ORGANIZATION

plain concepts

PLATINUM SPONSORS

Microsoft

COLLABORATORS

CEU | Universidad San Pablo

Thank you!

Eduard Tomàs

Doing stuff with computers

Level 10 standard freak. I love videogames, **good** fantasy and sci-fi books and playing role games (*real* ones, you know with paper, stencils and strange dices). I also love programming and creating things using computers since I was a child and I started with BASIC on my Spectrum.

Right now I am focused in cloud-native architectures and Microsoft technologies. That means containers, Kubernetes, netcore, Azure,…

I am also a beer crafter and drinker, always ready to share a beer with everyone!
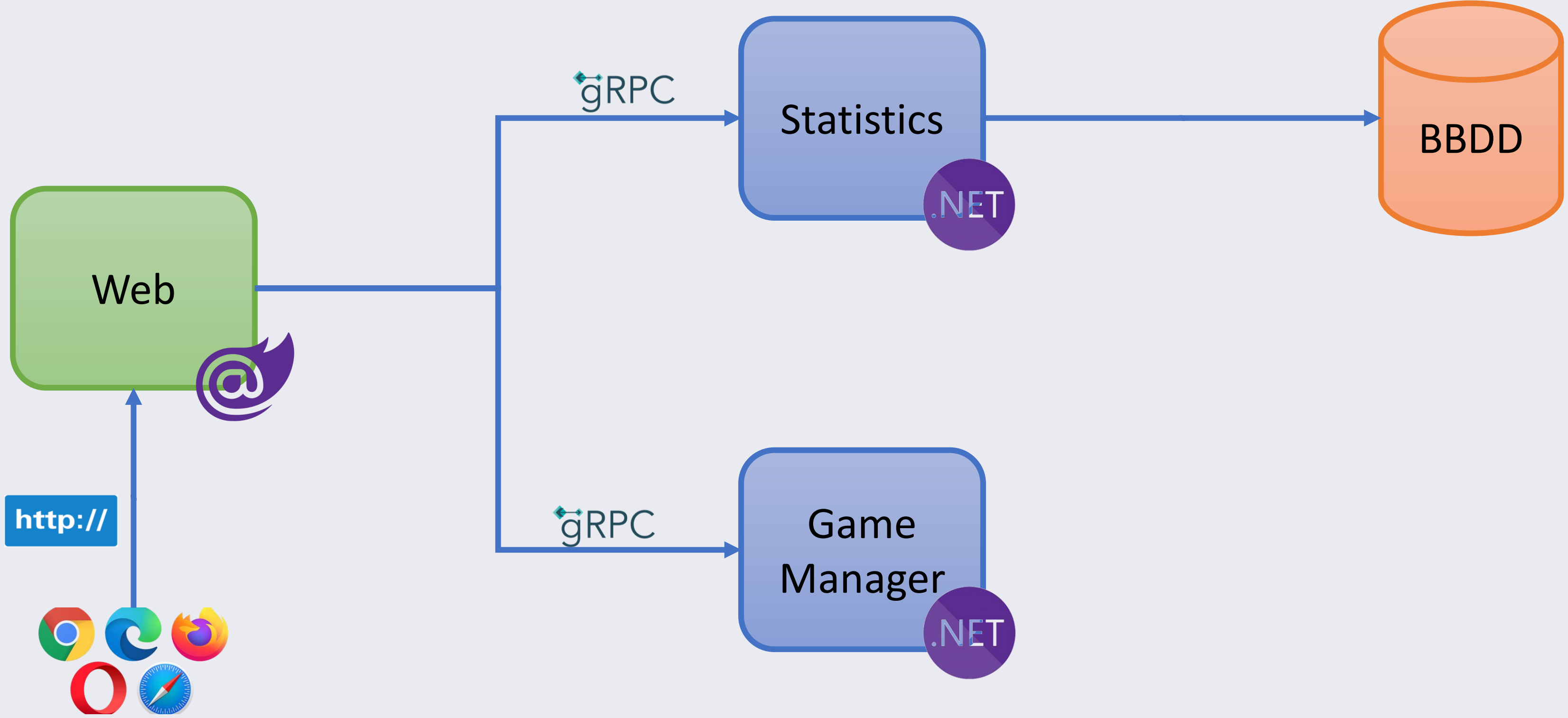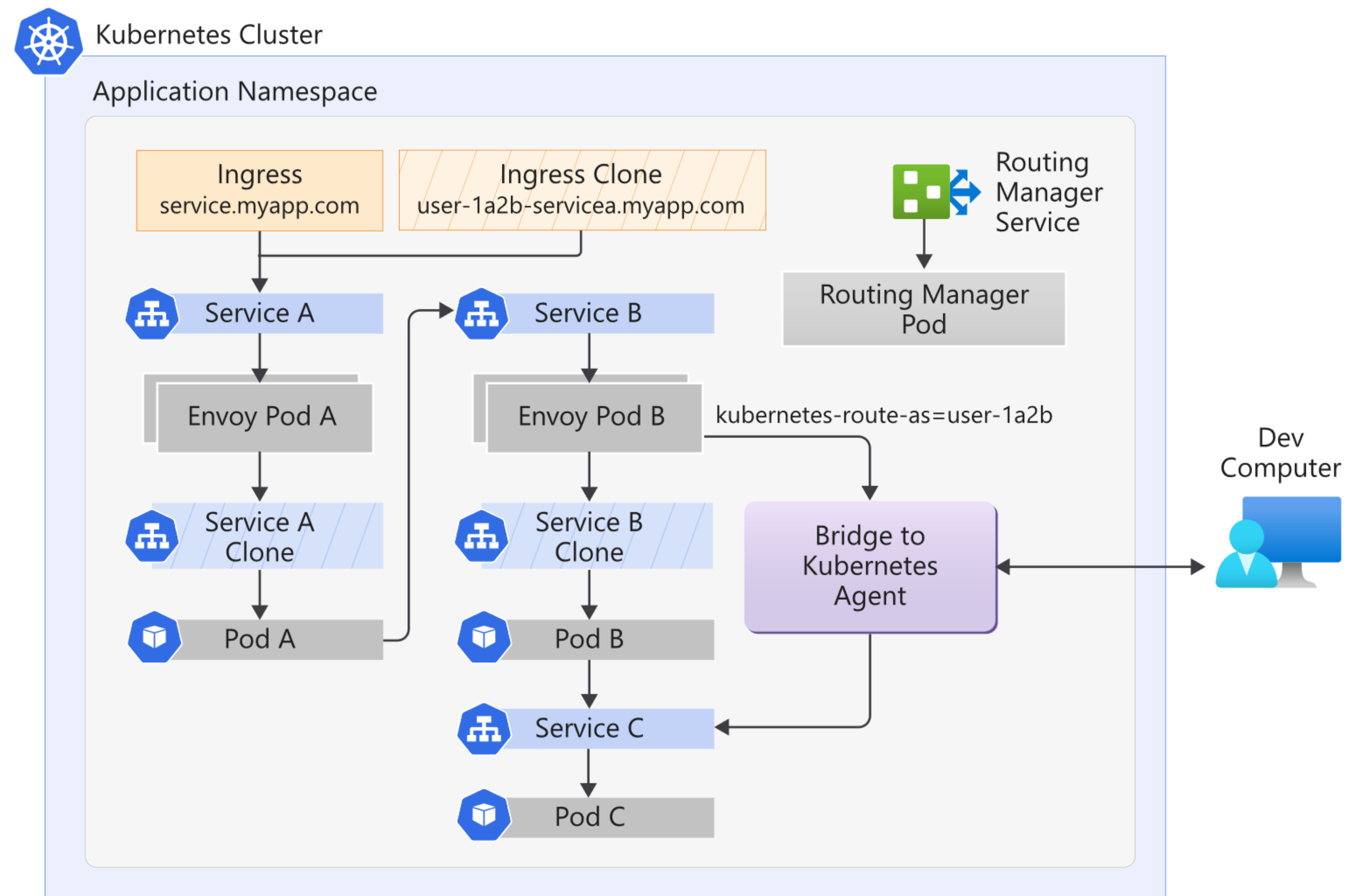
@eiximenis

etomas@plainconcepts.com

What we will see today?

1. How to debug my microservice without deploying everything

2. How to run locally my microservices without dying

3. Building microservices has never been so easily

4. REST was good, but maybe it's time to try something different

5. If you can't deploy it, you have nothing

6. What the hell is YARP and what we could expect from it

Bridge to Kubernetes

#DotNet2020

Kubernetes Cluster

Application Namespace

Ingress
service.myapp.com

Ingress Clone
user-1a2b-servicea.myapp.com

Routing Manager Service

Routing Manager Pod

Service A

Service B

Envoy Pod A

Envoy Pod B

kubernetes-route-as=user-1a2b

Dev Computer

Service A Clone

Service B Clone

Bridge to Kubernetes Agent

Pod A

Pod B

Service C

Pod C

How to run locally my microservices without dying...

Welcome to *Project Tye*!

How to run locally my microservices without dying
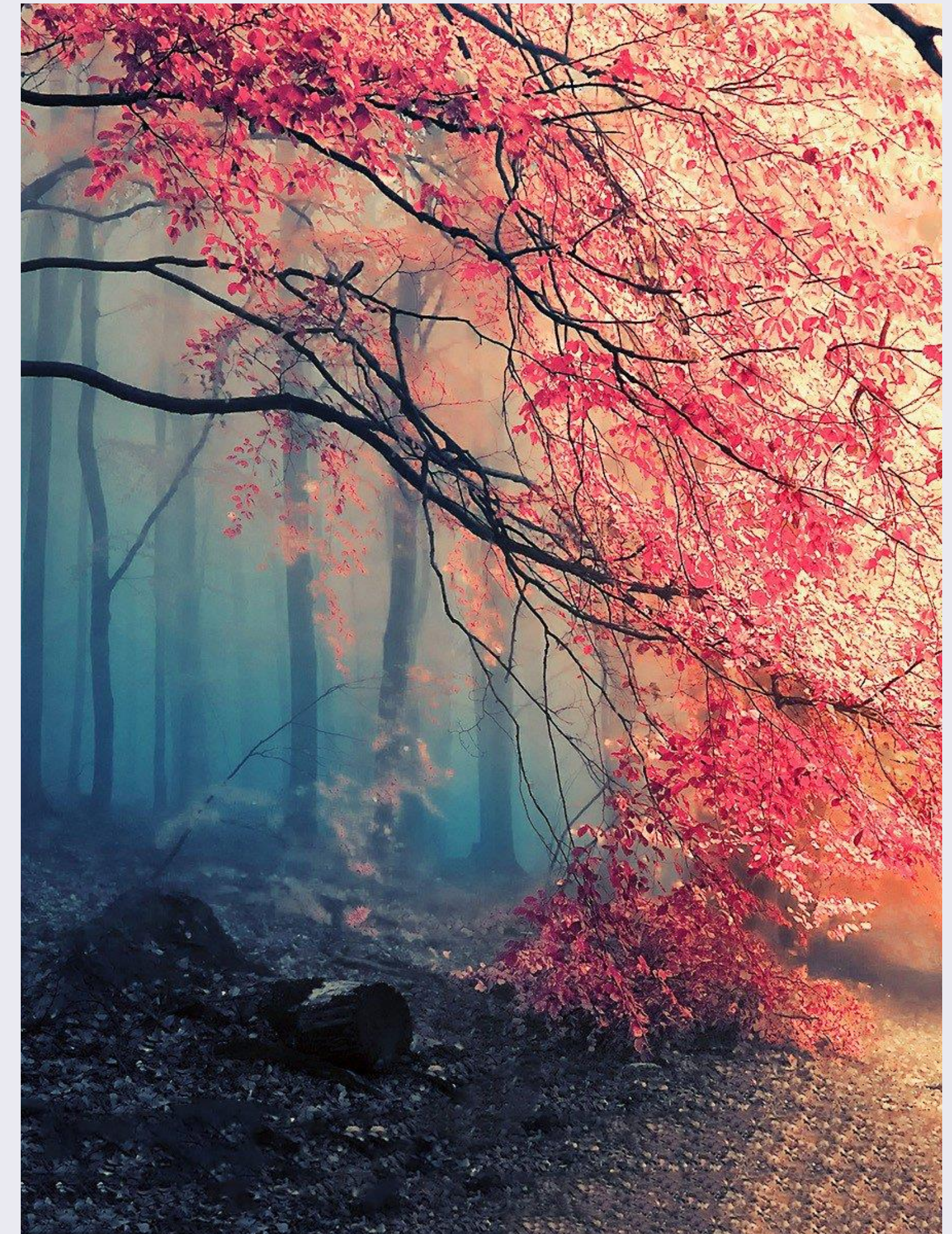
Project Tye

dotnet global tool

Can handle external dependencies (Docker images)

Gives service discovery and port remapping
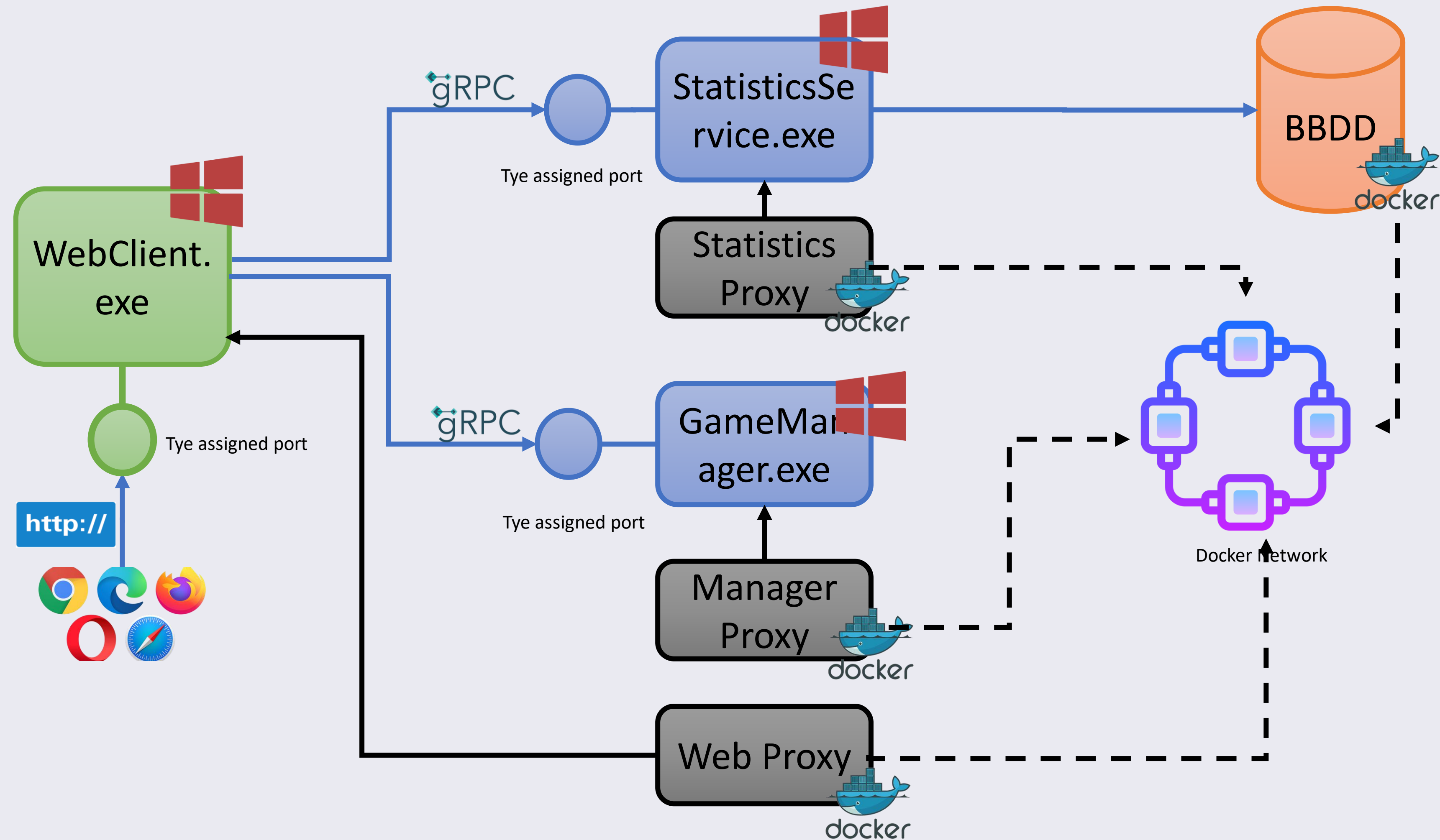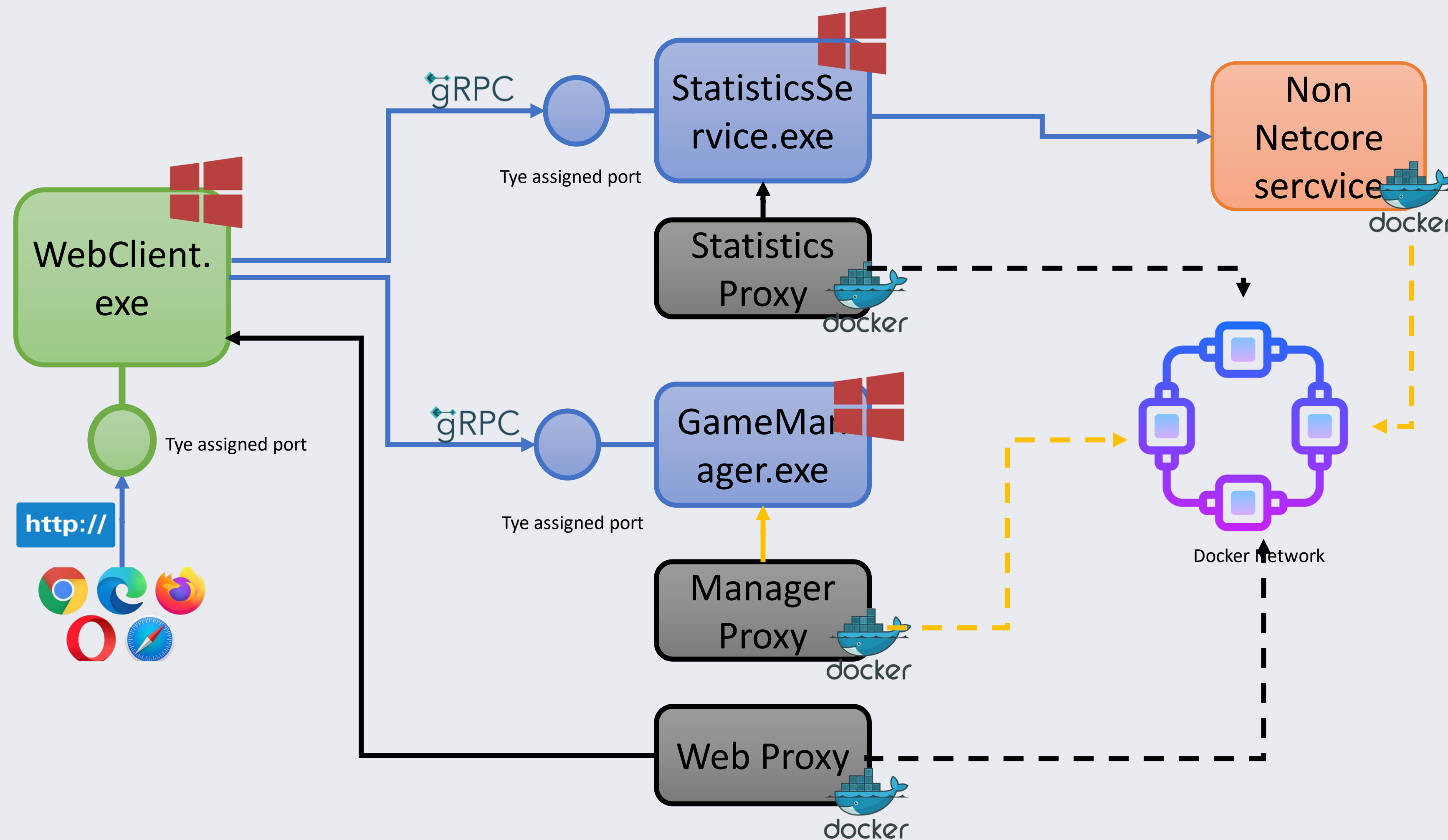
Handle failures

Collect logs and metrics

DotNet**2020**

Project tye its like "dotnet run" but for microservices

Someone in some tweet…

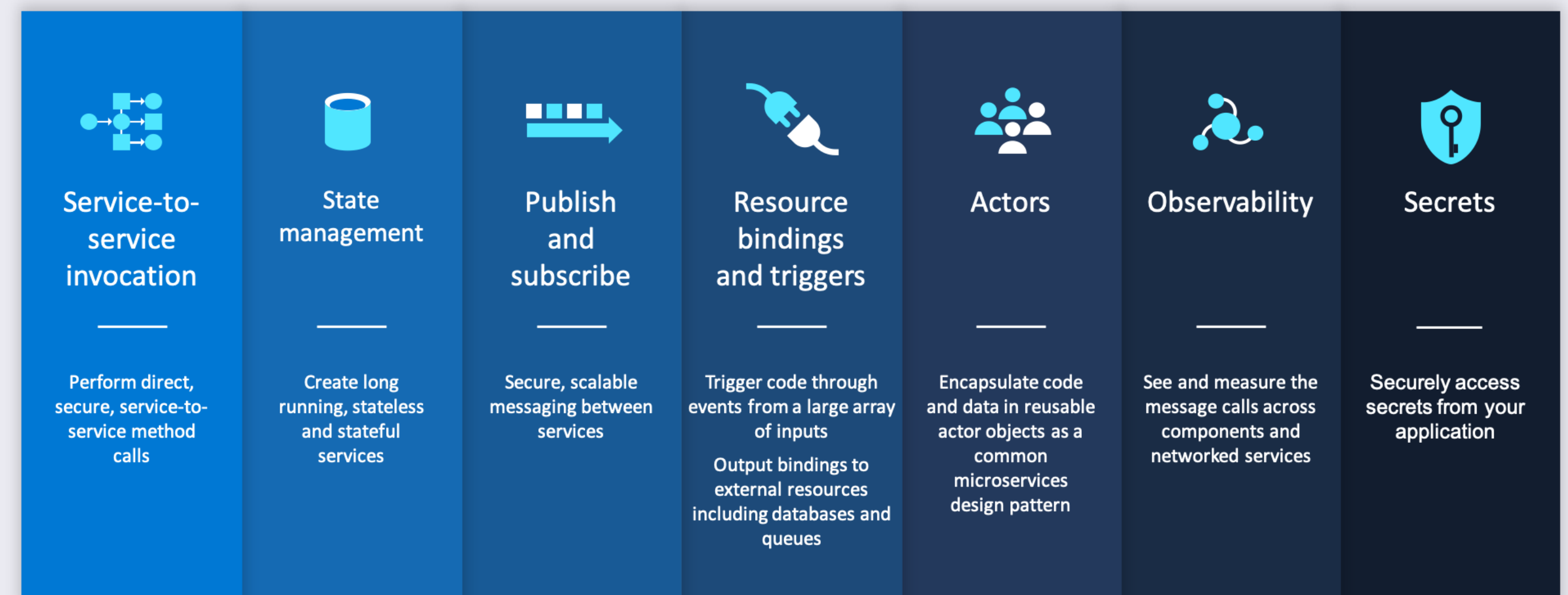# Building microservices has never been so easily
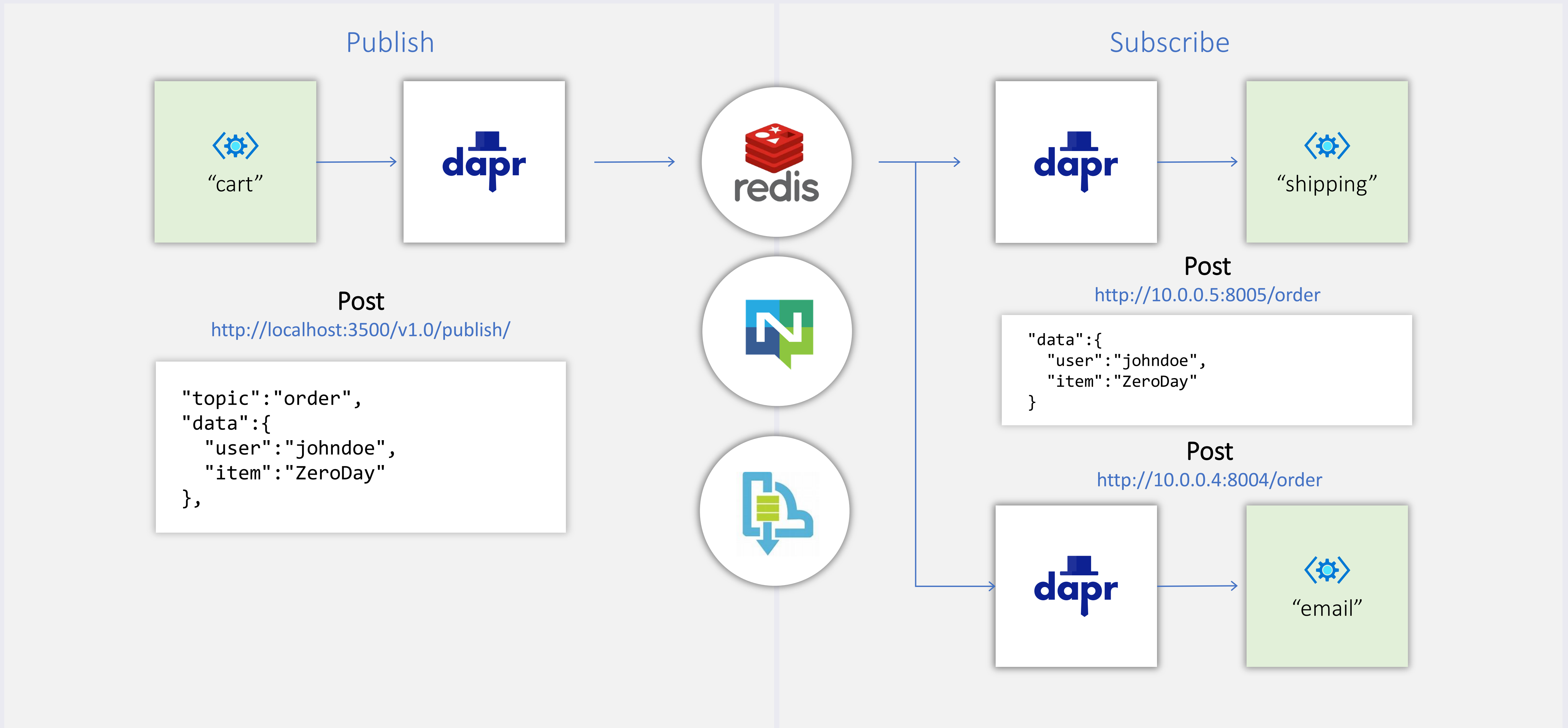
Do yo know about Dapr?

What is Dapr?

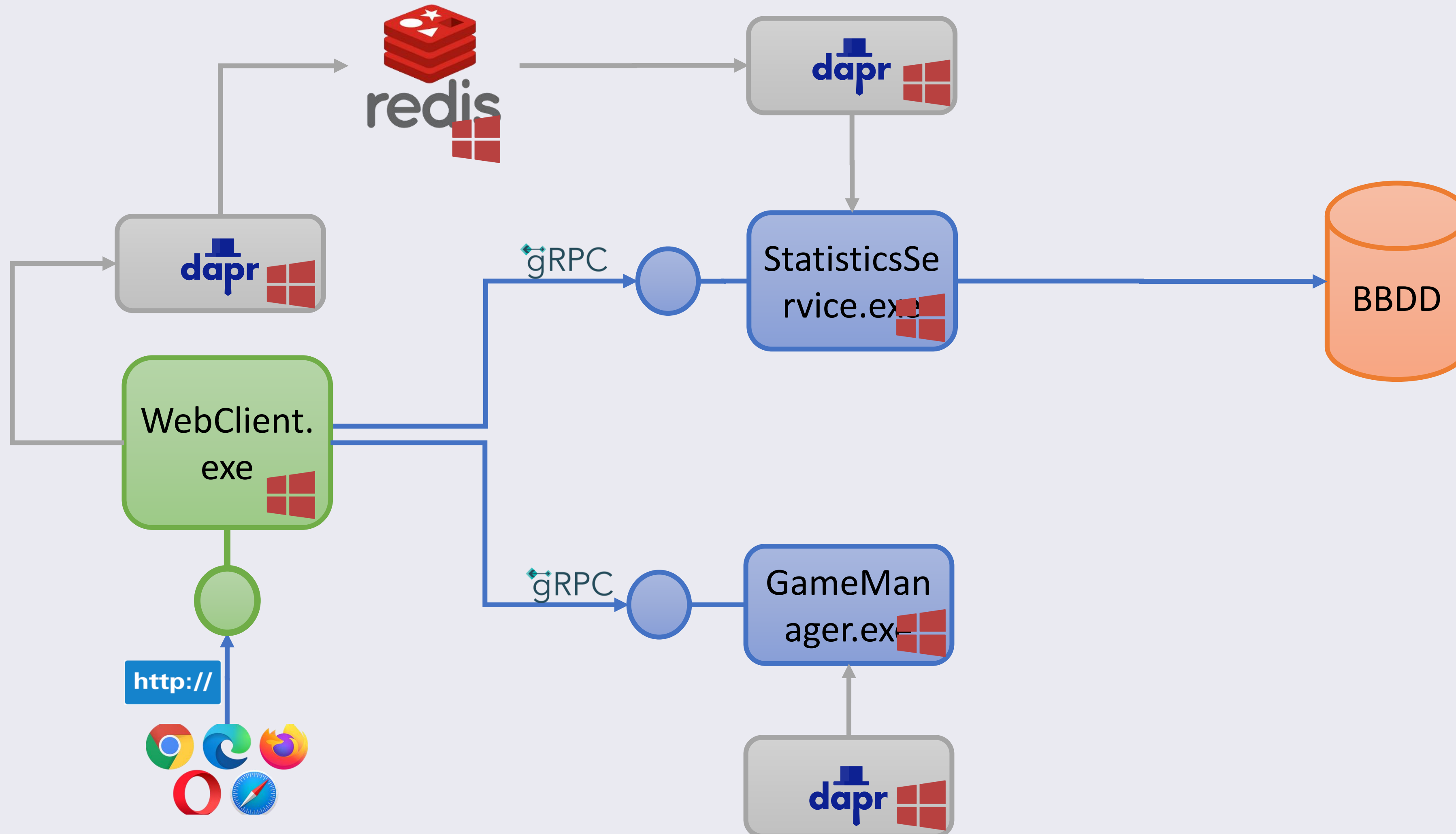## A Runtime to make easier build microservices

1. Technology agnostic (based on HTTP or gRPC APIs)

2. It's a **runtime** that must be installed alongside the application

3. Leverages the **sidecar pattern**

4. Exposes portable, technology independent, cloud independent and language independent ways to perform a large variety of tasks and patterns

Service-to-service invocation	State management	Publish and subscribe	Resource bindings and triggers	Actors	Observability	Secrets
Perform direct, secure, service-to-service method calls	Create long running, stateless and stateful services	Secure, scalable messaging between services	Trigger code through events from a large array of inputs. Output bindings to external resources including databases and queues	Encapsulate code and data in reusable actor objects as a common microservices design pattern	See and measure the message calls across components and networked services	Securely access secrets from your application

DotNet**2020**

#DotNet**2020**

"

Dapr is the Swiss knife of Microservices development

Me, right now ☺

What is gRPC? Why gRPC?

RPC implementation over HTTP/2 and protobuf

Language agnostic

Small payload (protobuf binary serialization)

Mandatory contract (.proto file)

Streaming support (client, server and bidirectional)

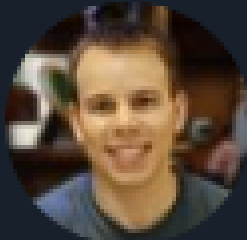No direct browser support (requires grpc-web)

gRPC and TLS

TLS is **not mandatory**, however is highly recommended

Without TLS is **not** possible to serve gRPC clients and HTTP clients through same port, due to lack of ALPN

Without TLS there is no way to promote from HTTP/1 to HTTP/2.

Supporting HTTP/2 requires the use of *prior knowledge*

Netcore gRPC client supports *prior knowledge*, but no browser support it. Other standard http clients may not support it also.

James Newton-King 👑 @JamesNK · 6h

I wrote a Blazor WebAssembly app that shows the performance benefits of gRPC-Web compared to JSON.

Both are fast with small payloads. With large data sets:
• gRPC network usage is 70% smaller
• gRPC deserialization is 10 times faster

Check it out here: grpcblazorperf.azurewebsites.net

https://twitter.com/JamesNK/status/1310875638585204738

YARP

Yet Another Reverse Proxy

YARP... What *really* is?

YARP is a **library** to help building customized reverse proxies easily

In the future will be a single executable too

Using YARP you can create your own ASP.NET Core projects to act as a reverse proxy over other services with little effort

Good for scenarios like custom aggregators

Currently supports request & headers transformations, CORS, Authentication & Authorization, Session Affinity.

Can be configured by code or by a configuration file