

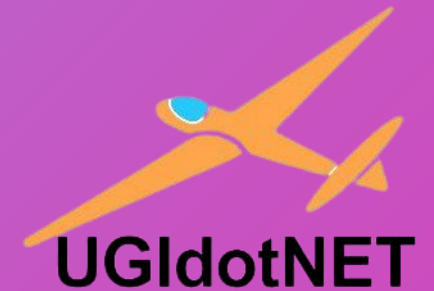
gRPC con .NET

Mauro Bennici

CTO | Harley & Dickinson (We are hiring!)

info@maurobennici.it

<https://www.linkedin.com/in/maurobennici/>



**.NET Conference
Italia 2023**



Agenda

- Cosa è gRPC
- Perché usarlo
- Come usarlo con .NET (e Python)
- Novità .NET 8

**** COMMODORE 64 BASIC V2 ****

64K RAM SYSTEM 38911 BASIC BYTES FREE

READY.

10 INPUT "YOUR NAME";X\$

20 PRINT "HI ";X\$

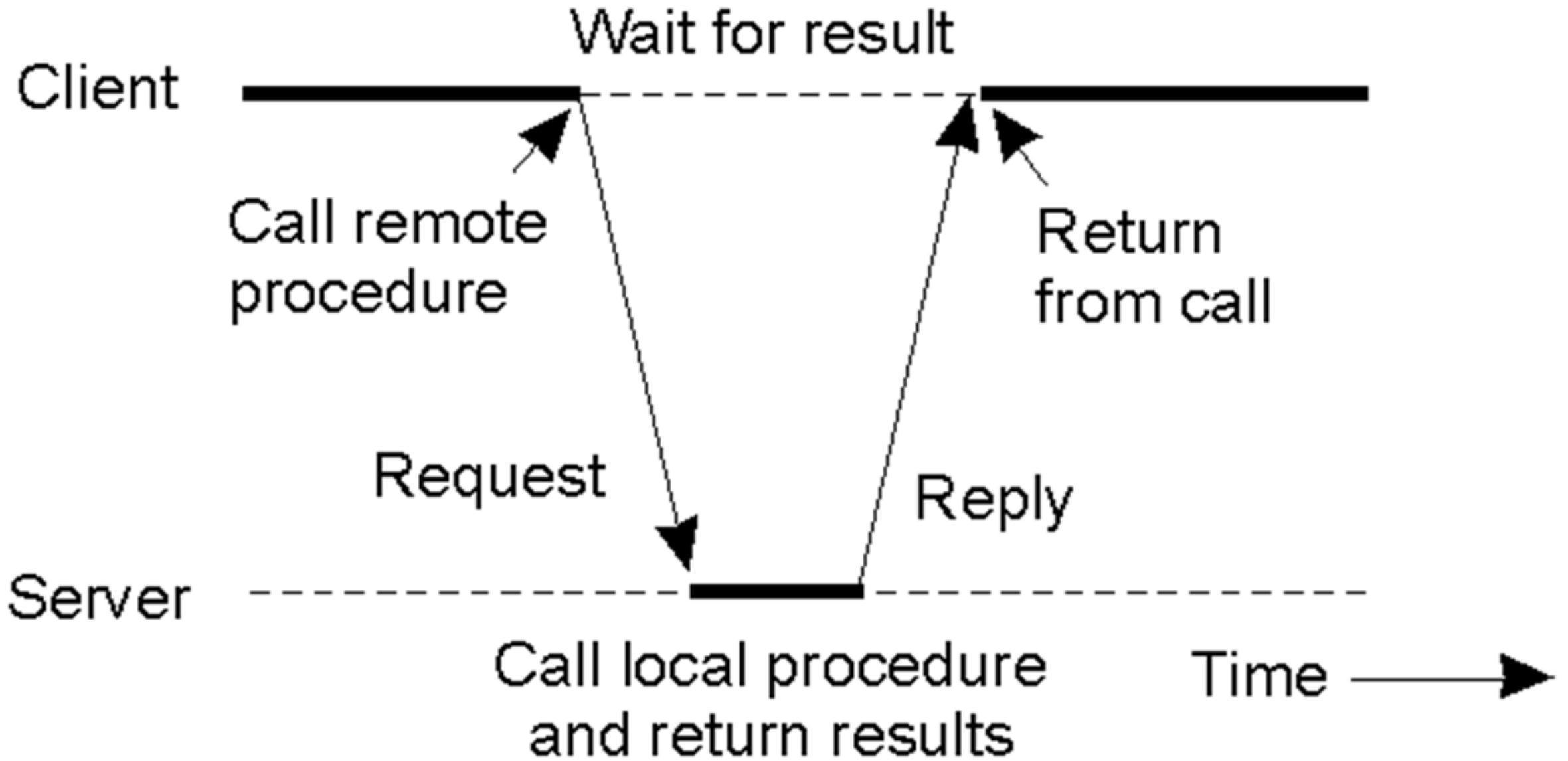
30 INPUT "HOW ARE YOU";X\$

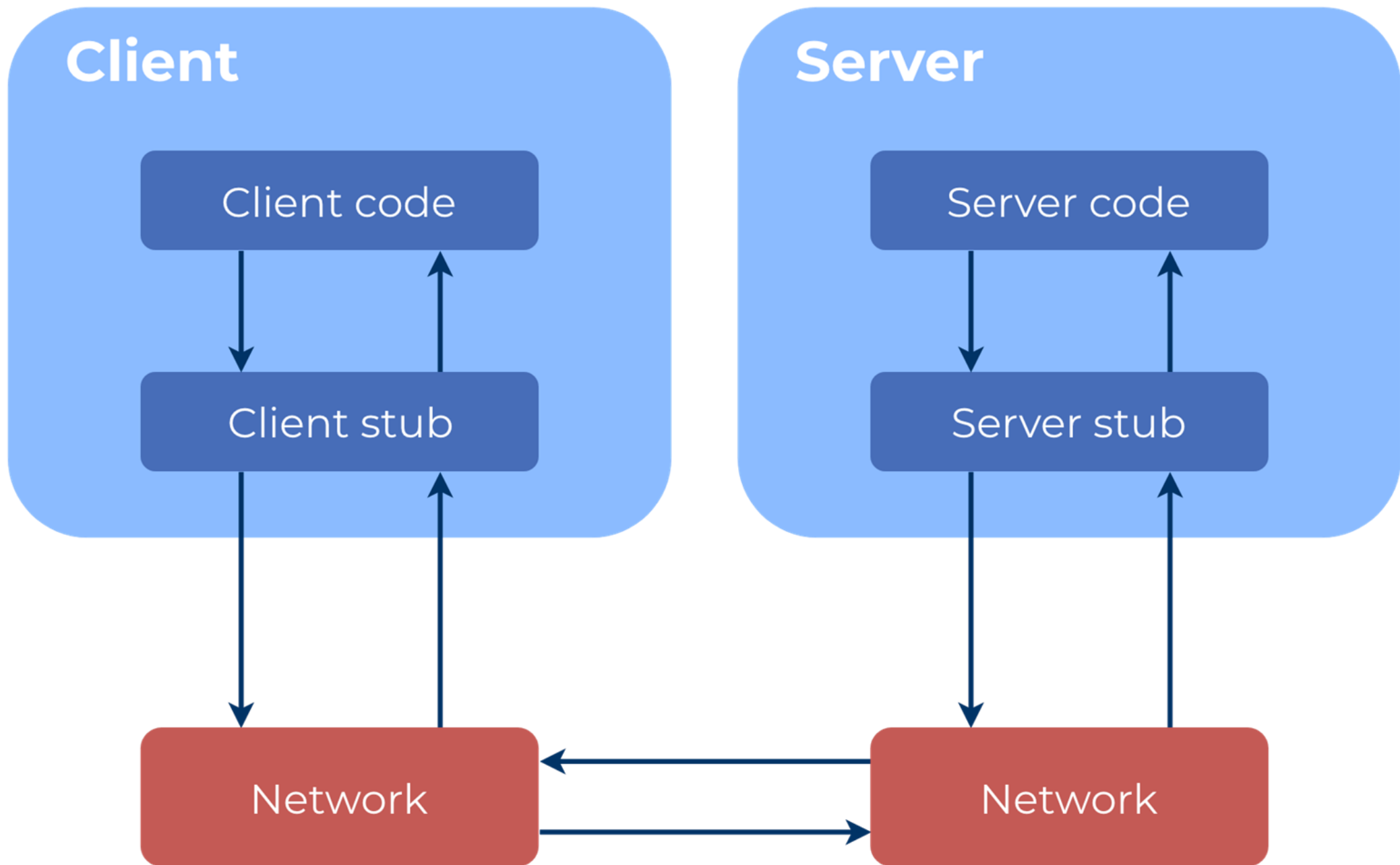
40 PRINT "NICE TO MEET YOU!"

50 PRINT "HAVE ■

gRPC

A high performance, open source
universal RPC framework





Problemi

- Piattaforma
- Compatibilità binaria
- Dati in binario
- Linguaggio di programmazione
- Sicurezza
- Latenza

Alternative (più o meno)

- SOAP
- CORBA
- (REST) API
- WebSocket

Limiti

- CORBA (enorme e complesso)
- REST API (nata NON per la velocità, non bi-direzionale)

Linguaggi supportati

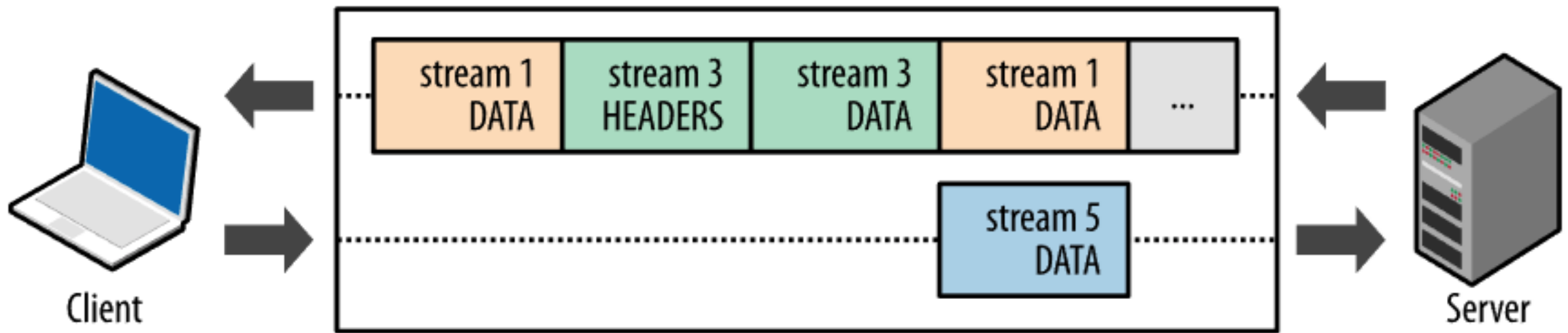
- C#
- C++
- Dart
- Go
- Java
- Kotlin
- Node
- Objective-C
- PHP
- Python
- Ruby

Protocol Buffers

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;  
}
```

HTTP 2

HTTP 2.0 connection



Demo

Python vs C#

**.NET Conference
Italia 2023**



.NET

Quando usare cosa?

gRPC vs REST: Understanding gRPC, OpenAPI and REST and when to use them in API design

<https://cloud.google.com/blog/products/api-management/understanding-grpc-openapi-and-rest-and-when-to-use-them>

gRPC vs REST

- Entrambi -> HTTP 2
- Entrambi -> ProtoBuf
- Entrambi -> SSL
- Entrambi -> Interceptor (implementazione)
- gRPC -> streaming
- gRPC -> bidirezionale
- REST -> funziona su browser (gRPC nì)
- REST -> interoperabile (gRPC nì)
- gRPC -> FAST!!!!
- gRPC -> strongly typed

gRPC e REST

Creare gateway REST API per gRPC

<https://grpc-ecosystem.github.io/grpc-gateway/>

Generare sia gRPC che REST API da file prot

<https://learn.microsoft.com/it-it/aspnet/core/grpc/json-transcoding?view=aspnetcore-8.0>

Swagger

```
// My amazing greeter service.
service Greeter {
  // Sends a greeting.
  rpc SayHello (HelloRequest) returns (HelloReply) {
    option (google.api.http) = {
      get: "/v1/greeter/{name}"
    };
  }
}

message HelloRequest {
  // Name to say hello to.
  string name = 1;
}

message HelloReply {
  // Hello reply message.
  string message = 1;
}
```

Swagger

The screenshot shows the Swagger UI interface for a gRPC transcoding example. At the top, the Swagger logo is on the left, and a dropdown menu on the right is set to "gRPC JSON transcoding example V1". The main title is "gRPC transcoding" with "v1" and "OAS3" tags. Below the title is the Swagger file path: "/swagger/v1/swagger.json".

The service is named "Greeter" with the description "My amazing greeter service." Below this, two API endpoints are listed:

- GET** /v1/greeter/{name} Sends a greeting.
- POST** /v1/greeter Sends a stream of greetings.

The "Schemas" section is expanded to show two schemas:

```
HelloReply {  
  message string  
  Hello reply message.  
}
```

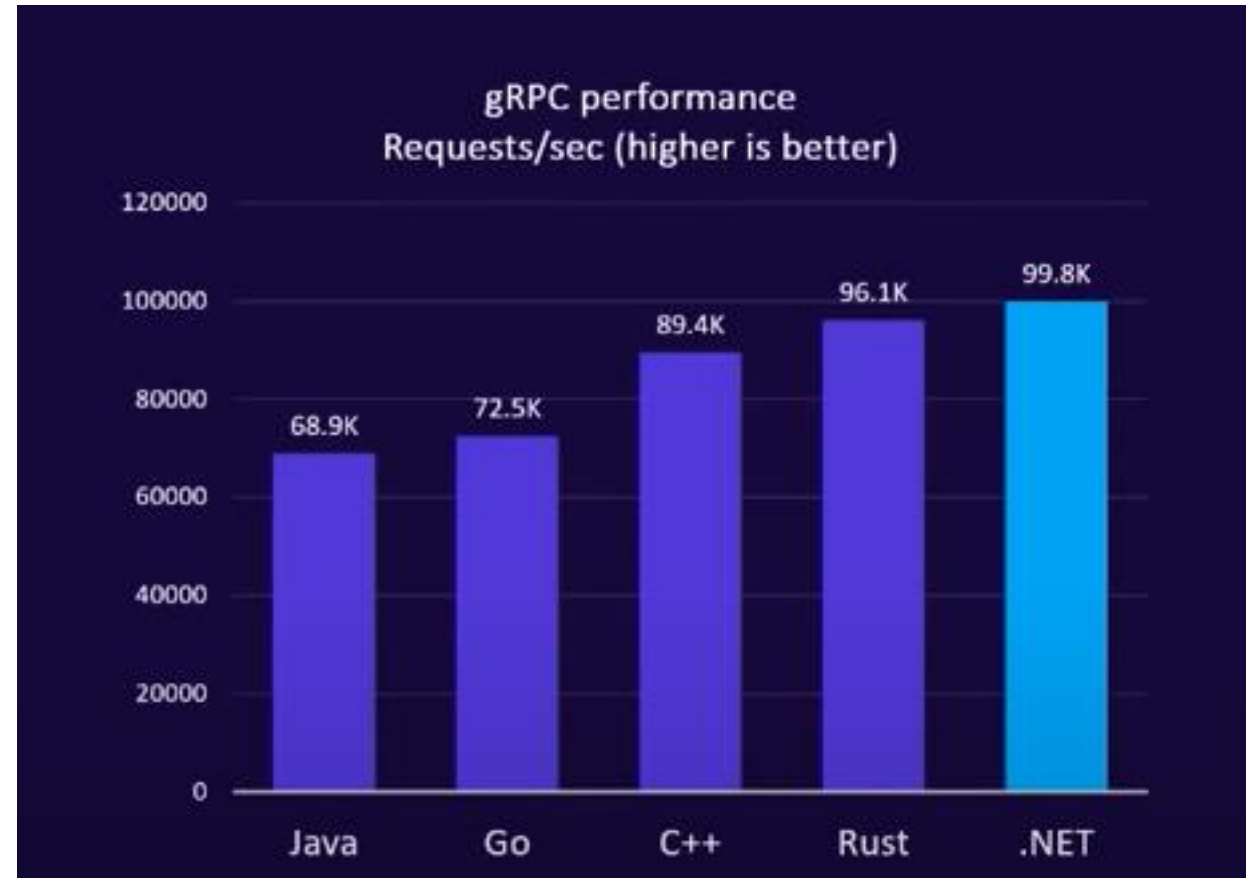
```
HelloRequestCount >
```

Benchmarks .Net 6 vs 7 vs other languages

NET gRPC team's benchmarks showed that server streaming was improved by 800%.

.NET 6 > 0.5M RPS

.NET 7 > 4.5M RPS






Performance e altro...












- <https://grpc.io/docs/guides/performance/>
- <https://learn.microsoft.com/en-us/aspnet/core/grpc/performance?view=aspnetcore-8.0>

Novità .NET 8 - Debug

grpc-dotnet 2.55.0

- ▶  ResponseAsync Id = 19, Status = RanToCompletion, Method = "{null}", Result = "{ \"message\": \"H..
- ▶  ResponseHeadersAsync Id = 60, Status = RanToCompletion, Method = "{null}", Result = "Grpc.Core.Metadata...
- ▶  Non-Public members

grpc-dotnet 2.56.0

-  IsComplete true
- ▶  Method /greet.Greeter/SayHello
- ▶  Response {{ "message": "Hello GreeterClient" }}
- ▶  ResponseHeaders  View  Count = 2
- ▶  Status {Status(StatusCode="OK", Detail="")}
- ▶  Trailers  View  Count = 0
- ▶  Raw View

Demo

Debug

**.NET Conference
Italia 2023**



.NET

Novità Azure - App Service

Topic	Answer
OS support	Currently gRPC is a Linux only feature. Support for Windows is coming in 2024 for .NET workloads.
Language support	gRPC is supported for each language that supports gRPC.
Client Certificates	HTTP/2 enabled on App Service doesn't currently support client certificates. Client certificates will need to be ignored when using gRPC.
Secure calls	gRPC must make secure HTTP calls to App Service. You cannot make insecure calls.
Activity Timeout	gRPC requests on App Service have a timeout request limit. gRPC requests will time out after 20 minutes of inactivity.
Custom Containers	HTTP/2 & gRPC support is in addition to App Service HTTP/1.1 support. Custom containers that would like to support HTTP/2 must still support HTTP/1.1.

Caso Studio

DropBox

<https://dropbox.tech/infrastructure/how-we-migrated-dropbox-from-nginx-to-envoy>

gRPC

dalle basi ad avanzato, sviluppiamo un'applicazione reale

<https://talks.codemotion.com/introduzione-11?playlist=grpc-dalle-basi-ad-avanzato-sviluppiamo->





LIVE AperiTech

20 dicembre 2023 | 18.00 - 21.30

Toolbox - Via Agostino da Montefeltro 2 - TORINO

Mauri XMAS

Libri, segreti, novità di Azure SQL DB



Ospite: Davide Mauri

Principal Product Manager
Azure SQL DB @Microsoft

EVENTO
LIVE



<https://www.linkedin.com/in/maurobennici/>



info@maurobennici.it



Slide e materiale su
<https://www.dotnetconference.it/>

**.NET Conference
Italia 2023**

