



# REST vs gRPC

## gRPC

### ► Why:

- Alta velocidad y eficiencia (Protobuf)
- Multilenguaje (C++, Java, Python, Go, Ruby, Objective-C)
- Compatibilidad con múltiples patrones de llamada unidireccionales y bidireccionales.

### ► When:

- gRPC es más adecuado para aplicaciones que requieren una alta velocidad y eficiencia.
- Aplicaciones IoT y Streaming de video.

### ► How?



# REST vs gRPC

## ■ Why:

- Estándares abiertos: HTTP y JSON (fácil integración)
- Amplia compatibilidad con plataformas y lenguajes.
- Fácil de entender
- Escalabilidad

## ■ When:

- Cuando se necesita una aplicación sin gran complejidad y sin necesidad de una velocidad de respuesta rápida.
- Aplicaciones que pueden requerir escalabilidad.