Hands-on intro to gRPC

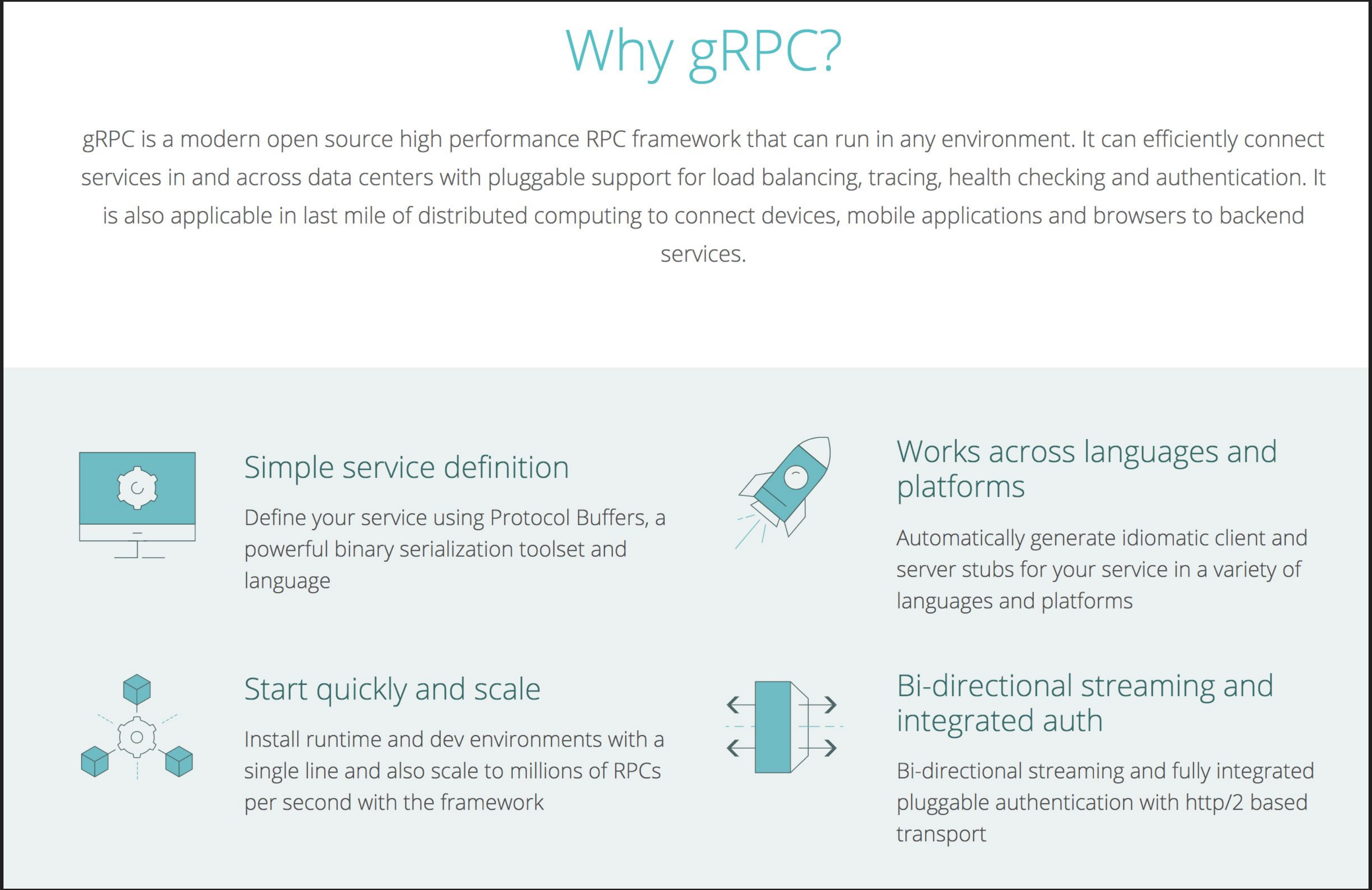
and its scaling pitfalls

Patrick Ellul - Assembly Payments

Serversiders Meetup - 30 October 2019

Agenda

- What is gRPC and why?
- Jump straight into code
    - The Proto file
    - Running Server and Client
    - Versioning and Backwards compatibility
    - Connection Robustness
    - Graceful Shutdown
- Scaling and Upgrading the Server
    - Goal: Zero Downtime, Zero Errors
    - Load balancing Long Lived Connection Complication
    - Demo of the Hard Way
    - The Cloud Native Way
- Questions



Why gRPC?

gRPC is a modern open source high performance RPC framework that can run in any environment. It can efficiently connect services in and across data centers with pluggable support for load balancing, tracing, health checking and authentication. It is also applicable in last mile of distributed computing to connect devices, mobile applications and browsers to backend services.

Simple service definition

Define your service using Protocol Buffers, a powerful binary serialization toolset and language

Works across languages and platforms

Automatically generate idiomatic client and server stubs for your service in a variety of languages and platforms

Start quickly and scale

Install runtime and dev environments with a single line and also scale to millions of RPCs per second with the framework

Bi-directional streaming and integrated auth

Bi-directional streaming and fully integrated pluggable authentication with http/2 based transport

Code and Demo

Why we load balance...

- Scale
- High Availability
- Rolling Upgrades
- Graceful Shutdowns
- Zero Downtime
- Zero Noticeable Errors
- Allows for multiple daily deployments - hassle free

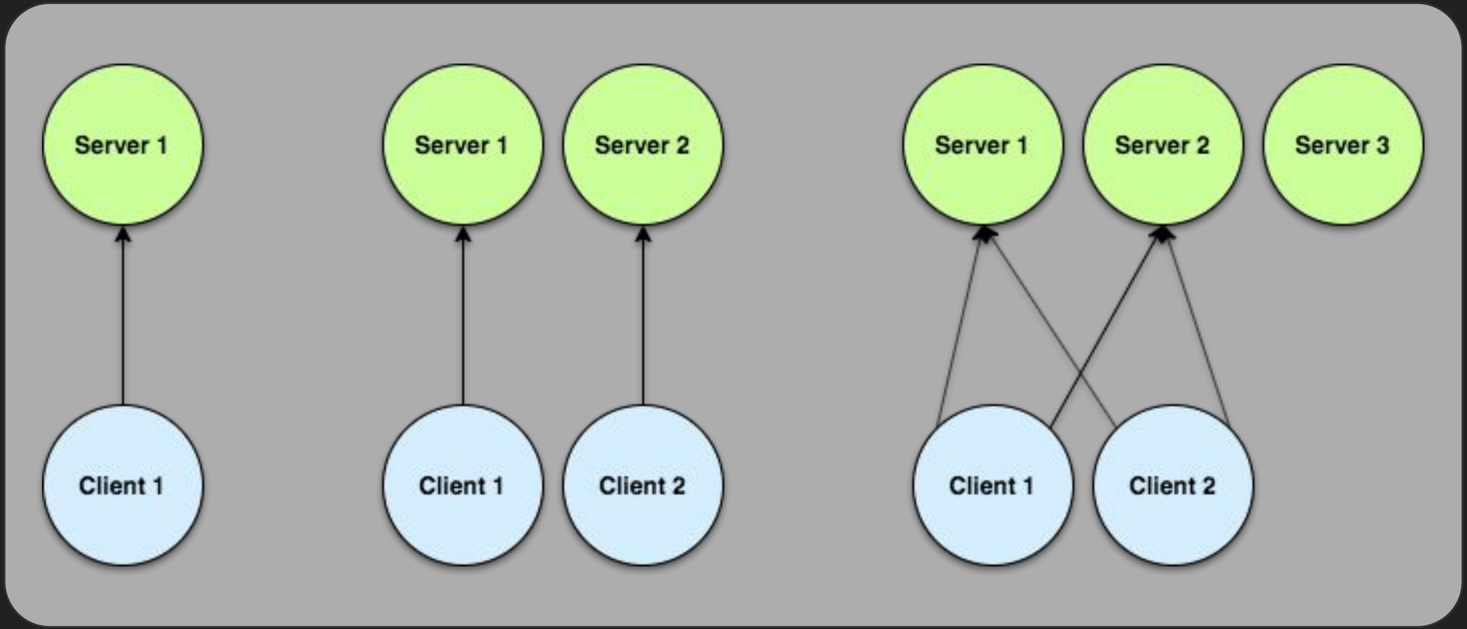
The gRPC load balancing problem (in a nutshell)

… because gRPC is built on HTTP/2, and HTTP/2 is designed to have a single long-lived TCP connection, across which all requests are multiplexed

… connection-level balancing isn’t very useful. Once the connection is established, there’s no more balancing to be done. All requests will get pinned to a single destination pod …

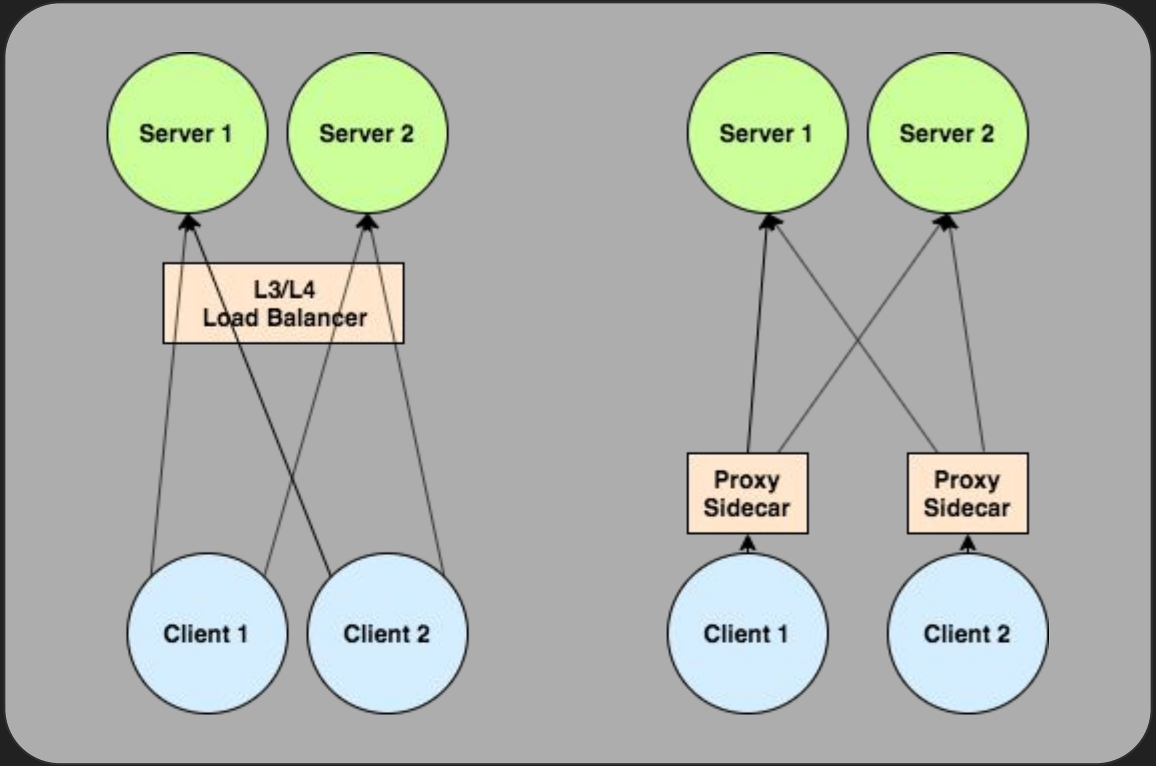
From: gRPC Load Balancing on Kubernetes without Tears

Basic Load Balancing



Basic Load Balancing Demo

Load Balancing in the Cloud



Links

- https://grpc.io/
- https://developers.google.com/protocol-buffers/docs/proto3
  - https://developers.google.com/protocol-buffers/docs/proto3#updating
- https://kubernetes.io/blog/2018/11/07/grpc-load-balancing-on-kubernetes-without-tears/
- https://itnext.io/on-grpc-load-balancing-683257c5b7b3
-
- https://github.com/ellulpatrick/grpc-demo

Thank You