium.com/trainingcenter/protobuf-uma-alternativa-ao-json-e-xml-a35c66edab4d>>. Acesso em: 10 de fevereiro de 2022
- 6 IProtocol Buffers. Wikipedia. Disponível em: <[https://pt.wikipedia.org/wiki/Protocol\\_Buffers](https://pt.wikipedia.org/wiki/Protocol_Buffers)>. Acesso em: 10 de fevereiro de 2022



**Obrigado!**