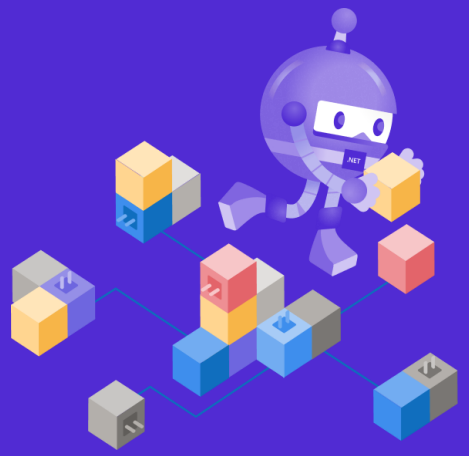
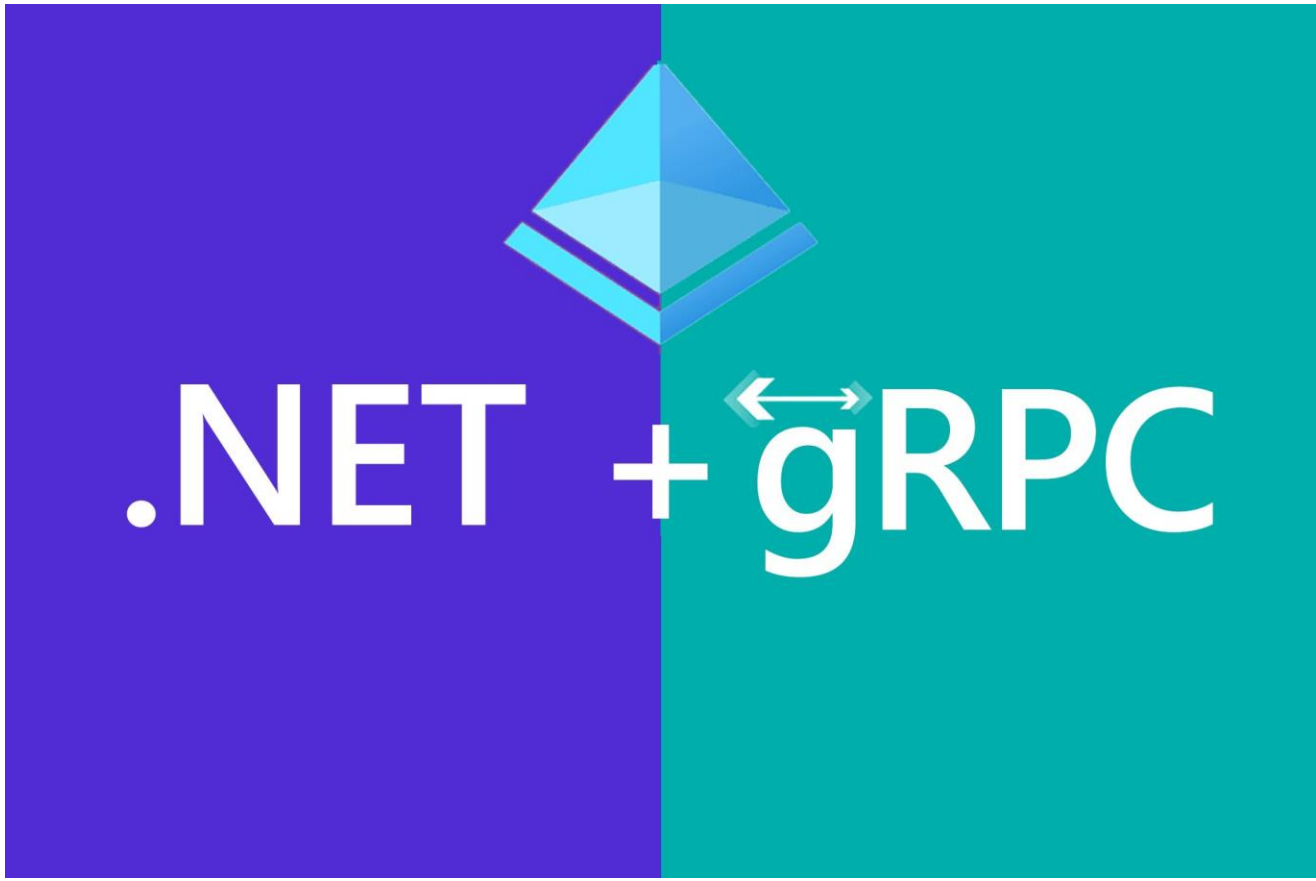
Using gRPC in Microservices Communication w/ .Net5



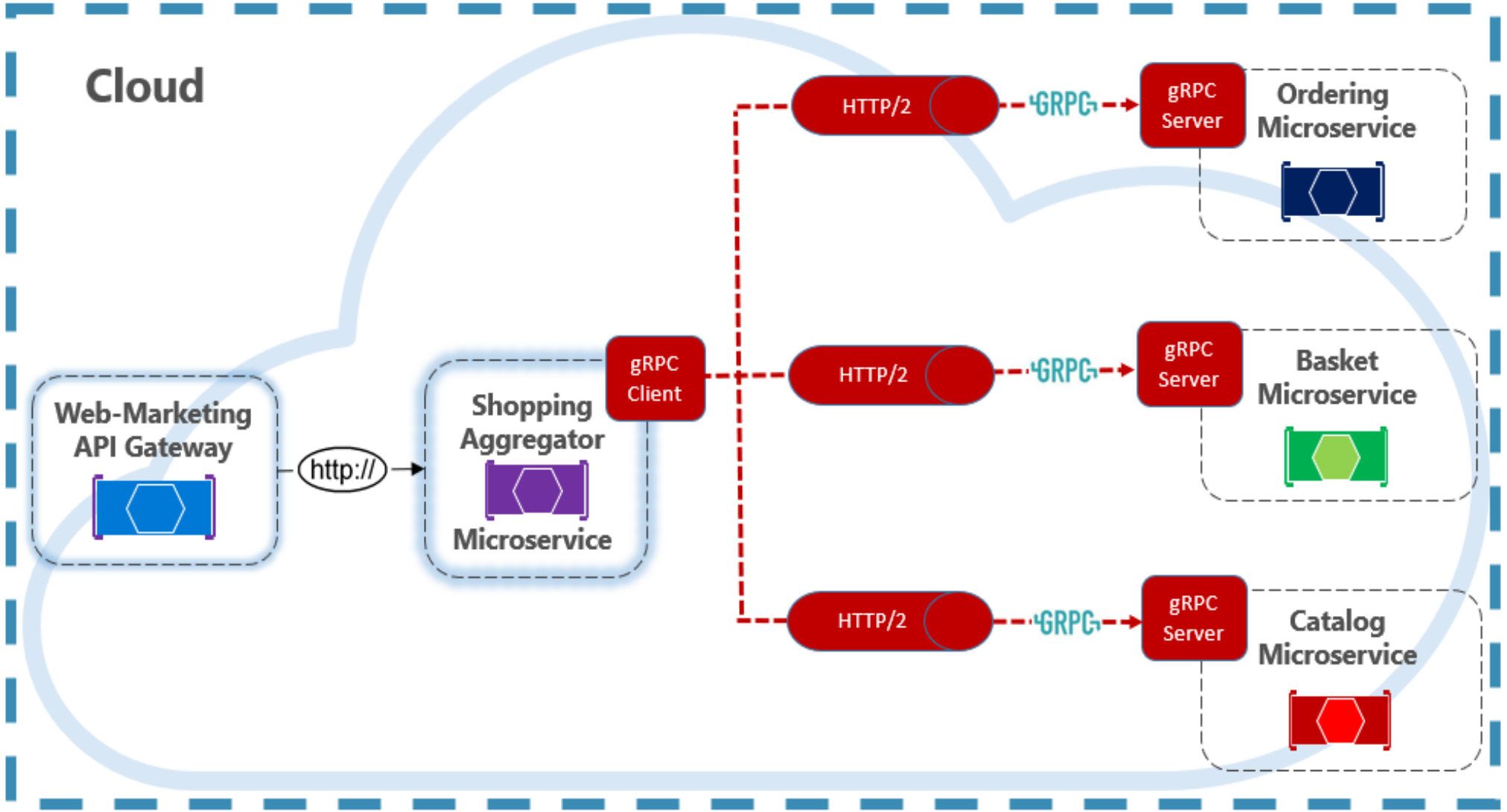
Mehmet Ozkaya
Software Architect, .NET
@ezozkme

gRPC in Microservices Communication

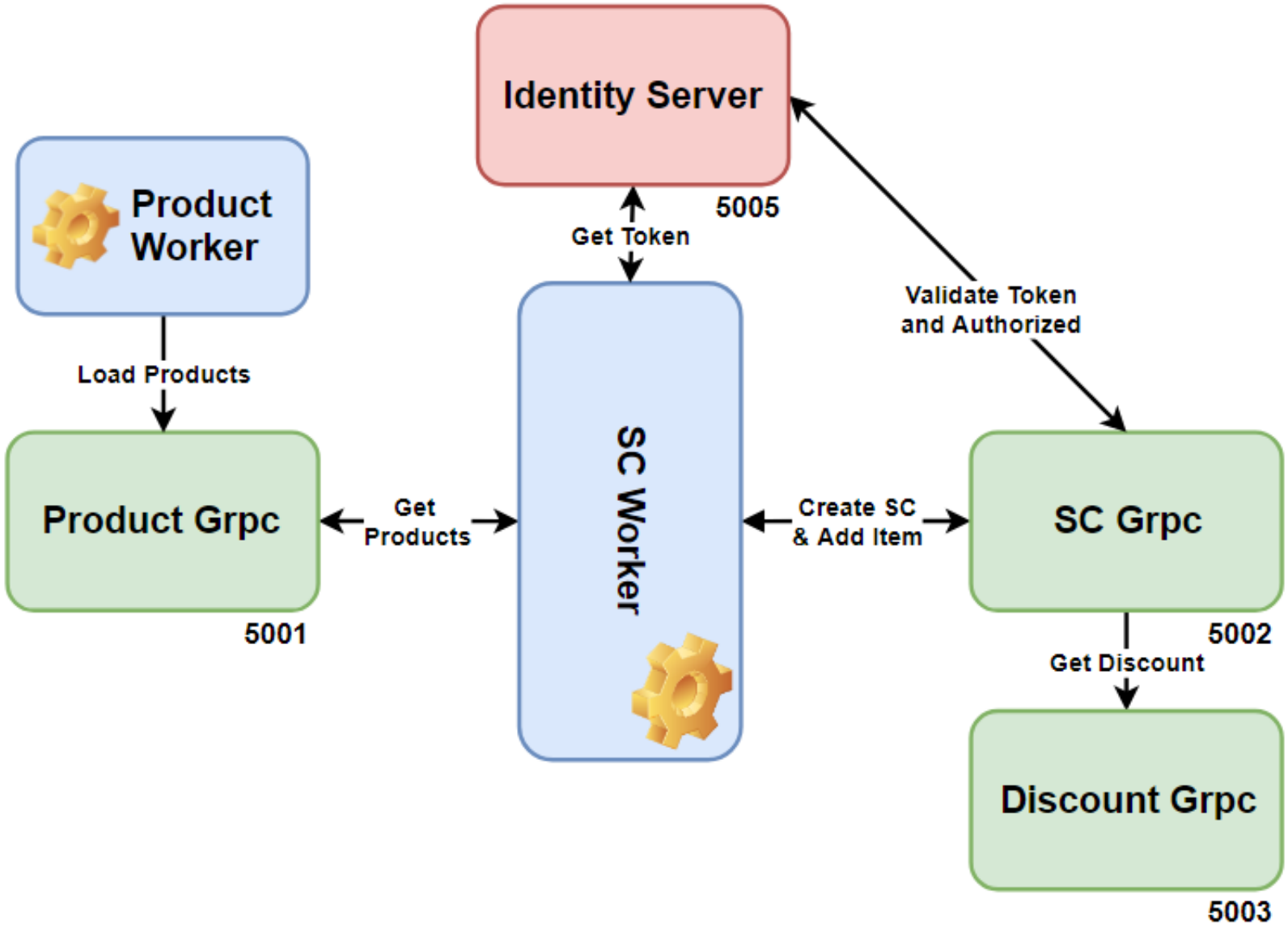
- Highly Performant Inter-service Communication
- Modern distributed system
- HTTP/2 as base transport protocol and ProtoBuf encoding
- first-class support for creating gRPC services with Asp.Net 5



gRPC usage of Microservices



Big Picture

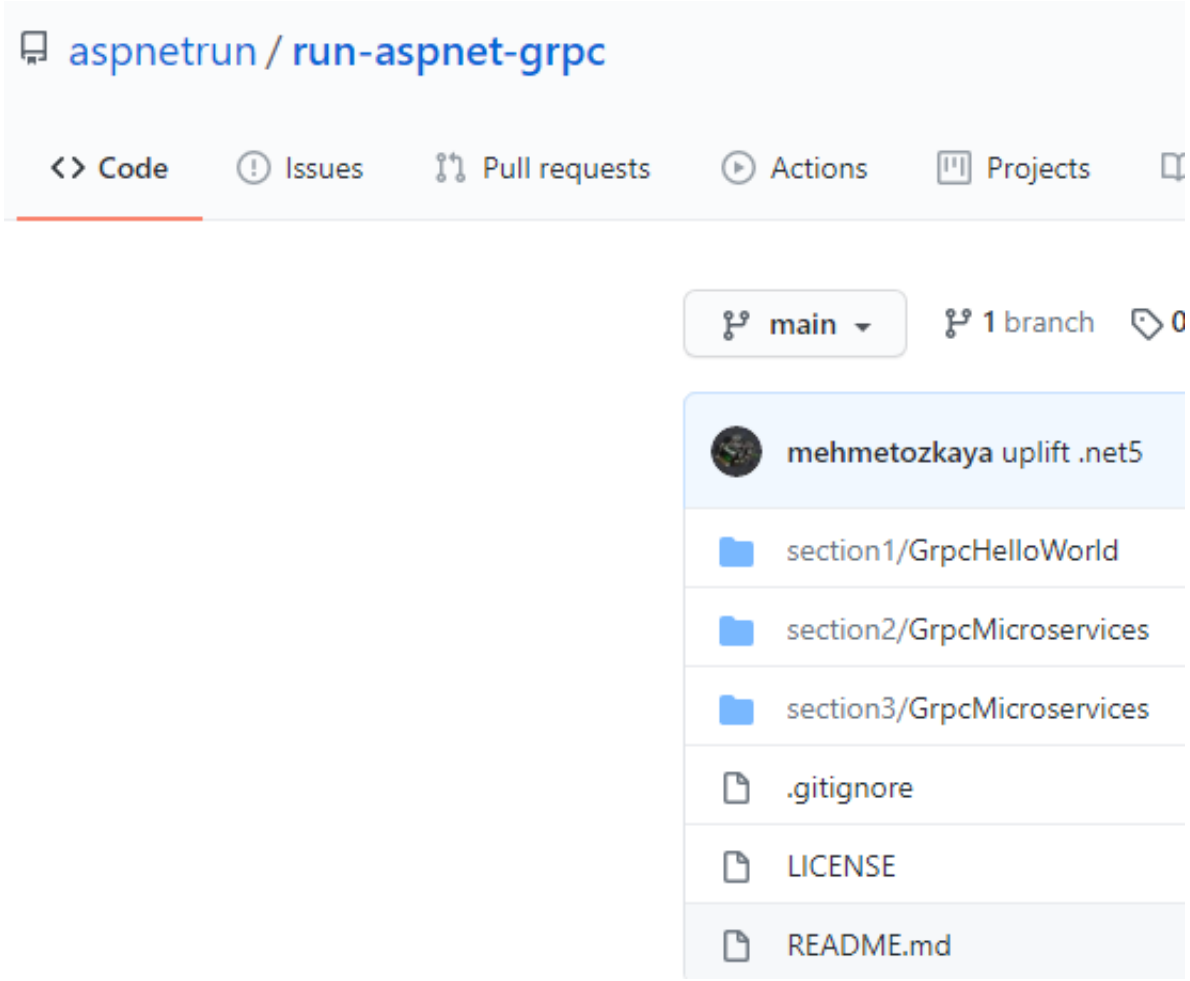

Identity Server
5005
Get Token
Validate Token
and Authorized
Product
Worker
Load Products
Product Grpc
5001
Get
Products
SC Worker
Create SC
& Add Item
SC Grpc
5002
Get Discount
Discount Grpc
5003

Prerequisites

- Good knowledge of C#
- Have knowledge of Asp.Net 5 Web Applications
- No need to know about gRPC
- Basics of Microservices Architecture

Source Code on Github

- Source code on Github
- Fork the repository
- Open issues
- Send Pull Requests



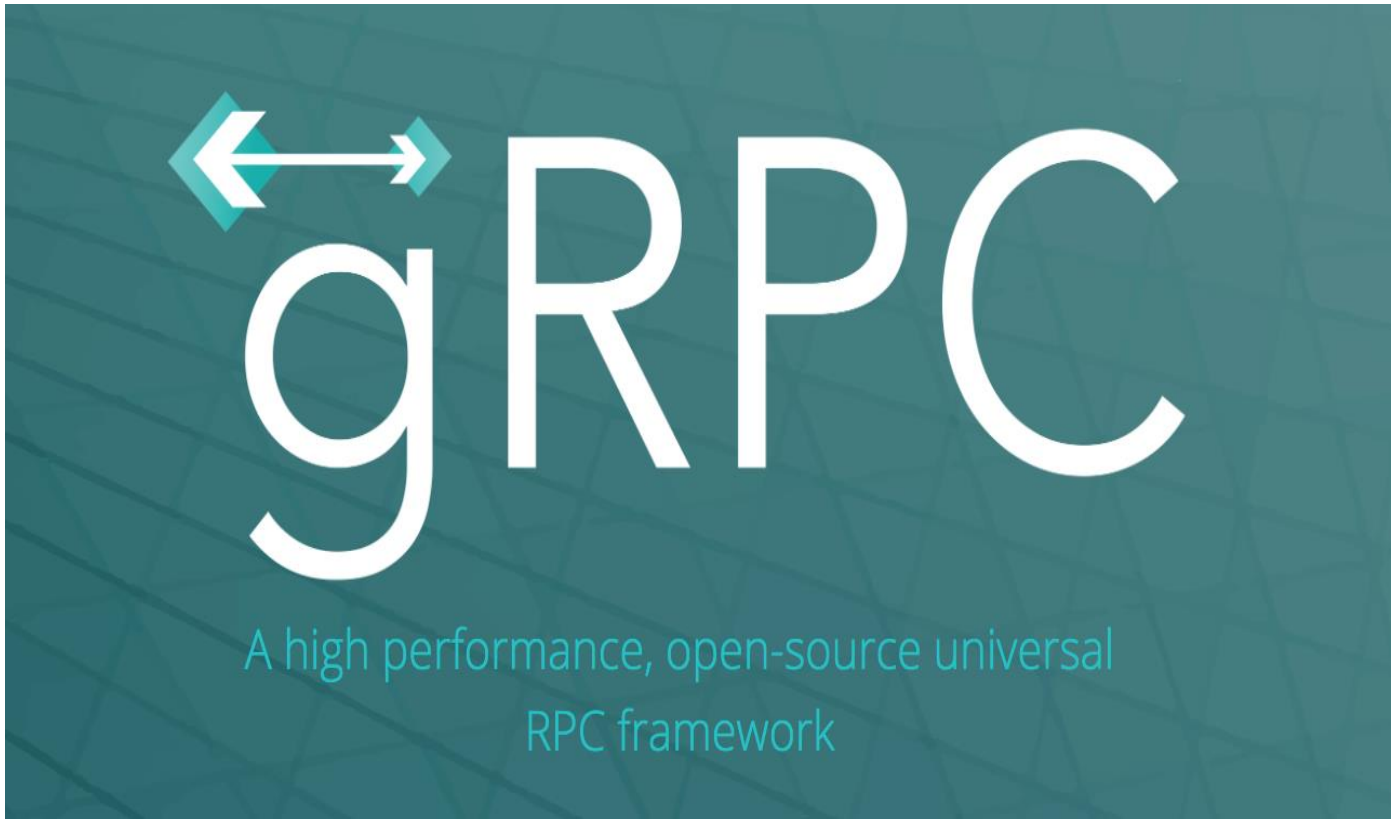
https://github.com/aspnetrun/run-aspnet-grpc

Tooling

- Visual Studio 2019 v16.8
- gRPCurl

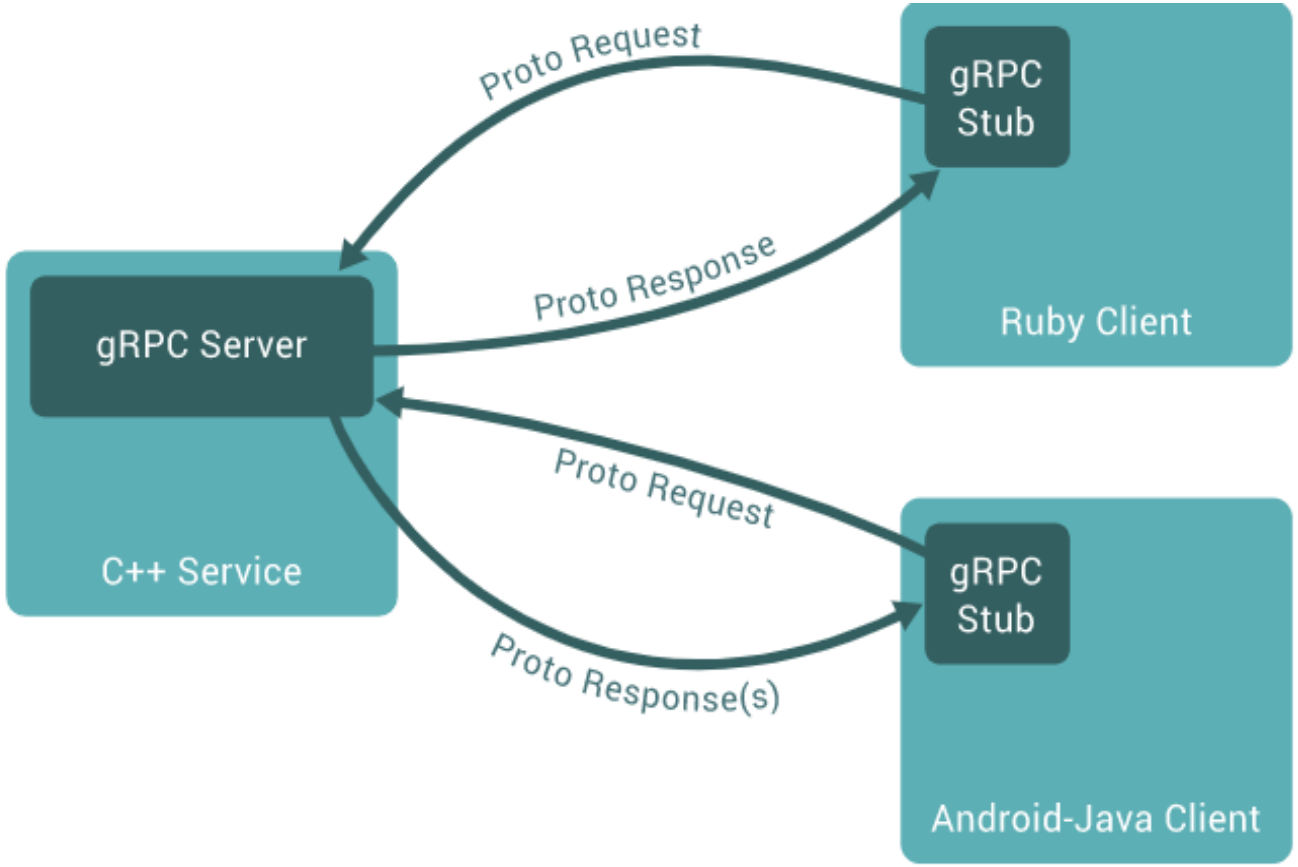
What is gRPC ?

- gRPC (gRPC Remote Procedure Calls)
- HTTP/2 protocol to transport binary messages
- Protocol Buffers, also known as Protobuf files
- Cross-platform client and server bindings



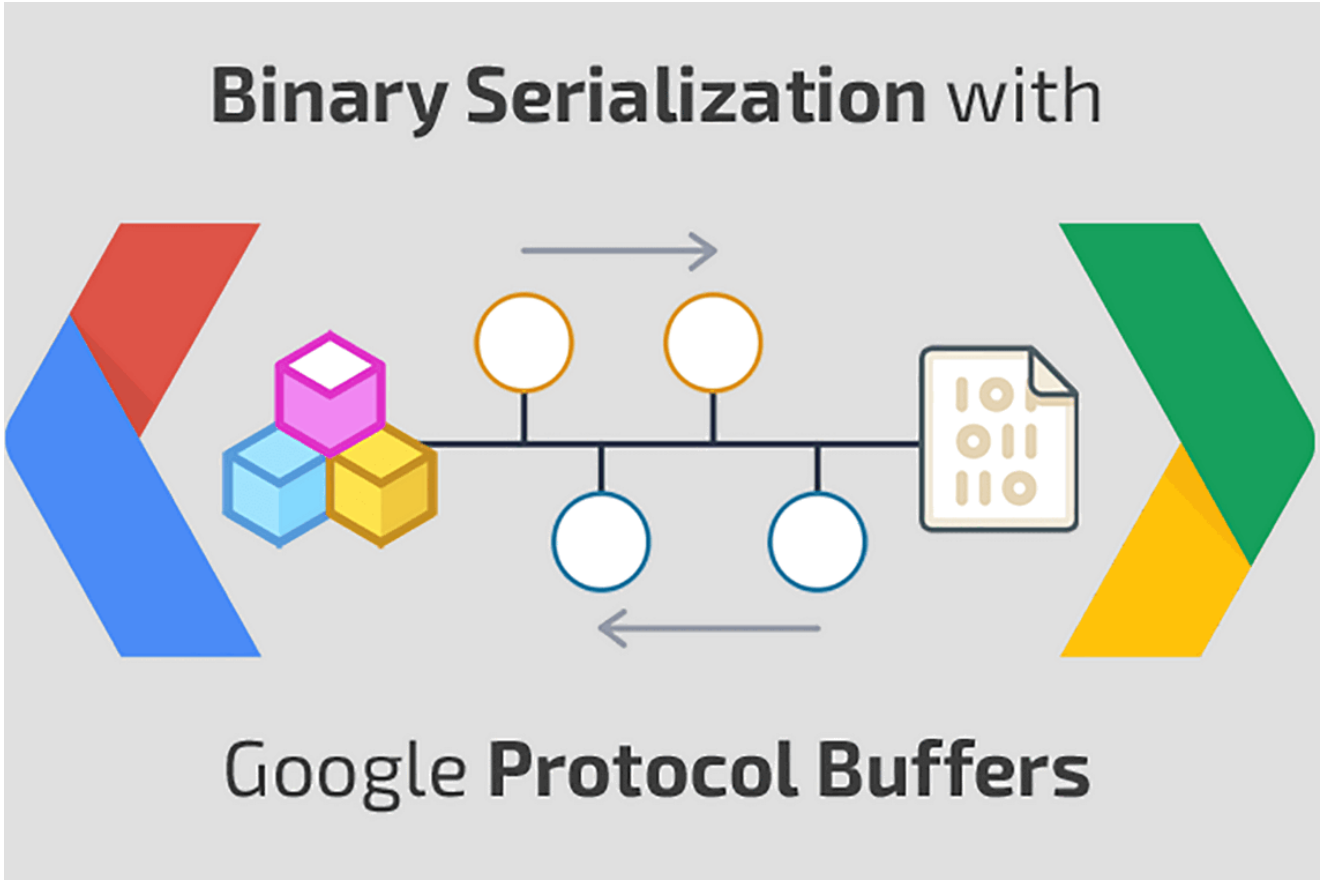
How gRPC works ?

- gRPC directly call a method on a server
- Build distributed applications and services
- Defining a service that specifies methods that can be called remotely
- Can be written in any language that gRPC supports



Working with Protocol Buffers

- gRPC uses Protocol Buffers by Default.
- Protocol Buffers are Google's open source mechanism for serializing structured data.
- serialize in a proto file with the extension .proto.
- Series of name-value pairs



Working with Protocol Buffers 2

- Protocol buffer compiler protocol to create data access classes
- Simple accessors for each field such as name () and set_name (), and methods to serialize
- https://developers.google.com/protocol-buffers/docs/overview

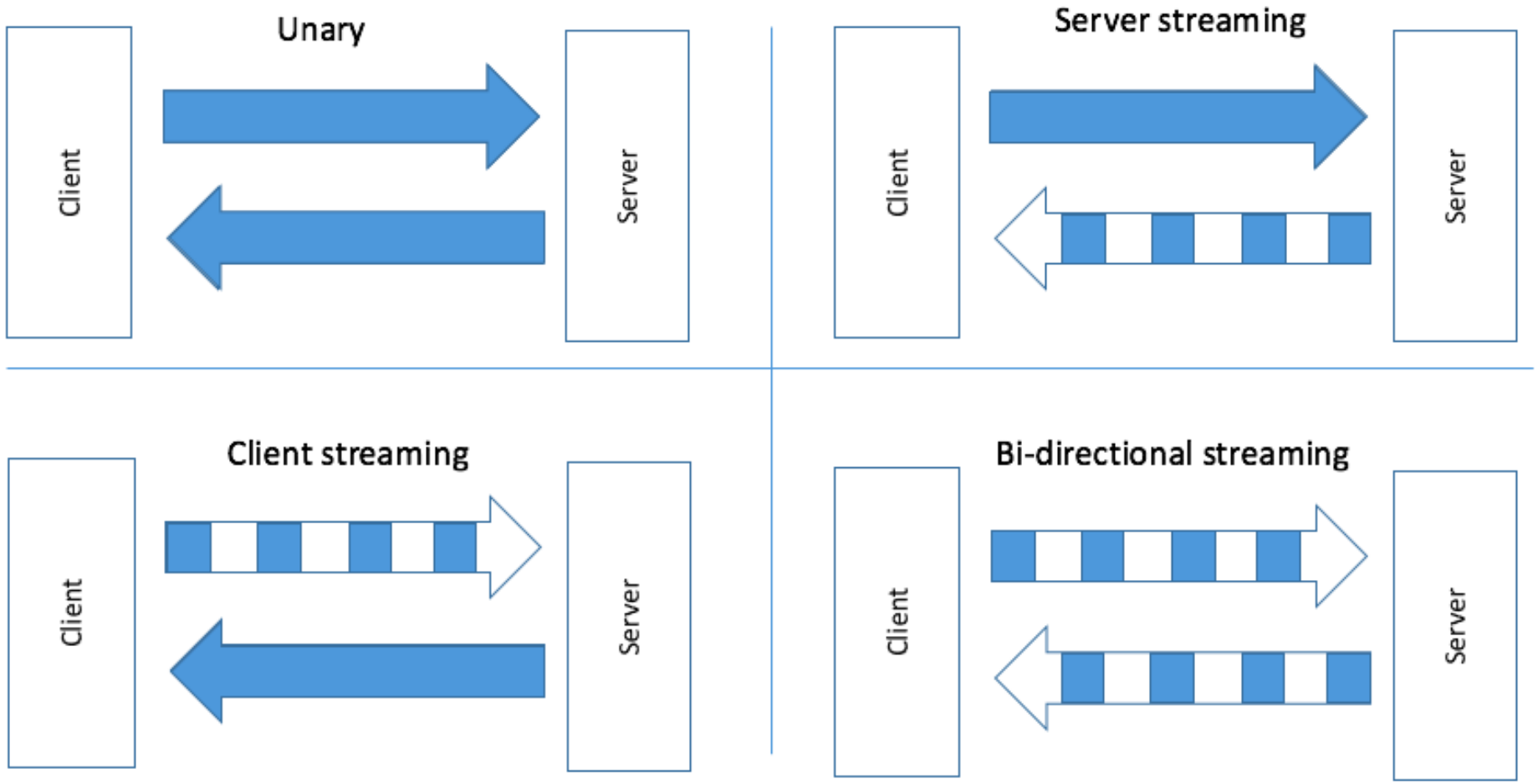
```
message Person {
  string name = 1;
  int32 id = 2;
  bool has_ponycopter = 3;
}
```

Working with Protocol Buffers 3

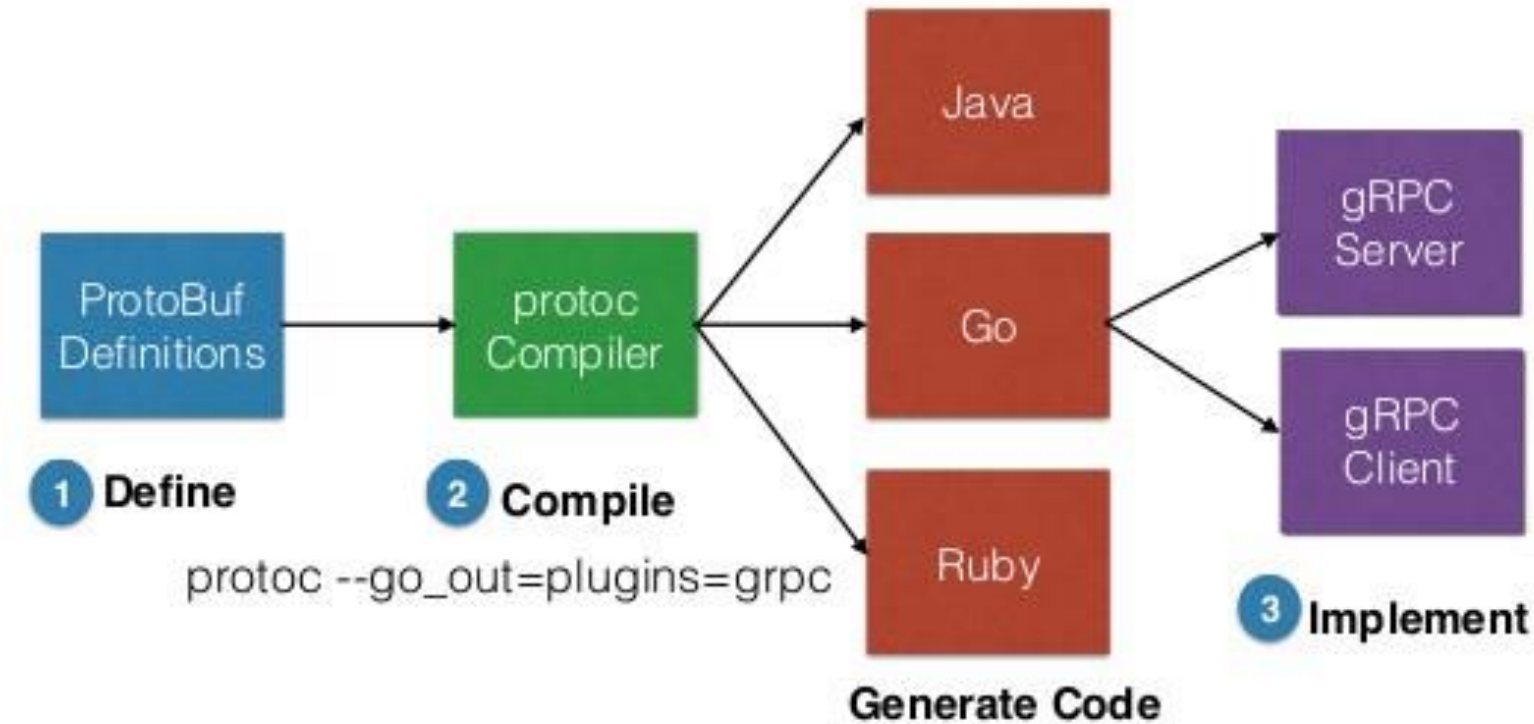
- gRPC service proto files
- RPC method parameters and protocol buffer messages
- https://developers.google.com/protocol-buffers/docs/overview

```
// The greeter service definition.
service Greeter {
  // Sends a greeting
  rpc SayHello (HelloRequest) returns (HelloReply) {}
}

// The request message containing the user's name.
message HelloRequest {
  string name = 1;
}

// The response message containing the greetings
message HelloReply {
  string message = 1;
}
```

gRPC Method Types - RPC life cycles



gRPC Workflow



Advantages of gRPC

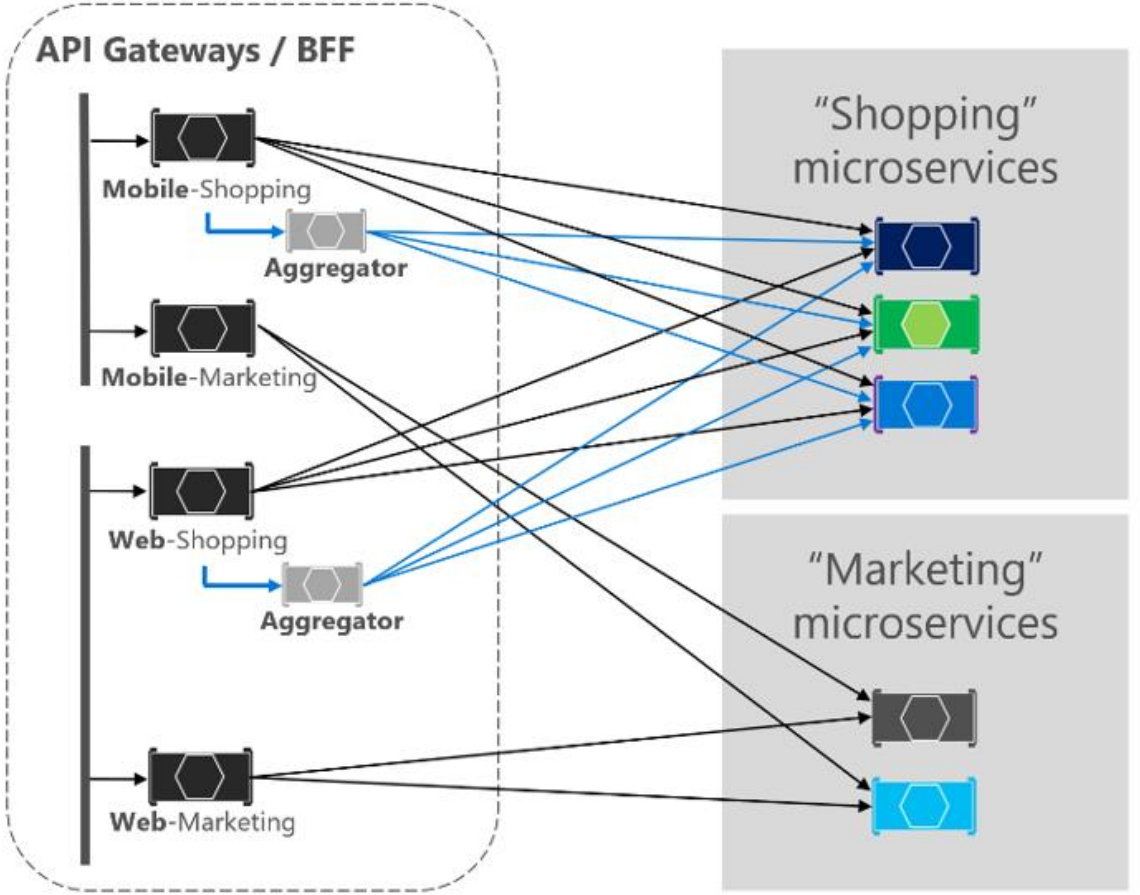
- Using HTTP / 2
- Binary serialization
- Supporting a wide audience with multi-language / platform support
- Open Source and the powerful community behind it
- Supports Bi-directional Streaming operations

gRPC vs REST

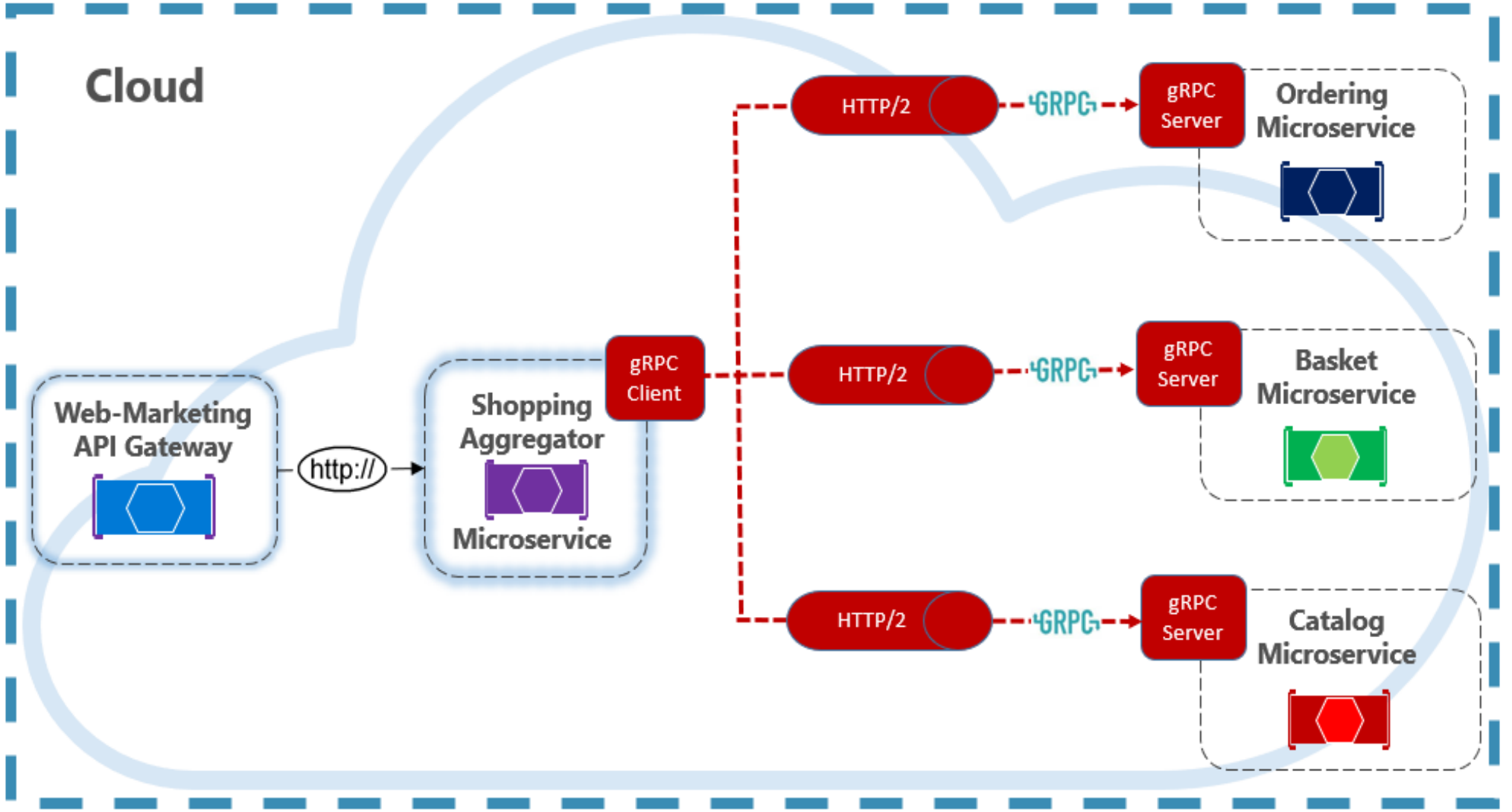
	Goal	REST (HTTP/JSON)	gRPC
COMPATIBILITY	Single source of truth	✗	✔
	Multi-platform + languages built in	✗	✔
	Handle non-breaking changes.	✗	✔
PERFORMANCE	Network: connection handling	Manual, 1 per call ✗	Built-in, Multi per conn ✔
	Speed: Transmission of data	Human-readable Text ✗	Binary ✔
	CPU: Improved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

gRPC usage of Microservices Communication

- Synchronous backend microservice-to-microservice communication
- Polyglot environments
- Low latency and high throughput communication
- Point-to-point real-time communication
- Network constrained environments

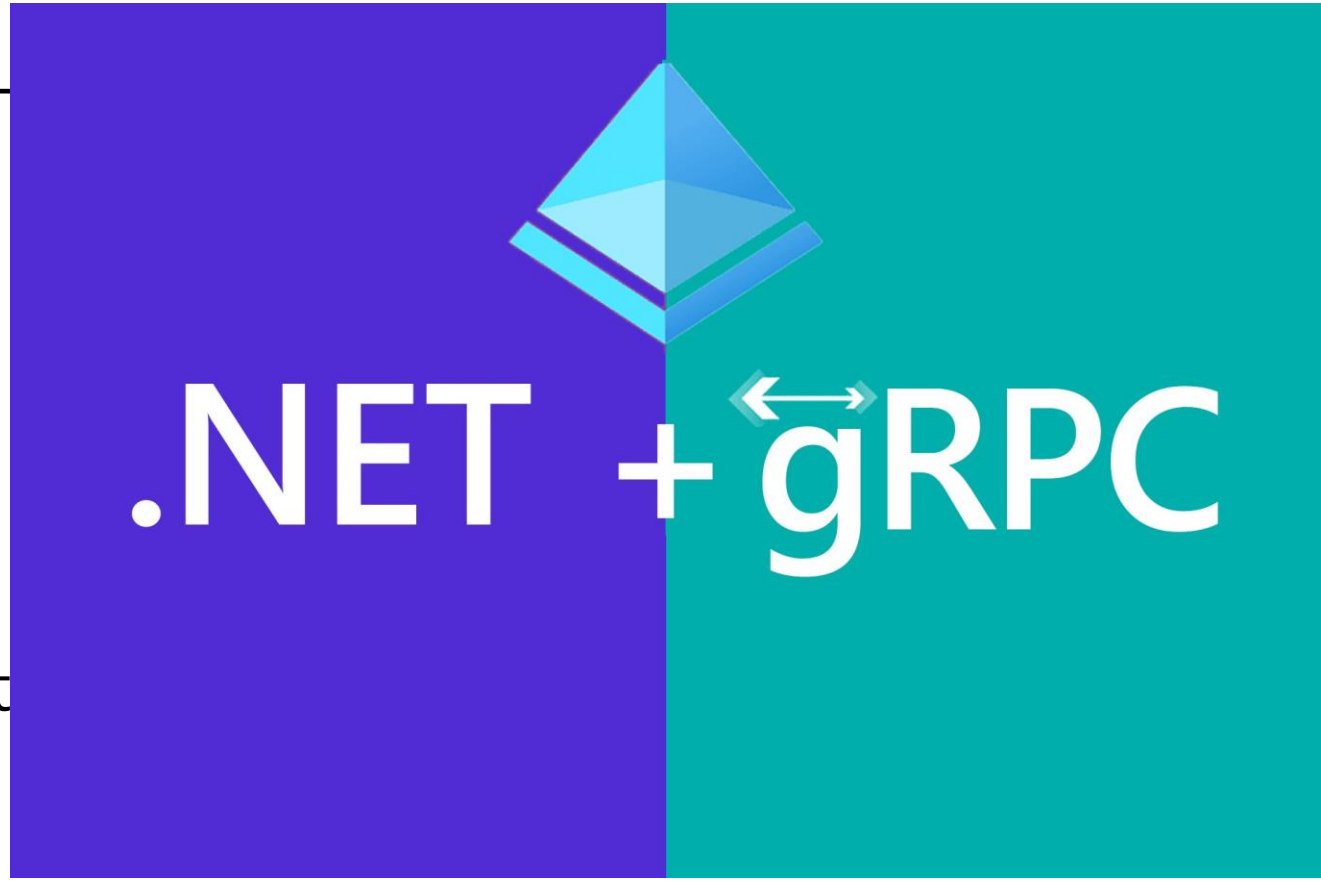


Example of gRPC in Microservices

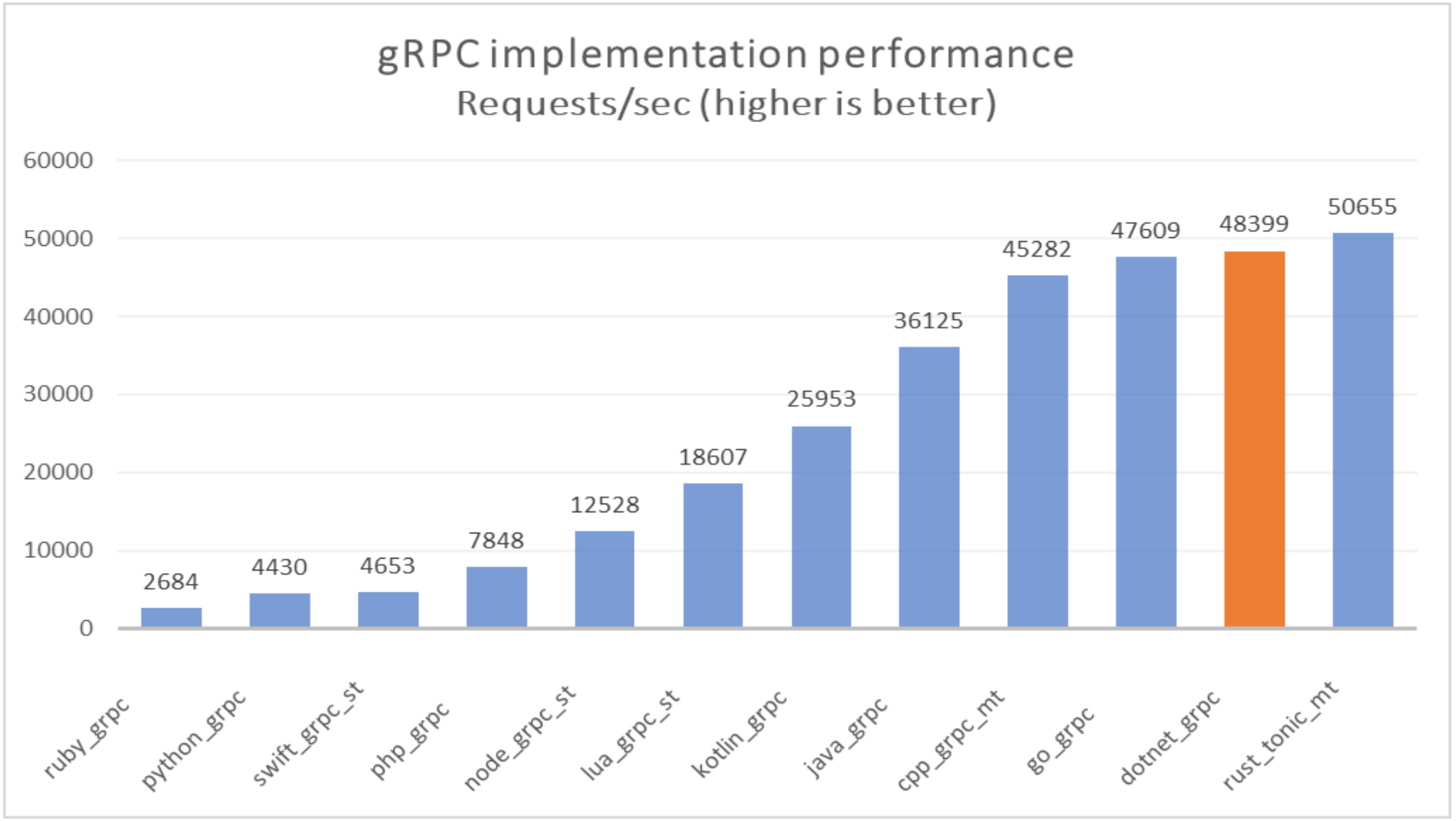


gRPC with .NET 5

- First-class support for gRPC .NET
- Contributed a new implementation of gRPC
- Integrated into .NET Core 3.0 SDK and later
- Visual Studio 2019 template that scaffolds a skeleton project



gRPC performance in .NET 5



- https://devblogs.microsoft.com/aspnet/grpc-performance-improvements-in-net-5/

END OF SECTION

Next Section

HelloWorld gRPC with Asp.Net 5

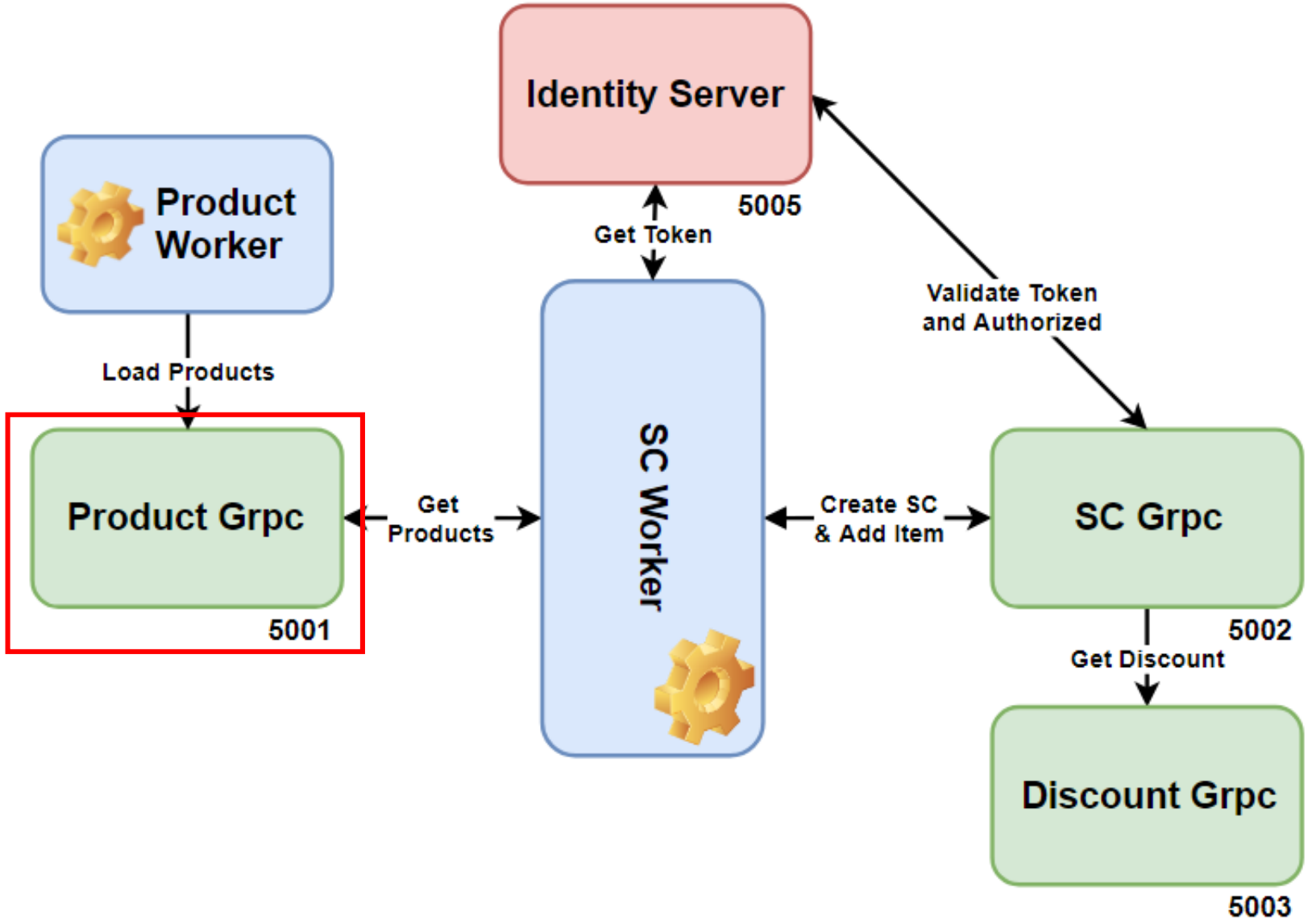
Section 2
HelloWorld gRPC with Asp.Net 5

for building HelloWorld Asp.Net gRPC Project

Section 3
Building Product Grpc Microservices

for Exposing Product CRUD APIs

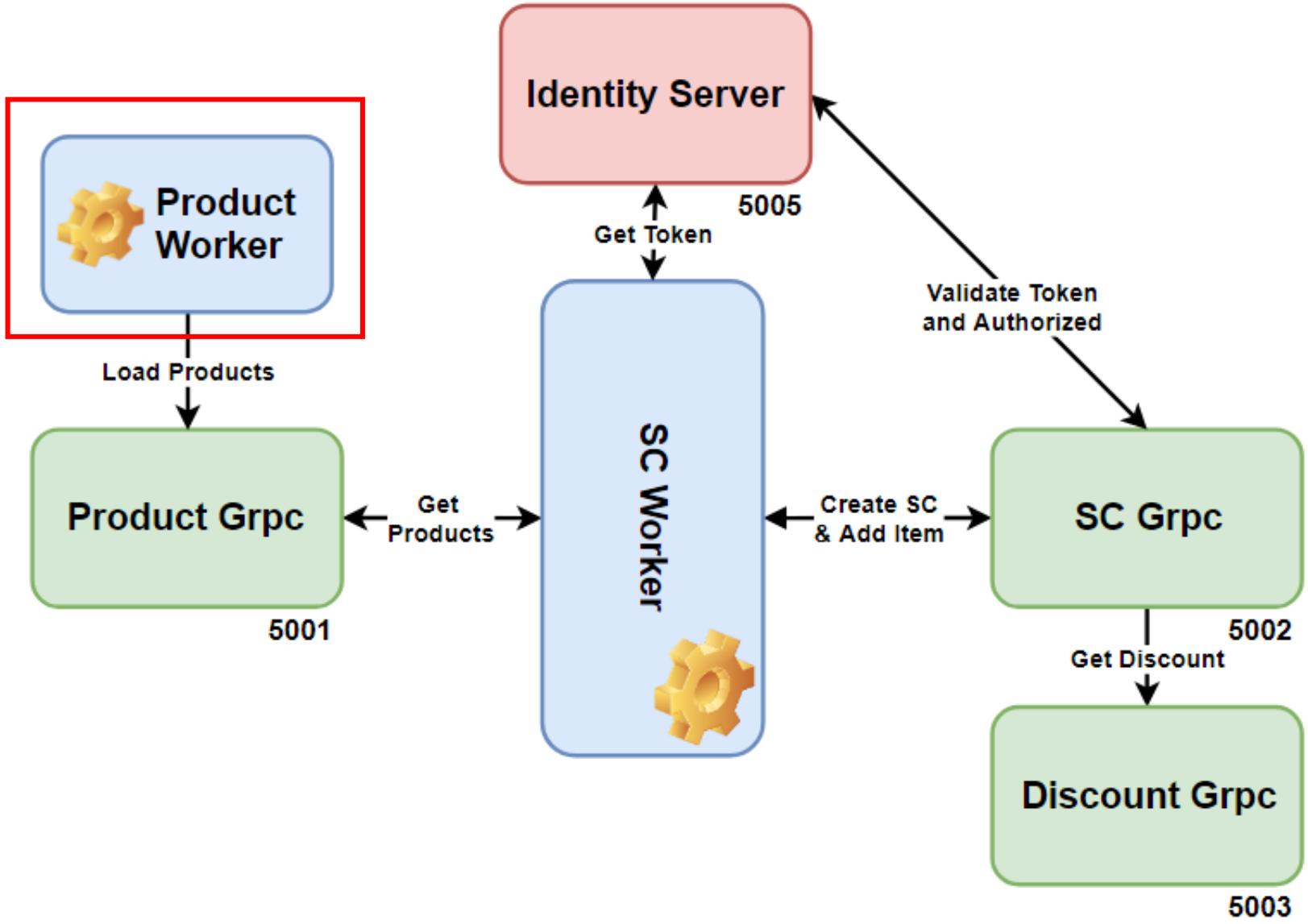
Big Picture


Identity Server
5005
Get Token
Validate Token
and Authorized
Product
Worker
Load Products
Product Grpc
5001
Get
Products
SC Worker
Create SC
& Add Item
SC Grpc
5002
Get Discount
Discount Grpc
5003

Section 4
Building Product WorkerService Microservices

for Generate and Insert Product to ProductGrpc Microservices

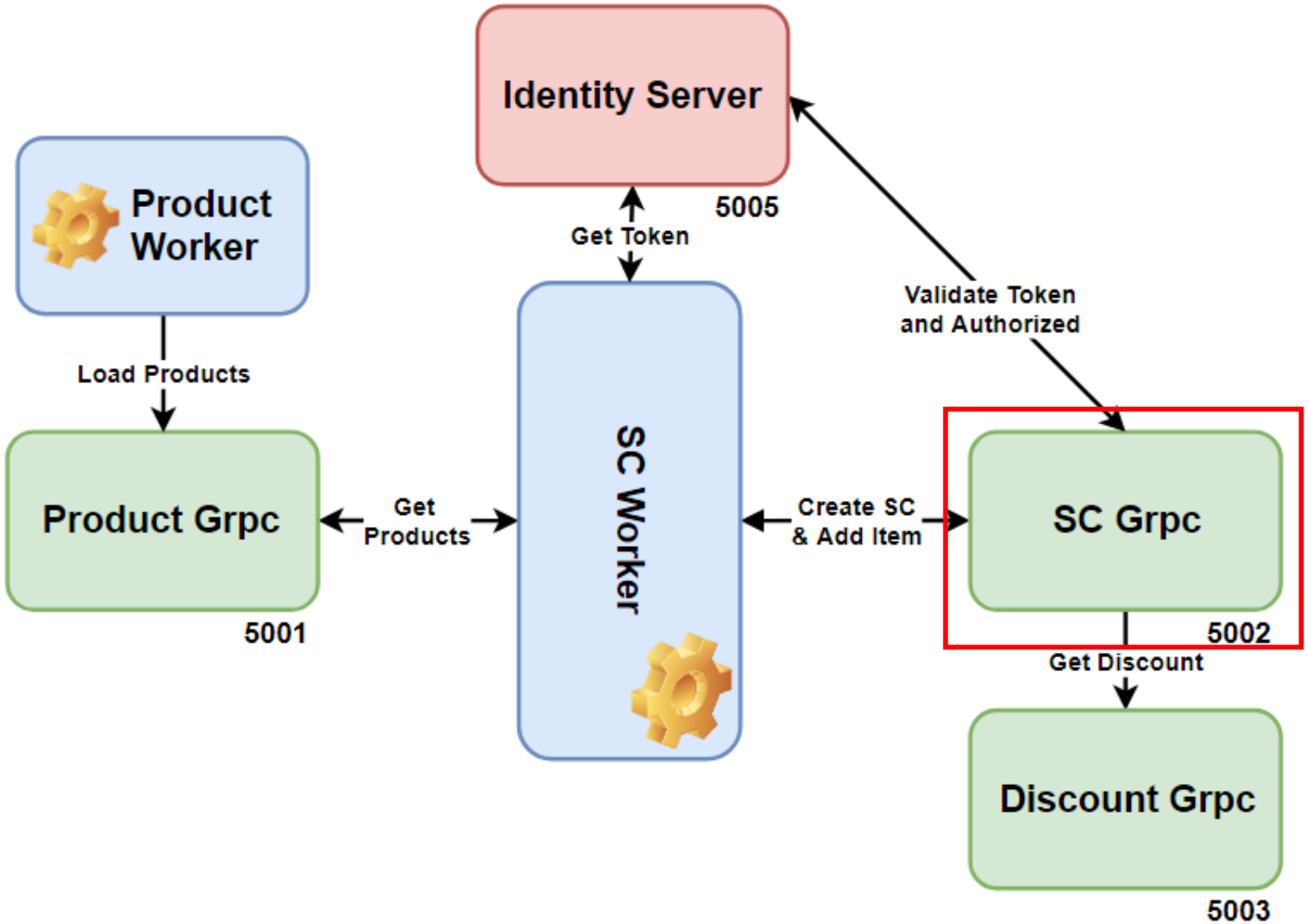
Big Picture


Identity Server
5005
Get Token
Validate Token
and Authorized
Product
Worker
Load Products
Product Grpc
5001
Get
Products
SC Worker
Create SC
& Add Item
SC Grpc
5002
Get Discount
Discount Grpc
5003

Section 5
Building Shopping Cart Grpc Server Application

for Storing Products into Cart

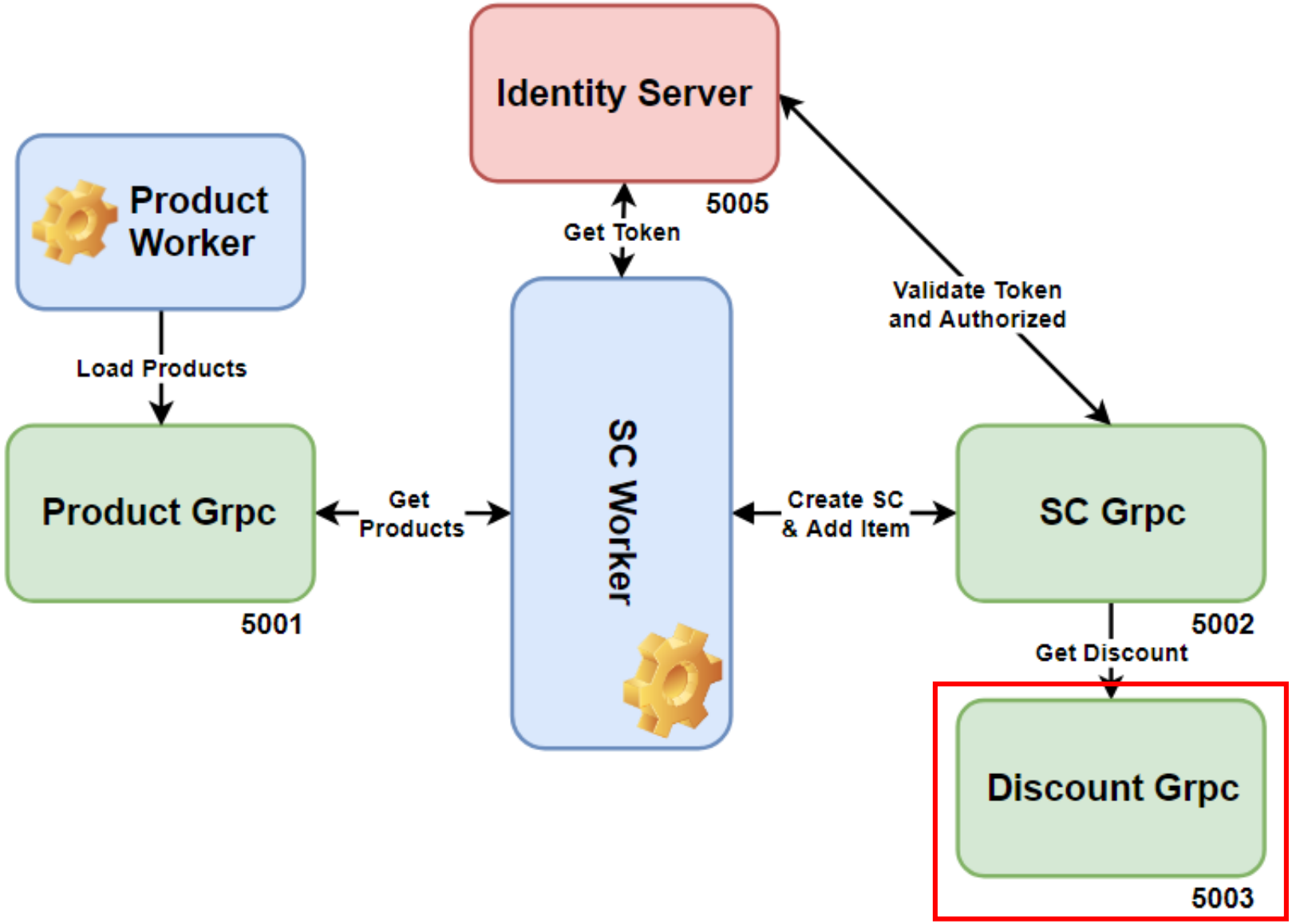
Big Picture


Identity Server
5005
Get Token
Validate Token
and Authorized
Product
Worker
Load Products
Product Grpc
5001
Get
Products
SC Worker
Create SC
& Add Item
SC Grpc
5002
Get Discount
Discount Grpc
5003

Section 6
Building Dicsount Grpc Microservice

for Inter-Service Communication for Calculating Every Cart Item Last Price

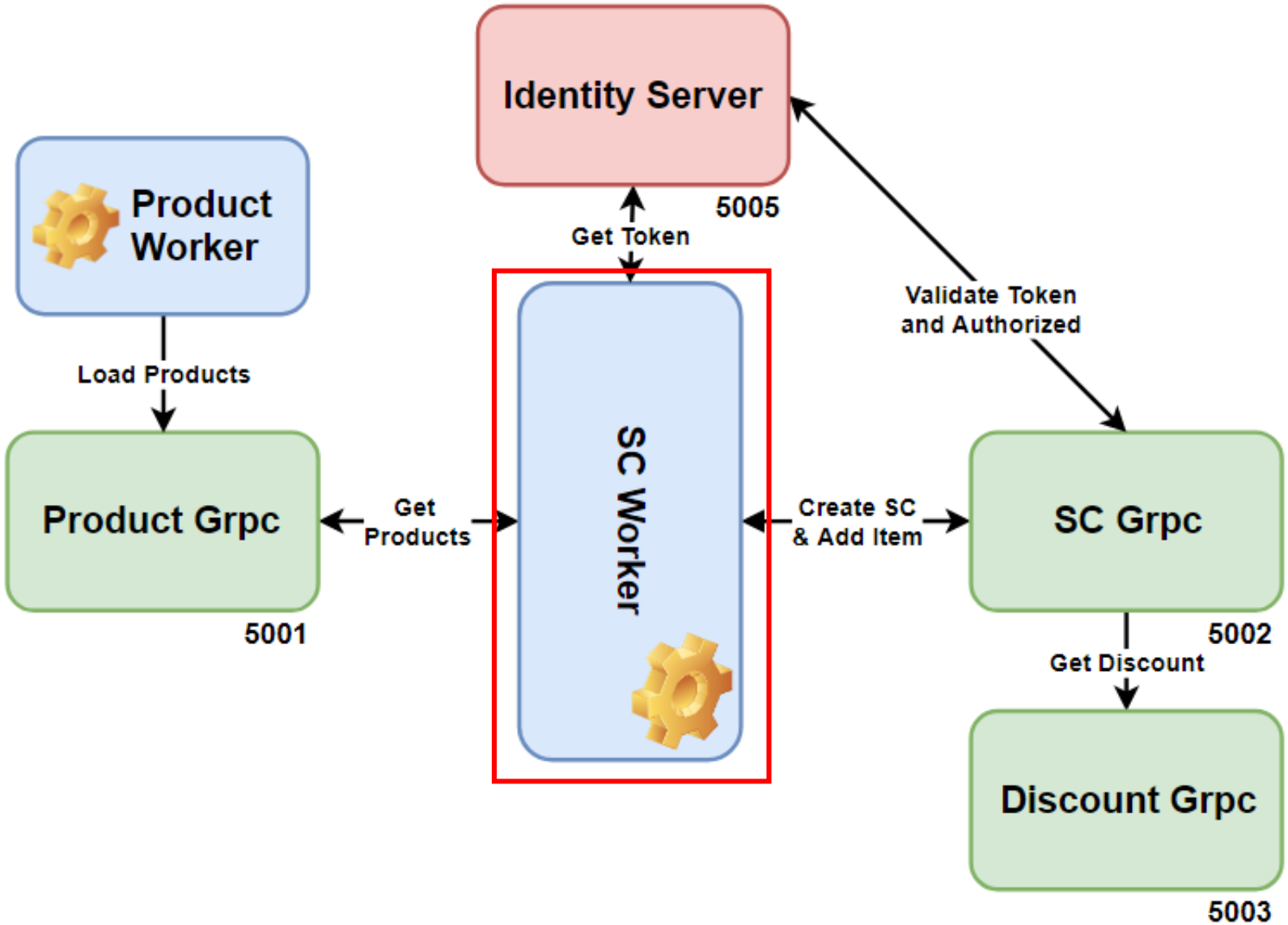
Big Picture


Identity Server
5005
Get Token
Validate Token
and Authorized
Product
Worker
Load Products
Product Grpc
5001
Get
Products
SC Worker
Create SC
& Add Item
SC Grpc
5002
Get Discount
Discount Grpc
5003

Section 7
Building ShoppingCart WorkerService Microservice

for Retrieve Products and Add to ShoppingCart with consuming ProductGrpc and ShoppingCartGrpc Microservices

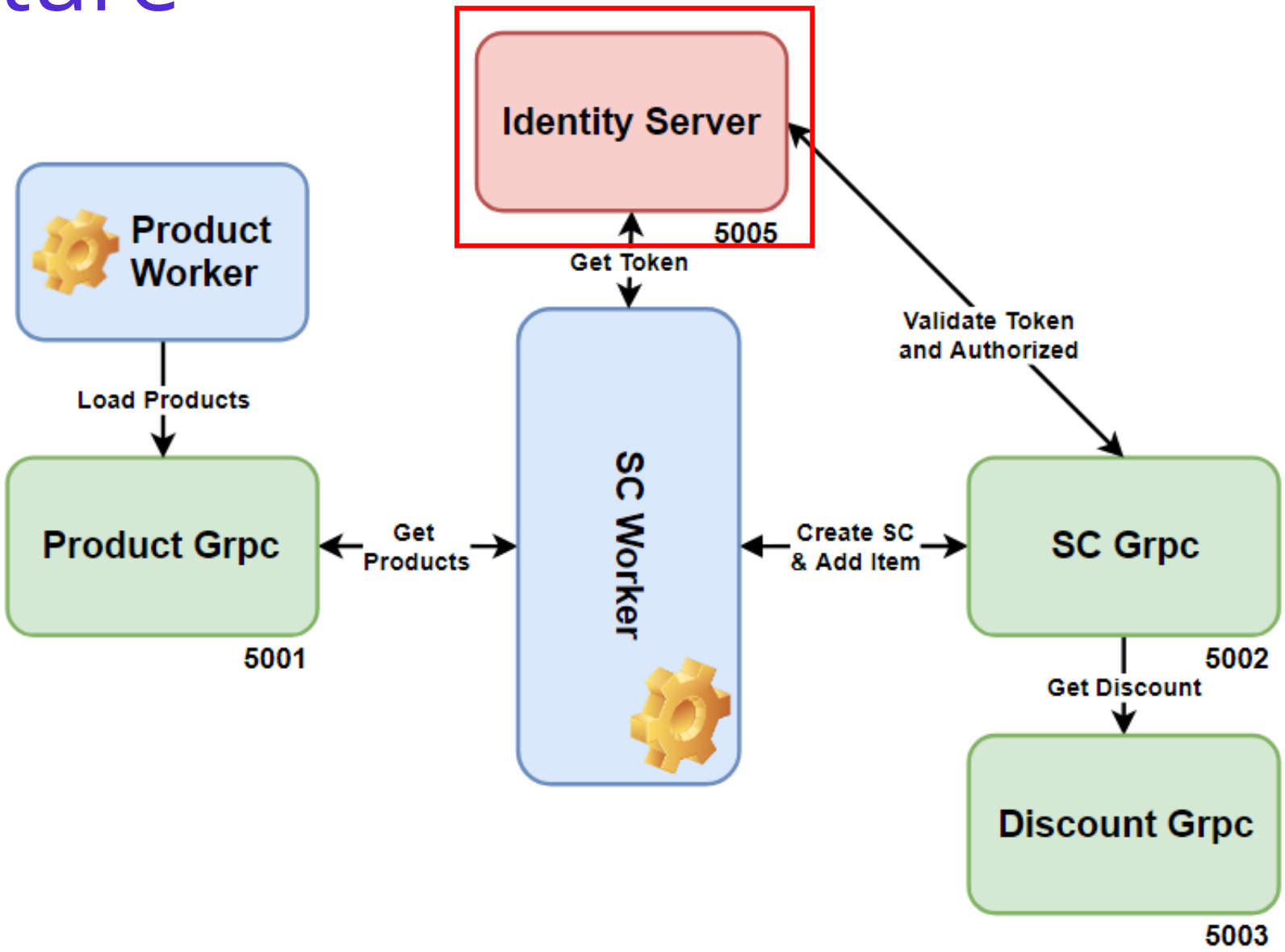
Big Picture


Identity Server
5005
Get Token
Validate Token
and Authorized
Product
Worker
Load Products
Product Grpc
5001
Get
Products
SC Worker
Create SC
& Add Item
SC Grpc
5002
Get Discount
Discount Grpc
5003

Section 8
Authenticate gRPC Services with IdentityServer4

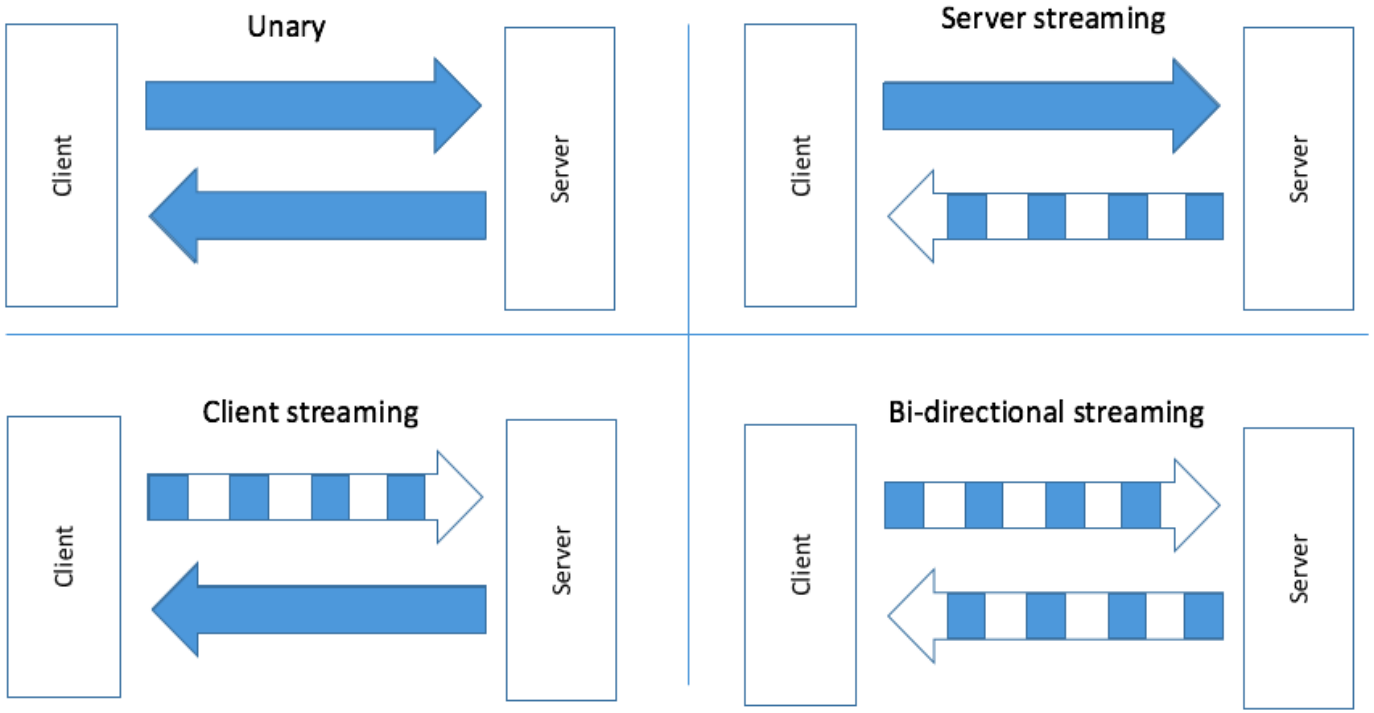
for protect ShoppingCartGrpc Method with OAuth 2.0 and JWT Bearer Token

Big Picture


Identity Server
5005
Get Token
Validate Token
and Authorized
Product
Worker
Load Products
Product Grpc
5001
Get
Products
SC Worker
Create SC
& Add Item
SC Grpc
5002
Get Discount
Discount Grpc
5003

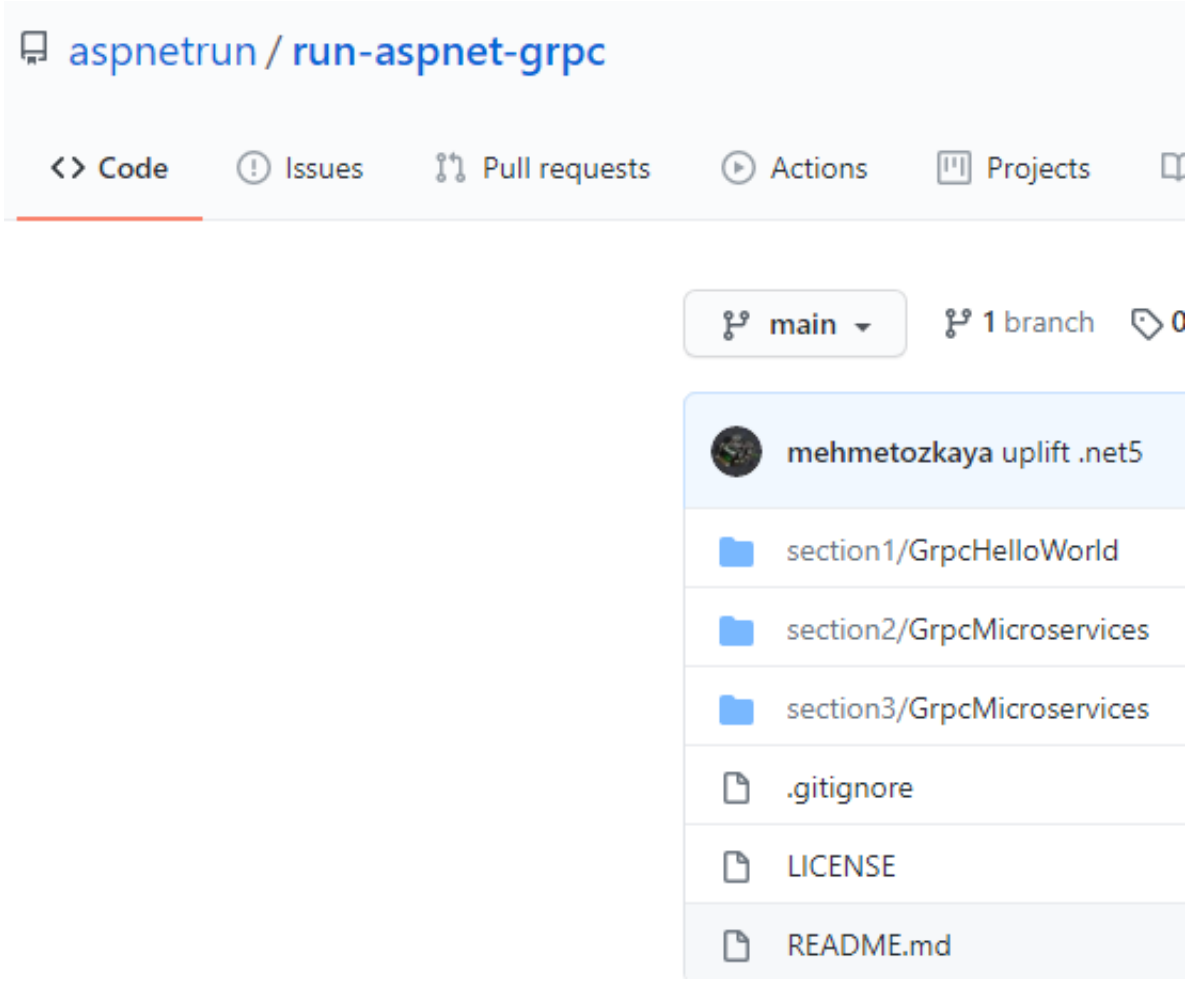
Assignment - Bidirectional streaming RPCs with Support Microservices

- Bidirectional streaming RPCs
- Support gRPC Server
- AI-powered assistant with chatting on the user channel
- Bi-directional streaming gRPC communication with client and Support gRPC server
- Share your repositories with me



Source Code on Github

- Source code on Github
- Fork the repository
- Open issues
- Send Pull Requests



https://github.com/aspnetrun/run-aspnet-grpc

Thanks!

Follow me on github
https://github.com/mehmetozkaya

Follow me on twitter
https://twitter.com/ezozkme

Follow me on medium
https://medium.com/@mehmetozkaya

Send mail me anything you can ask
ezozkme@gmail.com

Check my other courses on udemy:
https://www.udemy.com/user/ff0e5c8c-dd71-443e-be0a-e73ba821f7d7/