

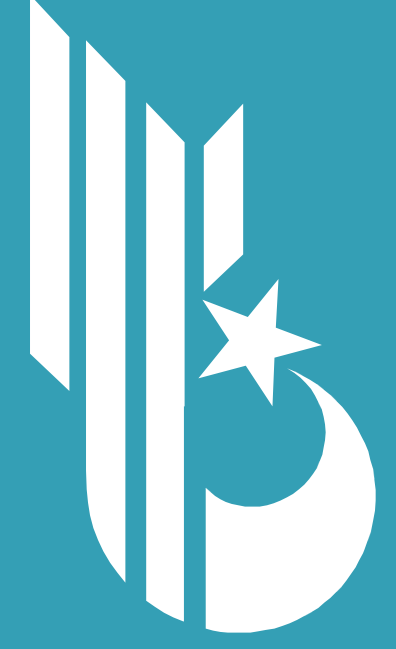


BTK
AKADEMİ

Web Servislerine

Giriş

Zafer CÖMERT



BTK
AKADEMİ

Bölüm 8

gRPC

Web Servislerine Giriş



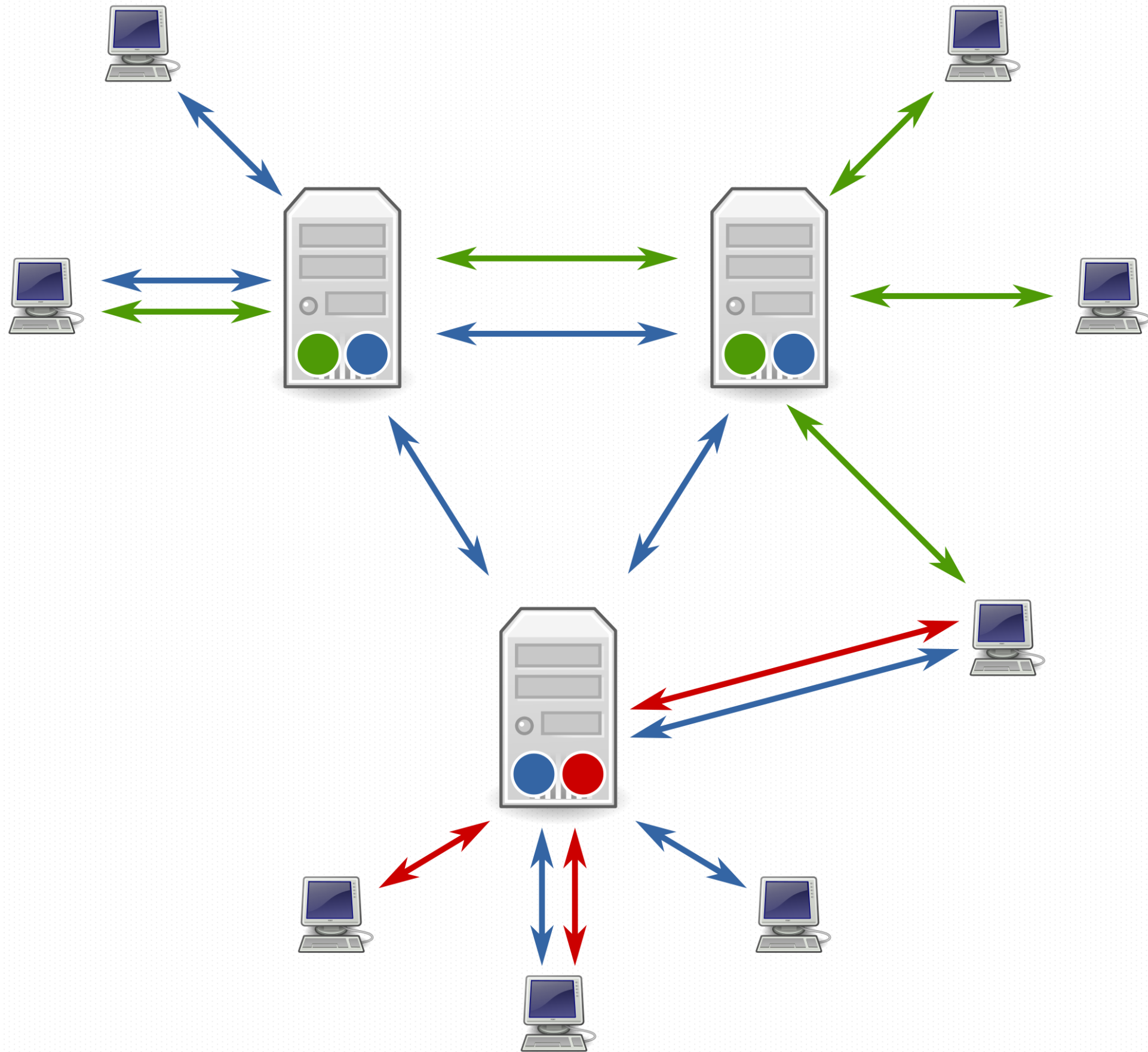
gRPC

1. gRPC
2. Servis Tanımı
3. gRPC Avantajları
4. gRPC Dezavantajları
5. gRPC'de API Türleri
6. gRPC ve REST
6. gRPC ve Diğer Protokoller
7. Gerçek Dünyada gRPC Örnekleri
8. gRPC Uygulaması

gRPC

A high performance, open source universal RPC framework

Süreçler Arası İletişim



Pipe

Socket

Shared memory

Message Queues

Semaphore

Signal

gRPC



Yüksek performanslı açık kaynaklı evrensel bir RPC çerçevesidir.

"Stubby" (Google'ın dahili RPC sistemi) tabanlıdır.

Cloud Native Computing Foundation (CNCF) bir parçasıdır.

gRPC bu kadar verimli yapan nedir?

So what makes gRPC so effective?

Interface Defination Language

API tanımı ve yük mesajlarının yapısı.

gRPC

```
1  syntax = "proto3";
2  package greet;
3
4  option go_package = "btkakademi";
5
6  message GreetRequest {
7      |   string first_name = 1;
8      |   }
9
10     message GreetResponse {
11         |   string result = 1;
12         |   }
13
14     service GreetService {
15         |   rpc Greet(GreetRequest) returns (GreetResponse);
16         |   }
```

gRPC

Protocol Buffers

Protokol tamponları, Google'ın dil bağımsız, platform bağımsız, genişletilebilir yapıdaki veri yapılarını serileştirmek için kullanılan mekanizmasıdır.

Güçlü bir şekilde yazılmış

Geriye dönük uyumlu değişiklikler yapmak için kurallar

Ağ iletimi için verimli ikili (binary) veri gösterimi

Kapsamlı stil rehberi

Yenilikler

Binary protocol

HTTP/2, verileri ikili formatta iletişim kurar. Bu, verilerin metin tabanlı değil, ikili olarak kodlandığı anlamına gelir. Bu, daha hızlı ve daha verimli bir iletişim sağlar, çünkü metin tabanlı verilerin aksine ikili veriler daha sıkıştırılabilir ve işlenebilir.

Native Stream Support

HTTP/2, iletişim sırasında birden fazla veri akışını aynı bağlantı üzerinden destekler. Bu, özellikle web sayfalarını daha hızlı yüklemek için birden fazla kaynağın paralel olarak alınabilmesini sağlar. Bu, WebSocket gibi ek protokollere gerek kalmadan veri akışını destekler.

Stream Multiplexing

HTTP/2, tek bir bağlantı üzerinden birden fazla veri akışını aynı anda destekler. Bu, farklı kaynaklardan gelen verilerin aynı anda iletilmesine izin verir ve verimliliği artırır. Her veri akışı bağımsız olarak yönetilir ve bir akışın beklemesi veya tıkanması, diğer akışları etkilemez.

Header Compression

HTTP/2, başlıkların sıkıştırılmasını kullanarak daha az bant genişliği kullanır. Bu, özellikle talep ve yanıtlar arasındaki başlık verileri gibi tekrarlanan verilerin iletilmesini daha verimli hale getirir. Bu, daha hızlı veri aktarımını destekler.

gRPC

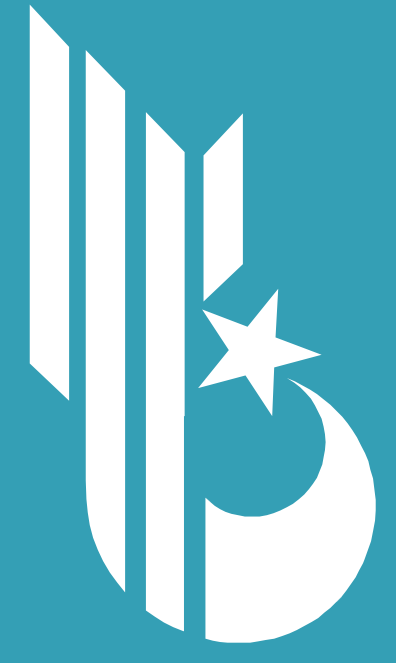


gRPC

gRPC Remote Procedure Call framework

HTTP/2

Protobuf3



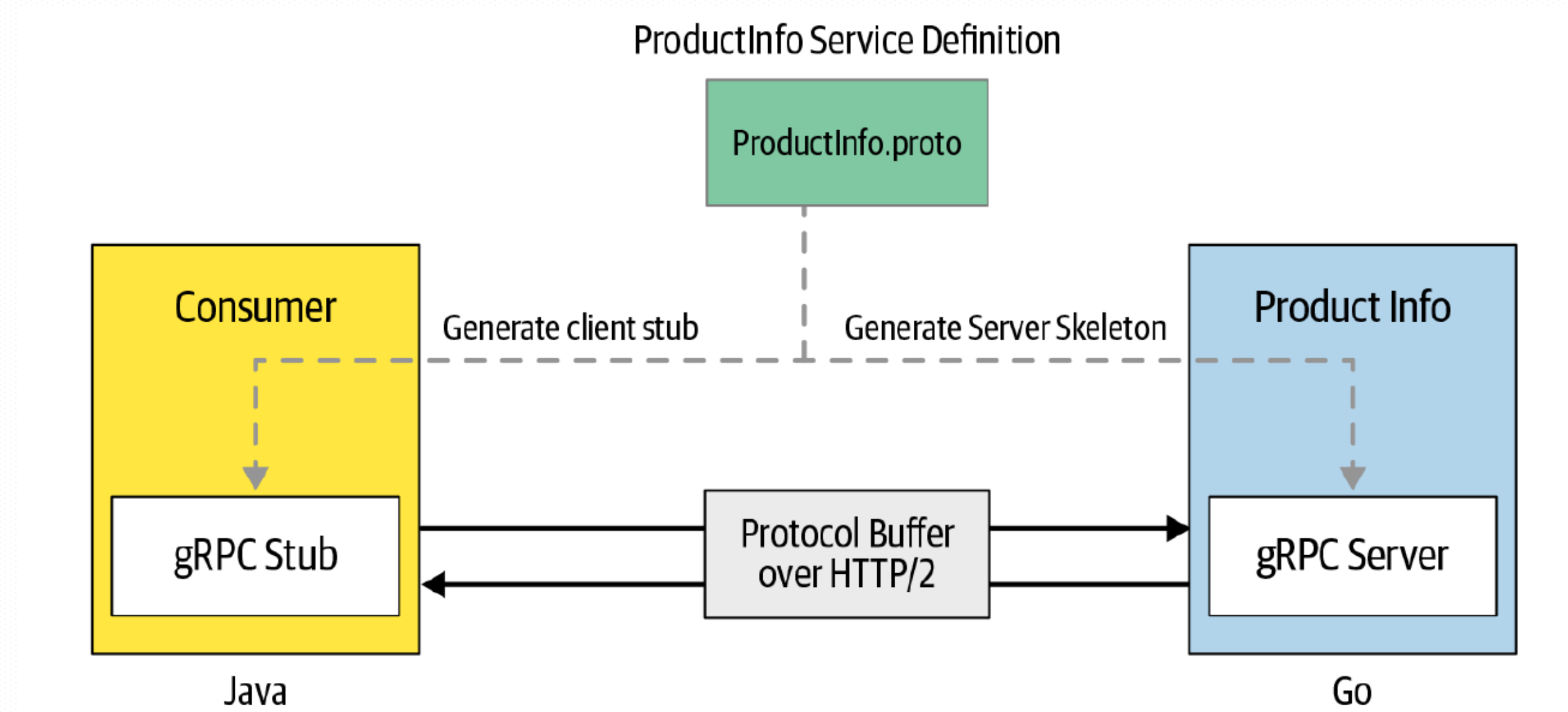
BTK
AKADEMİ

Bir kez tanımlandığında, desteklenen tüm diller için iyi yapılandırılmış kod üretir!

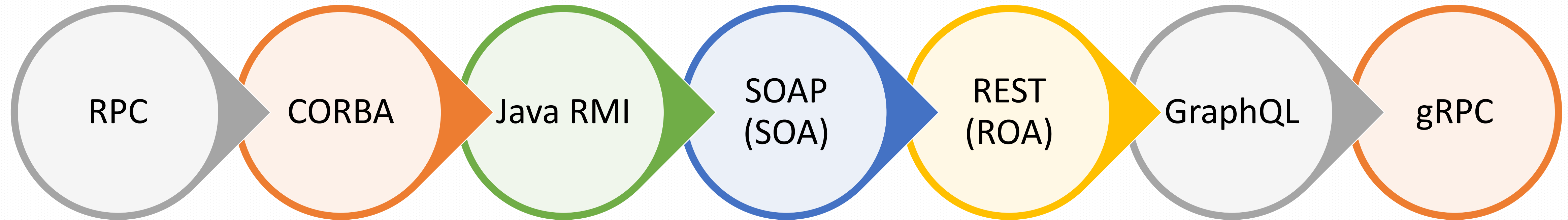
Neden gRPC?

Why gRPC?

gRPC



gRPC



gRPC'nin Avantajları

Süreçler arası iletişim için verimli

Basit iyi tanımlı servis arayüzleri ve şema desteği sağlar.

Güç tipleme

Polyglot

İkili akış desteği

Dahili standart özellikler

Kimlik
doğrulama

Şifreleme

Meta veri
alış verişi

Yük
dengeleme

Hizmet
keşfi

Bulut Doğal Ekosistemiyle Entegre

CNCF bir parçası

Prometheus

Olgun ve Geniş Ölçekte Benimsenmiş

Netflix

Docker

Cisco

gRPC dezavantajları nedir?

gRPC

Dışarıya
Yönlendirilen
Hizmetler için
Uygun Olmayabilir

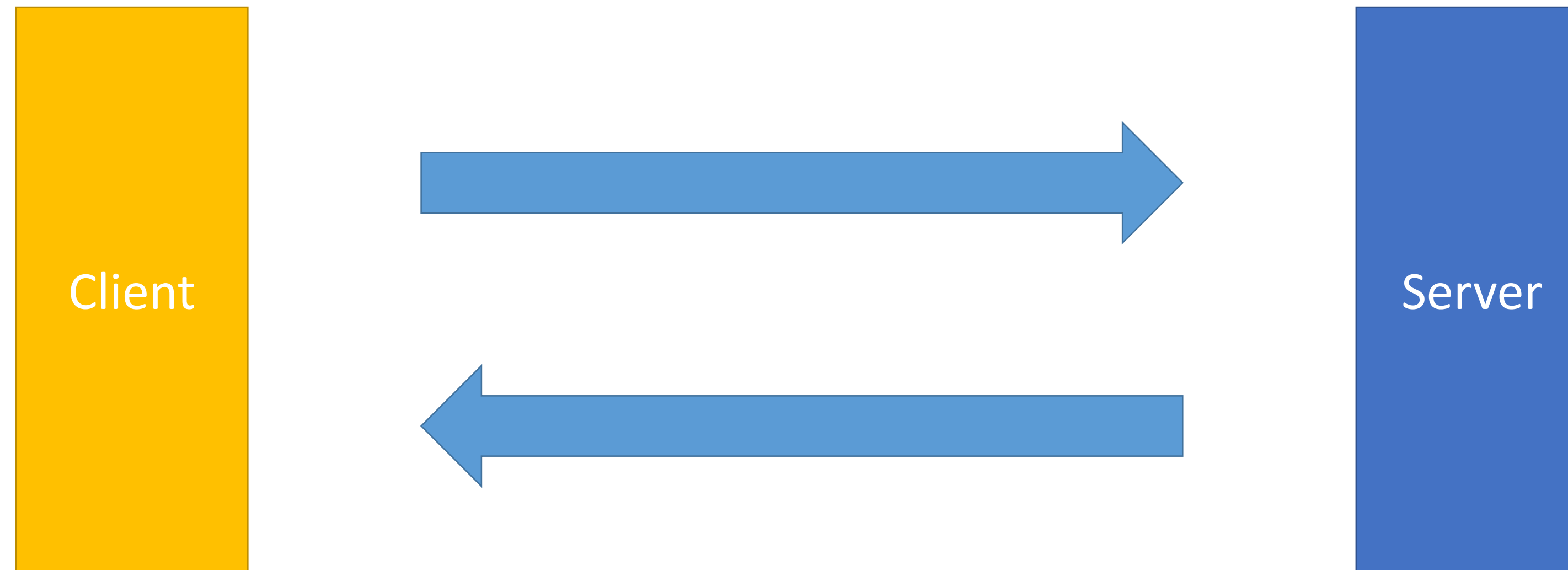
Hizmet Tanımı
Değişiklikleri
Karmaşık Bir
Geliştirme Sürecidir

Ekosistem Görece
Küçüktür

gRPC'deki API Türleri

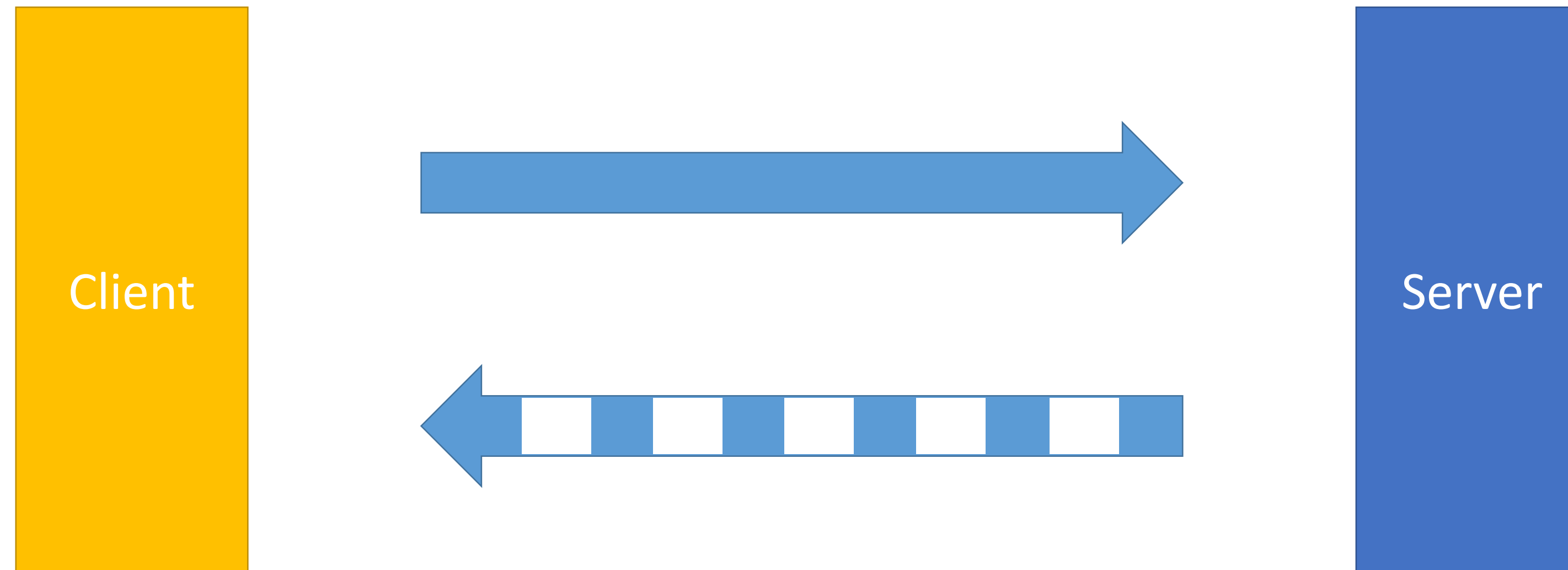
Types of APIs in gRPC

Unary



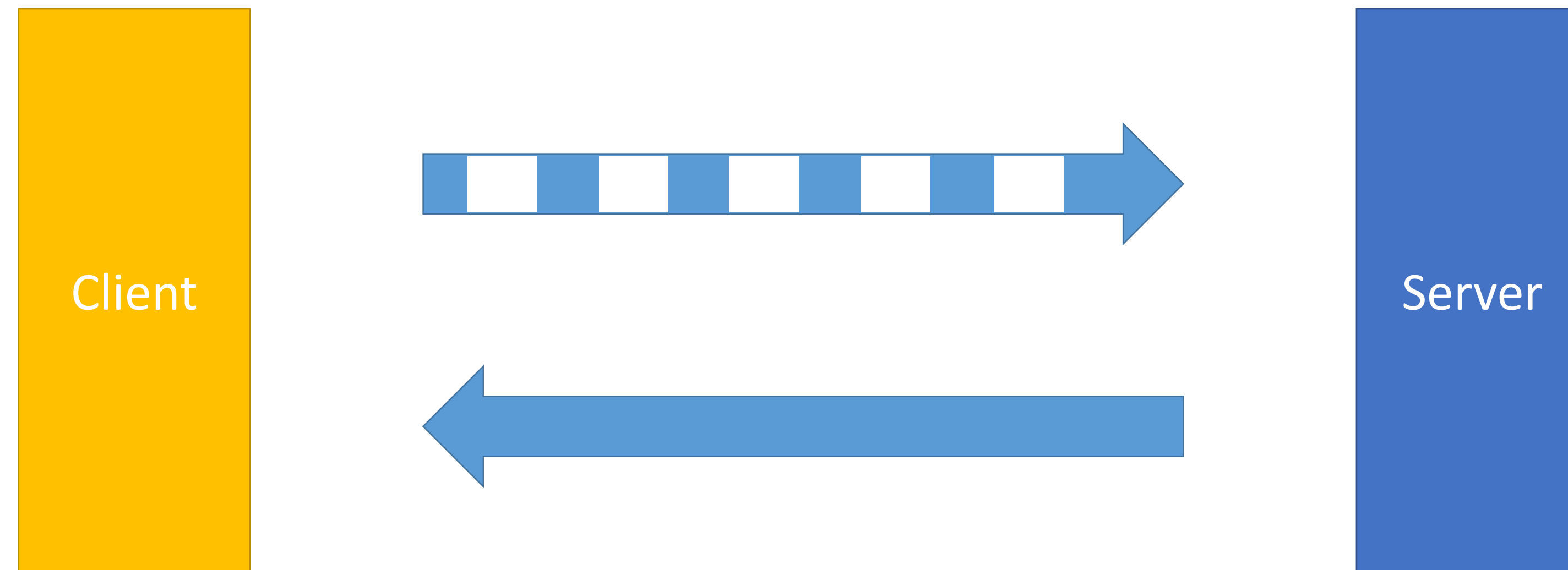
- İstemci tek bir istek gönderir ve tek bir yanıt alır.

Server Streaming



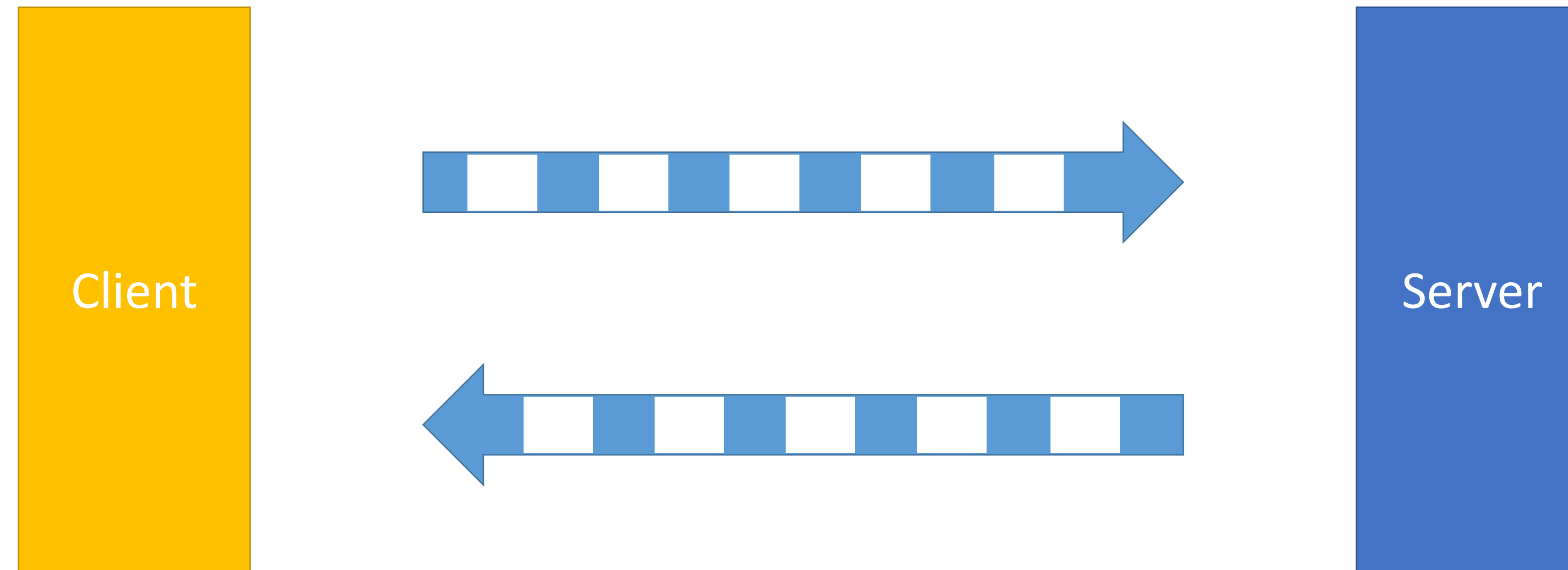
- Sunucu, istemcinin istek mesajını aldıktan sonra bir yanıt akışı gönderir.

Client Streaming



- İstemci, sunucuya tek bir istek yerine bir istek akışı gönderir.

Bidirectional Streaming



- Çağrı, istemcinin yöntemi çağırması ve sunucunun istemci meta verilerini, yöntem adını ve son teslim tarihini almasıyla başlatılır.

gRPC'deki API Türleri

Unary RPC

- İstemci tek bir istek gönderir ve tek bir yanıt alır.

Server streaming RPC

- Sunucu, istemcinin istek mesajını aldıktan sonra bir yanıt akışı gönderir.

Client streaming RPC

- İstemci, sunucuya tek bir istek yerine bir istek akışı gönderir.

Bidirectional streaming RPC

- Çağrı, istemcinin yöntemi çağırması ve sunucunun istemci meta verilerini, yöntem adını ve son teslim tarihini almasıyla başlatılır.

gRPC, REST API, GraphQL

gRPC vs REST

Karakteristik	gRPC	REST API
HTTP Protokol	HTTP/2	HTTP/1.1
Messaging Format	Protobuf (Protocol buffers)	JSON, XML ve diğerleri...
Code Generation	Native Protoc Compiler	Third-Party Solutions Like Swagger
Communication	Unary Client Request Bidirectional Streaming	Client-Request Only

gRPC vs REST

	Goal	REST (HTTP/JSON)	gRPC
COMPATIBILITY	Single source of truth	✗	✓
	Multi-platform + languages built in	✗	✓
	Handle non-breaking changes.	✗	✓
PERFORMANCE	Network: connection handling	Manual, 1 per call ✗	Built-in, Multi per conn ✓
	Speed: Transmission of data	Human-readable Text ✗	Binary ✓
	CPU: Improved resource usage	✗	✓
MAINTENANCE	Tracing	Manual ✗	Easy to plug in ✓
	Logging	Manual ✗	Easy to plug in ✓
	Monitoring	Manual ✗	Easy to plug in ✓

gRPC vs GraphQL

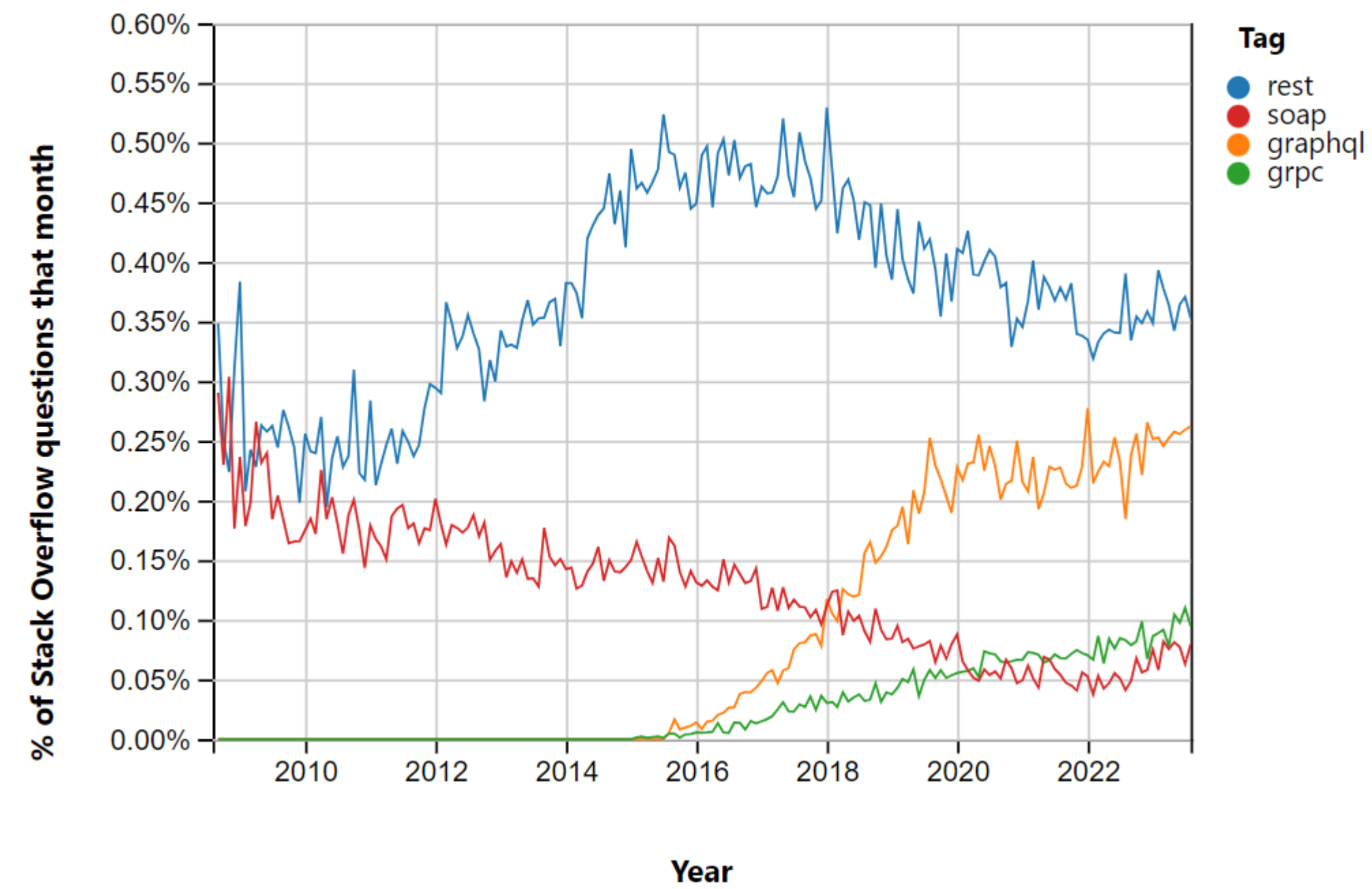
GraphQL

- Esnek veri isteme
- Öğrenmesi kolay
- Özellik tabanlı gelişim
- Karmaşıklık özellikle önbelleğe alma konusunda
- Güvenlik (birden fazla hizmet ile iletişim)

gRPC

- Performans
- Dil bağımsız
- Otomatik kod üretimi
- Katı şema tanımı
- Sıkı bağlantı
- HTTP/2 bağımlılığı

Trendler



Neler Öğrendik?

1. gRPC
2. gRPC Özellikleri
3. gRPC Avantajları
4. gRPC Dezavantajları
5. gRPC İletişim Türleri