

Overview



Understand gRPC

Find where it fits into your applications

Learn by doing

You'll see it all

What You'll Learn



gRPC in ASP.NET Core

Protocol Buffers

Writing Servers

Writing Clients

Using gRPC/Web

Securing gRPC

Using gRPC in ASP.NET Core

Version Check

Version Check



This version was created by using:

- .NET
- Visual Studio 2022

Version Check



This course is 100% applicable to:

- gRPC in ASP.NET Core
- Visual Studio 2022,
- Visual Studio Code, or
- Visual Studio for Mac

Not Applicable



This course is NOT applicable to:

- ASP.NET Core 2.1 or earlier
- May be compatible with ASP.NET Core 3.0, 3.1 and 5.0 but code examples will differ.

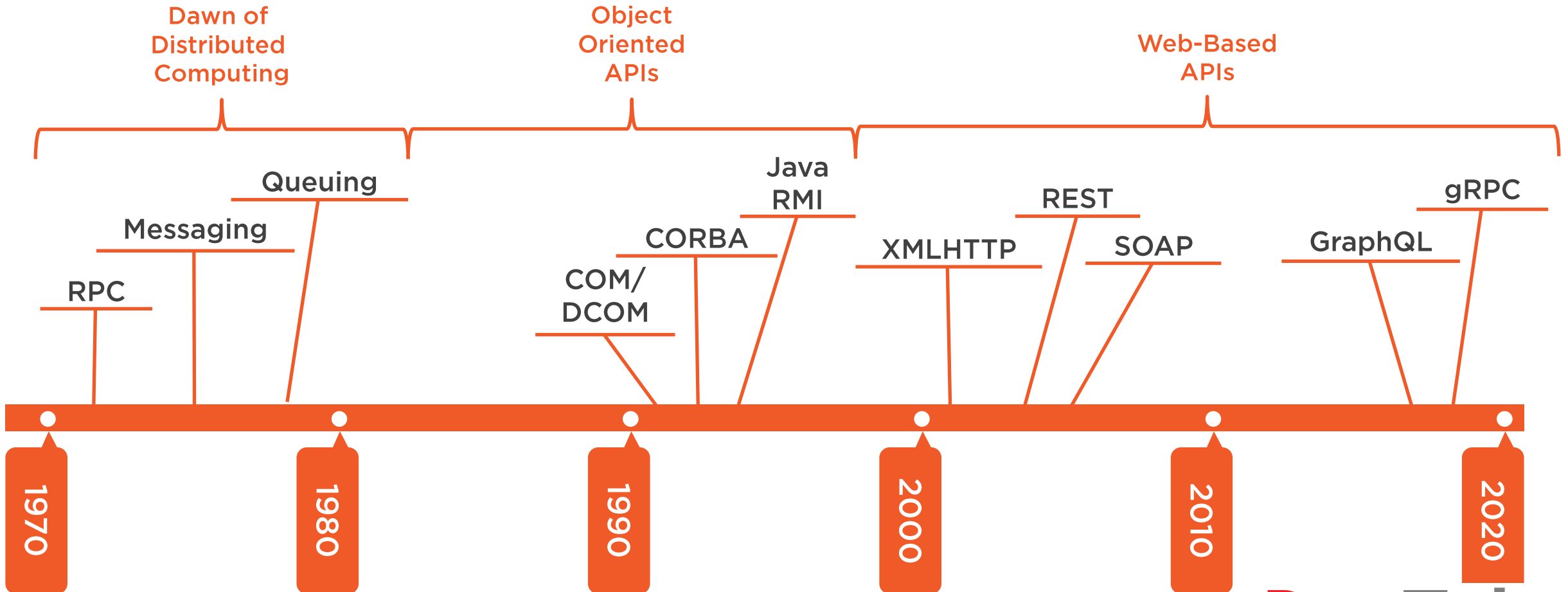
Relevant Notes



A note on frameworks and libraries:

- gRPC is based on Protocol Buffers v3

The History of Distributed APIs



gRPC:

A high performance, open-source
universal RPC framework

grpc.io



What Is gRPC?

- Binary communication
- Contract-based
- Available across ecosystems
- Secure by default
- Uni and Bi directional streaming

Does this mean REST is
dead?

No.

Is gRPC for the Web?

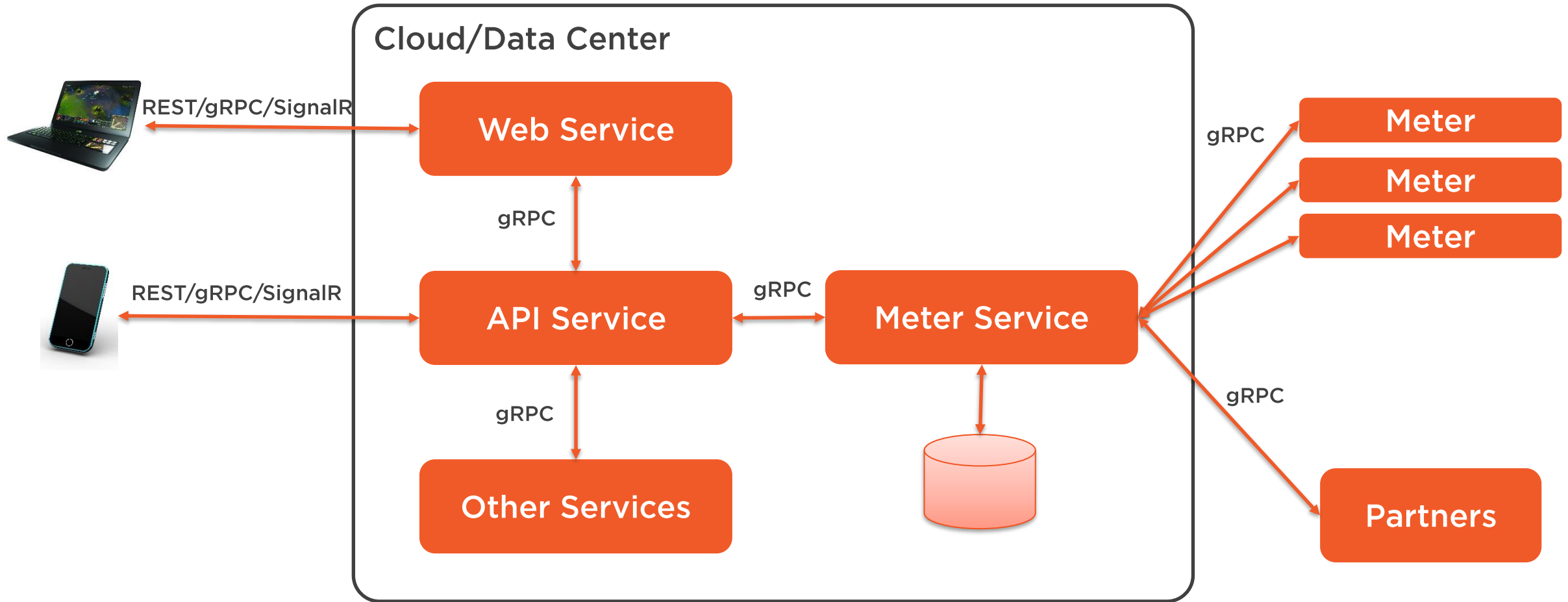
gRPC is Awesome

gRPC requires HTTP/2
Format is ordinal binary
Contracts are required

...but

Most browsers don't...yet
Harder for JS to parse
Aren't great for web apps

Where Does gRPC Belong?





Stop Thinking of Winners

- REST
- gRPC
- SignalR
- GraphQL
- Etc.

Matching Communication to Technology



REST: CRUD and purely web apps



SignalR: Multicasting and web messaging



GraphQL: Open querying of large datasets



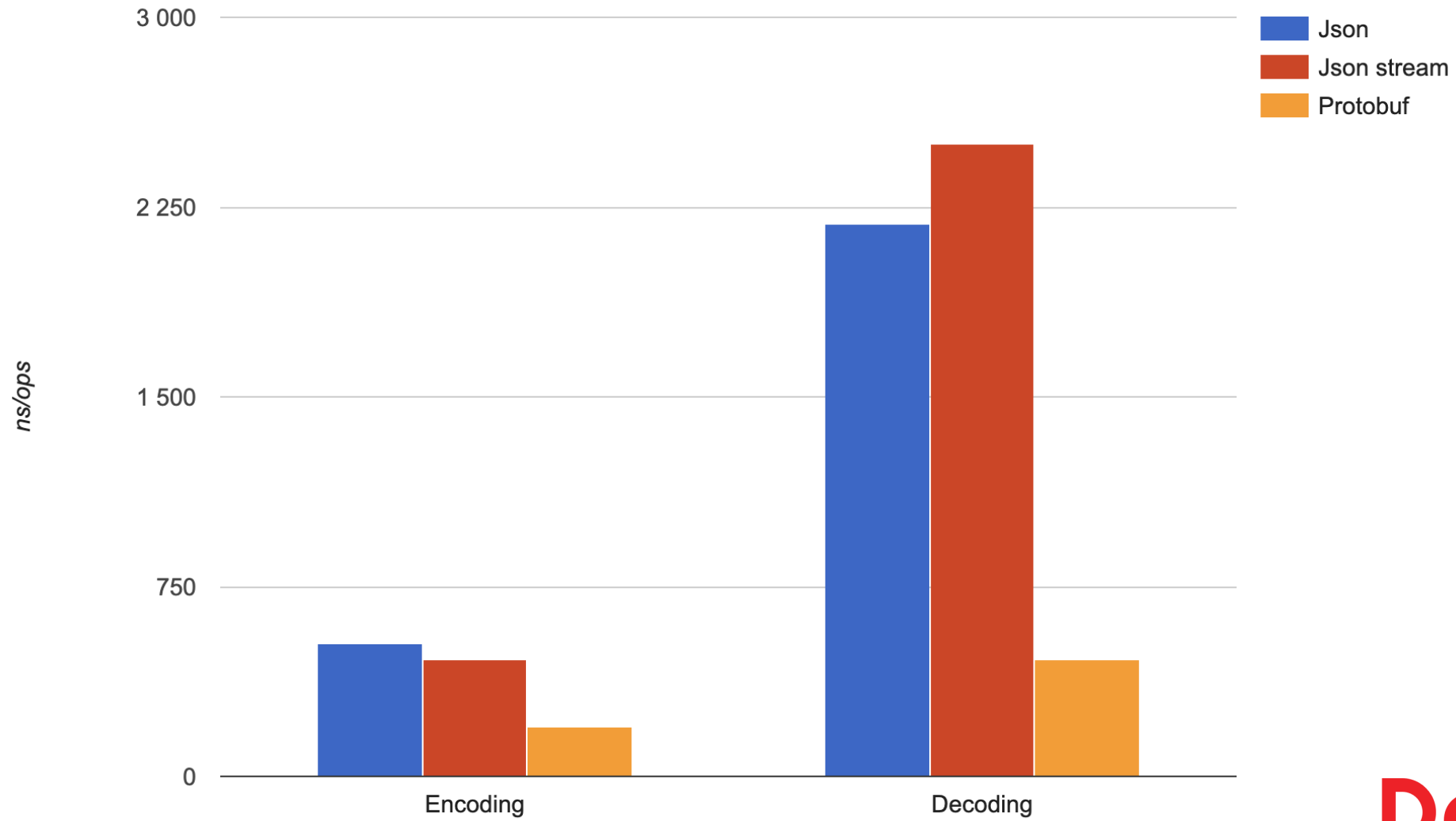
gRPC: Streaming, low resource clients, and inter-data center

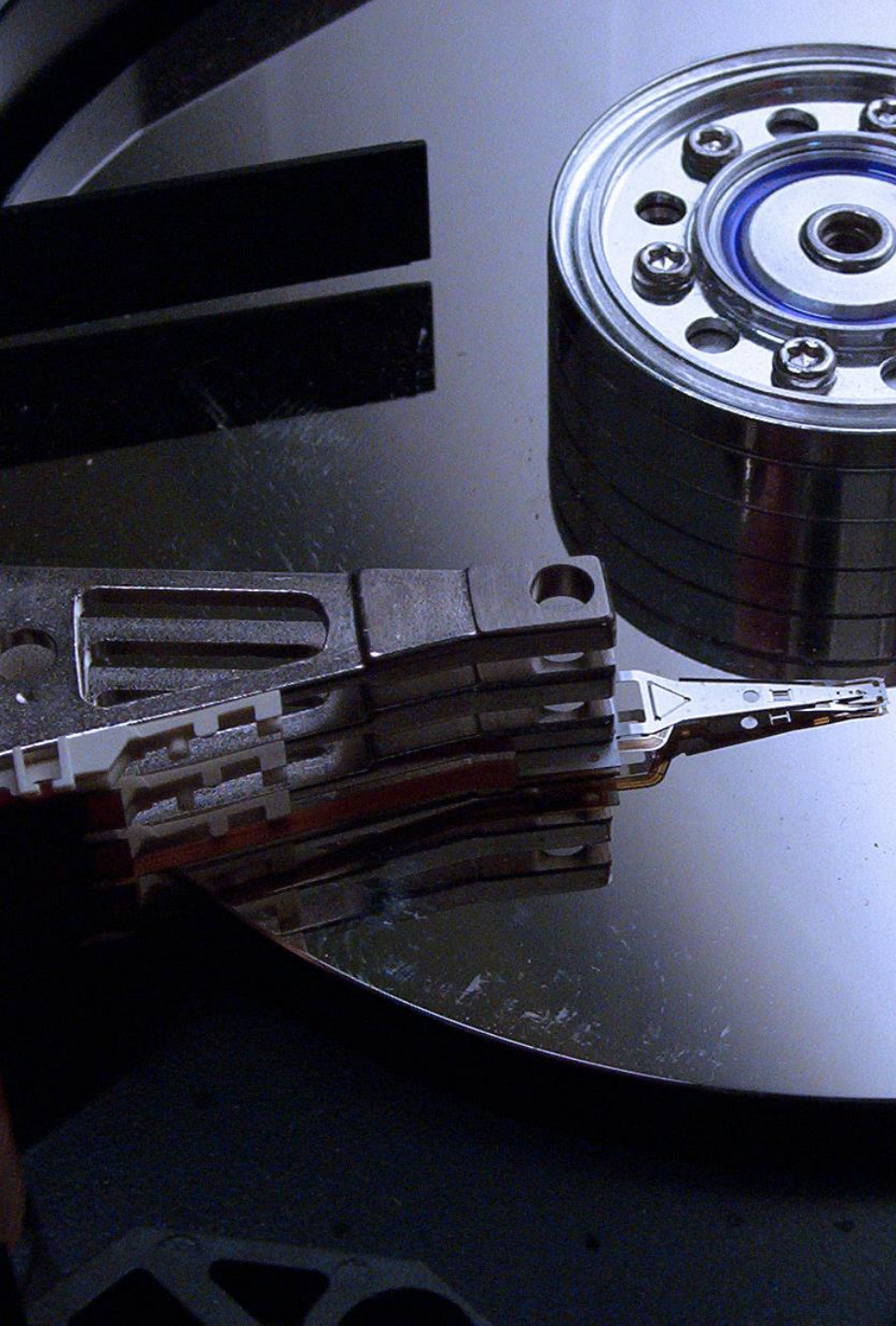


Contracts

- Generated Clients/Servers
 - Uses interface definition language
 - Protocol Buffers (ProtoBuf)
 - Language Independent
 - C++
 - Java
 - Obj-C
 - Python
 - Ruby
 - Go
 - NodeJS

“Is It Really Faster?”





Protocol Buffers

- Interface Definition Language
- Language-neutral
- Platform-neutral
- Extensible
- Serializable
- Not specific to gRPC

Pseudo Serialized ProtoBuf

Request

```
1: 100  
2: "Hello There"
```

Response

```
1: 94.5  
2: 15.4  
3: .5
```

What We've Learned



gRPC is a new, higher performance way to build APIs



Protocol Buffers is an interface language to build our contracts



gRPC isn't going to replace REST, but fill other service requirements

Coming Up Next:

Adding gRPC to ASP.NET Core