



## Aplicações realtime com gRPC

Leandro Lugaresi - GopherCon Brasil 2016

# STUBBY

I am your father!

# Open Source



# What is gRPC?

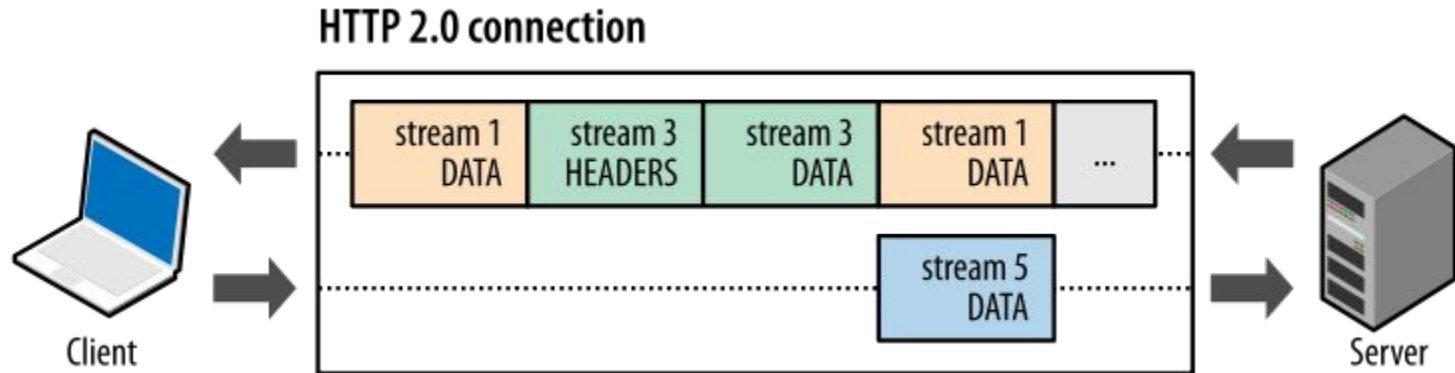
Universal RPC Framework

~10

languages

# HTTP/2

# Multiplexed Requests and Streams



# Protocol Buffers

## v3



# Structured data

```
message Message {  
  string id = 1;  
  string sender_id = 2;  
  oneof content {  
    ActionEvent event = 3;  
    string text = 5;  
    bytes audio = 6;  
    bytes image = 7;  
  }  
  google.protobuf.Timestamp creation_time = 8;  
  google.protobuf.Timestamp delivery_time = 9;  
}
```

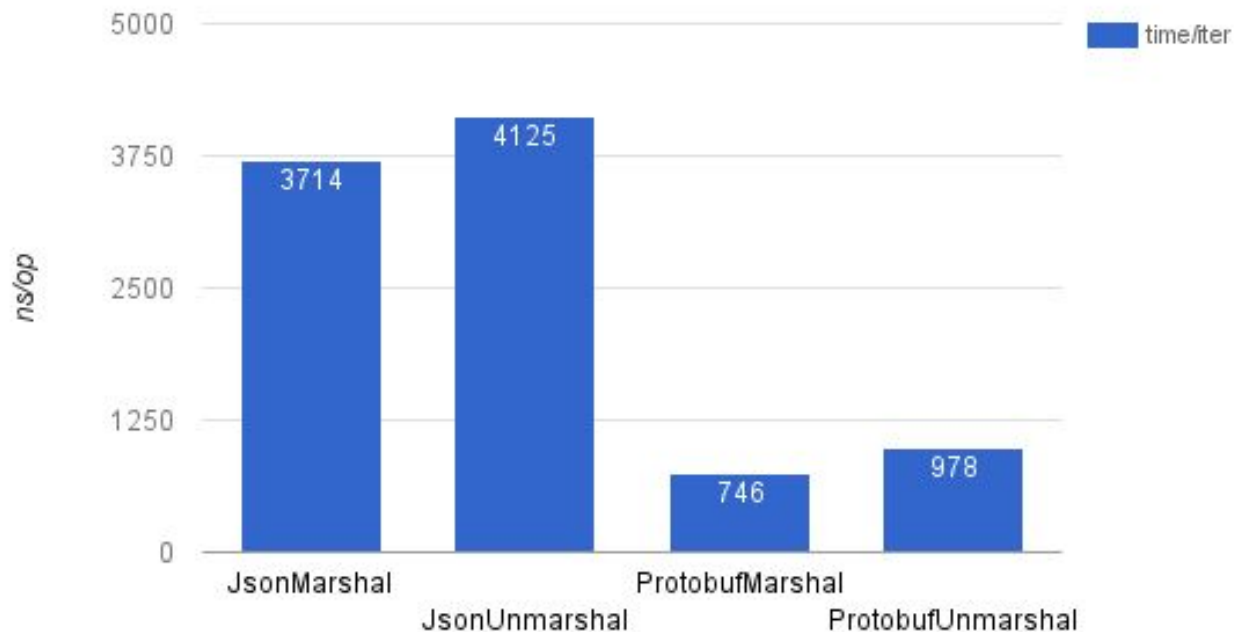
# Interface Definition Language (IDL)

```
service Chat {  
    // Request - Response  
    rpc GetHistory(GetHistoryRequest) returns (GetHistoryResponse) {}  
  
    // Server to client streaming  
    rpc GetEvents(GetEventsRequest) returns (stream ActionEvent) {}  
  
    // Client to server streaming  
    rpc SendHeartbeat(stream Heartbeat) returns (HeartbeatResponse) {}  
  
    // Bi-directional streaming  
    rpc Communicate(stream ChatMessage) returns (stream ChatMessage) {}  
}
```

# Fast

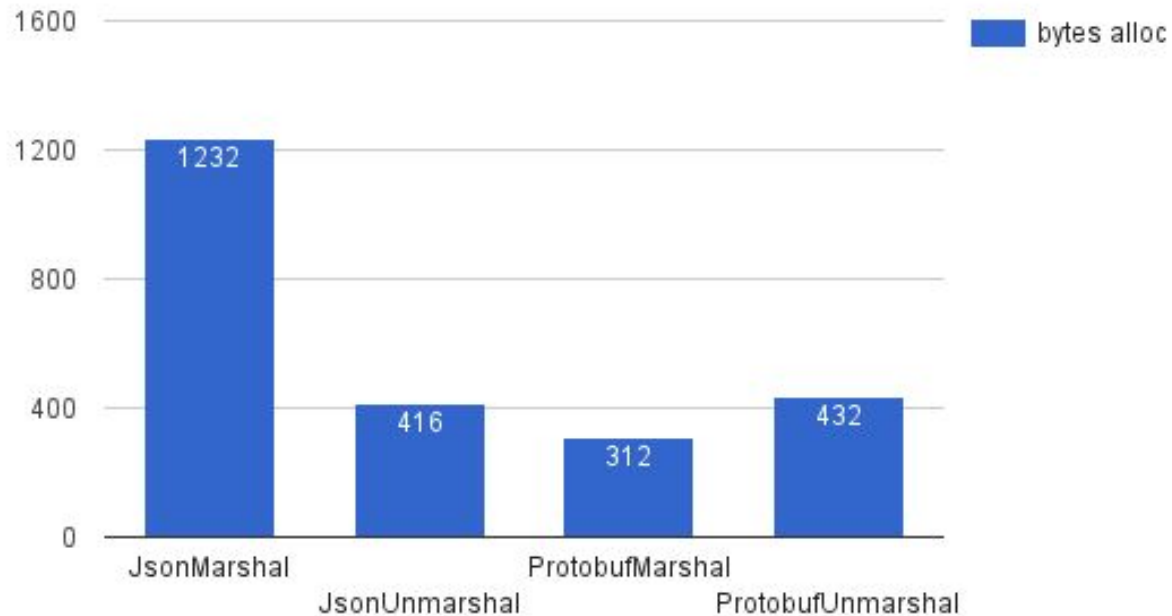


## Tempo por iteração



[https://github.com/alecthomas/go\\_serialization\\_benchmarks](https://github.com/alecthomas/go_serialization_benchmarks)

## Bytes alocados por iteração



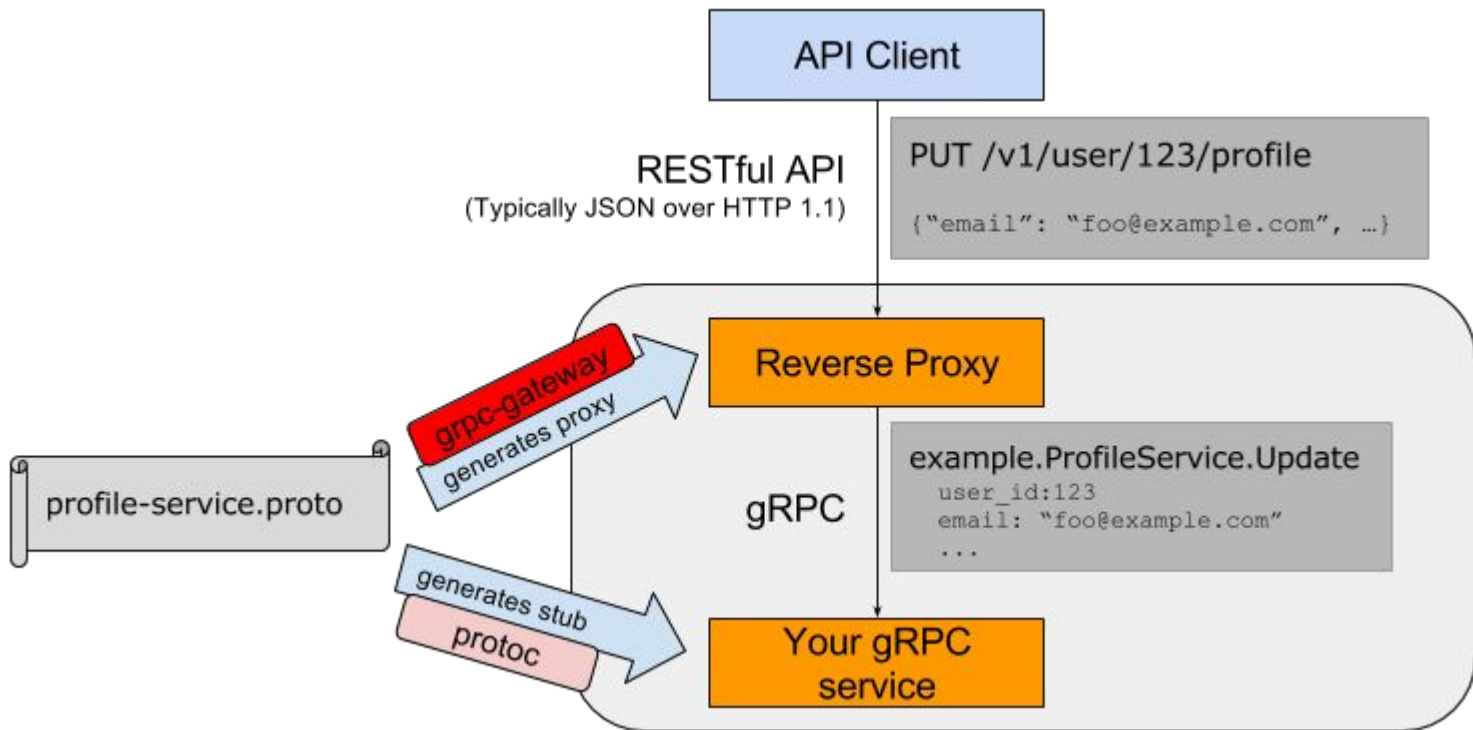
[https://github.com/alecthomas/go\\_serialization\\_benchmarks](https://github.com/alecthomas/go_serialization_benchmarks)

Save CPU usage  
and battery life



# Advanced gRPC

# grpc-gateway





# Context

Propagating timeouts and  
cancellations

# Interceptors

## Auth, logging, metrics, retry...

```
import "github.com/mwitkow/go-grpc-middleware"

myServer := grpc.NewServer(
    grpc.StreamInterceptor(grpc_middleware.ChainStreamServer(loggingStream, monitoringStream, authStream)),
    grpc.UnaryInterceptor(grpc_middleware.ChainUnaryServer(loggingUnary, monitoringUnary, authUnary),
)
```

---

# Client-side load balancing

# Referências

- <http://www.grpc.io>
- <https://developers.googleblog.com/2015/02/introducing-grpc-new-open-source-http2.html>
- <https://developers.google.com/protocol-buffers/>
- <https://github.com/grpc-ecosystem/grpc-gateway>
- <https://github.com/mwitkow/go-grpc-middleware>

# Thank you!

