Johnny Hooyberghs

Building Cloud Native applications with .NET and Azure

Agenda

08:00 - 09:00	Registration
09:00 - 10:45	Workshop part 1 (Theory)
10:45 - 11:15	Coffee Break
11:15 - 13:00	Workshop part 2 (.NET 7 & Containers)
13:00 - 13:45	Lunch
13:45 - 15:15	Workshop part 3 (AKS part I)
15:15 - 15:45	Coffee Break
15:45 - 17:00	Workshop part 4 (AKS part II)

Here's Johnny!

Johnny Hooyberghs

@djohnnieke

johnny.hooyberghs@involved.be

- Passionate Developer
- Principal Software Consultant/Architect (.NET)
- Microsoft MVP, Developer Technologies
- Operational Manager at Involved

Cloud Native

The Cloud Native Computing Foundation

Cloud Native Technologies

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, micro-services, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

The Cloud Native Computing Foundation seeks to drive adoption of this paradigm by fostering and sustaining an ecosystem of open source, vendor-neutral projects. We democratize state-of-the-art patterns to make these innovations accessible for everyone.


BORING

Cloud Native Technologies

Cloud native is microservices hosted in containers and/or serverless apps, that can run in multi-cloud environments and are managed by DevOps processes


OOOOOOOH!
THAT'S A BULLSHIT BINGO
imgflip.com

Cloud Native Technologies

Cloud native is microservices hosted in containers and/or serverless apps, that can run in multi-cloud environments and are managed by DevOps processes

CSharpWars & MySauna

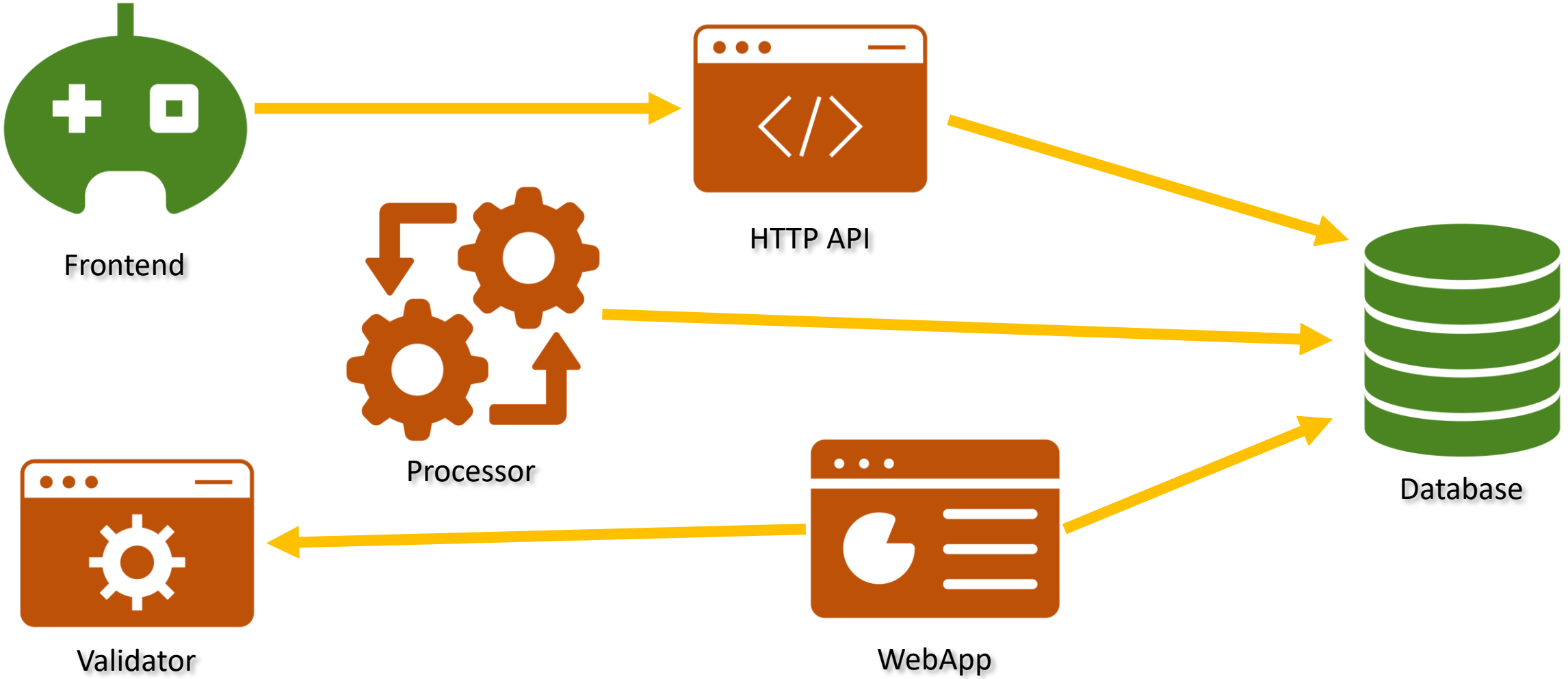
Why use a simplified sample app when you can use a real app ☺


geert (geert)
killer (djohnnie)
geert (geert)

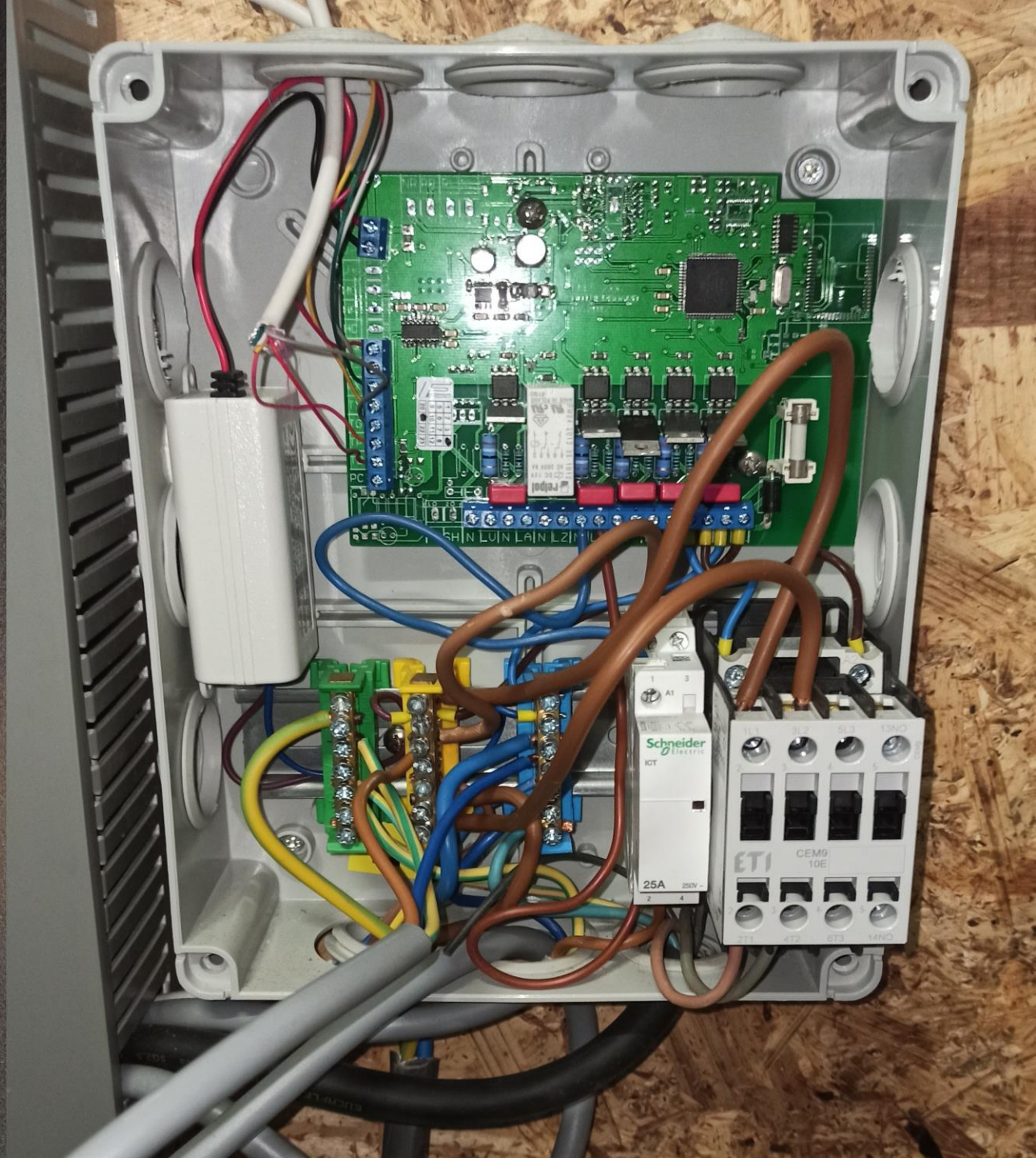
CSharpWars Robot Scripting


```
var step = LoadFromMemory<Int32>("STEP");
if( step % 3 == 0 )
{
    TurnLeft();
}
else
{
    WalkForward();
}
step++;
StoreInMemory<Int32>("STEP", step);
```

CSharpWars: Architecture

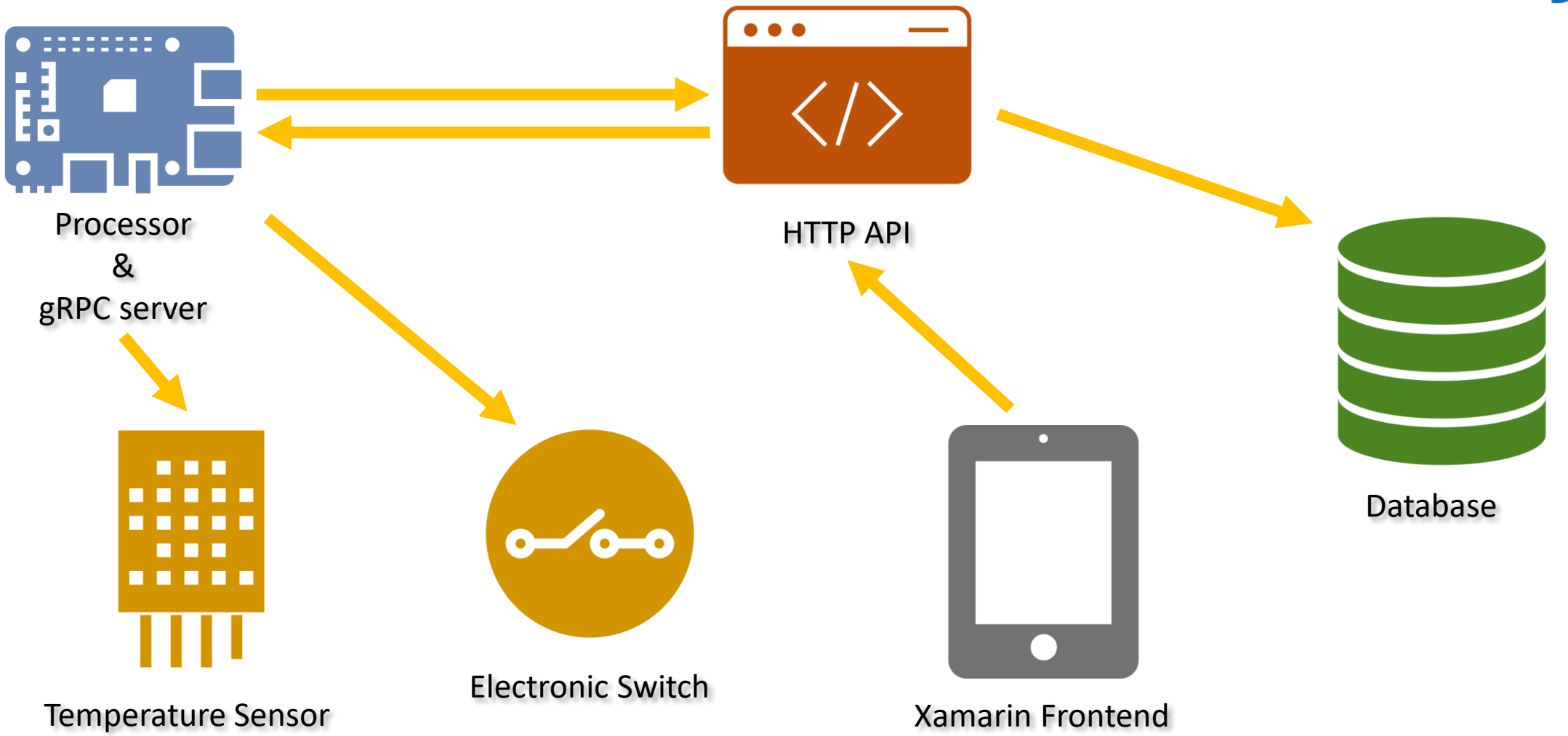



FES
MENU
ON
OK
OFF
STOP


Schneider
ICT
25A
ETI
CEM9
10E


zaterdag 2 mei 2020
17:02
sauna
106 °C
52:45
Lenovo

MySauna: Architecture



.NET

What can .NET do to help with cloud native

.NET vs .NET Framework

Platform independent

High performance

Very lightweight

Future-proof

Cloud Native compatible

The way to go for new apps

Backwards compatible

Restricted to Windows

Better Windows-integration (*)

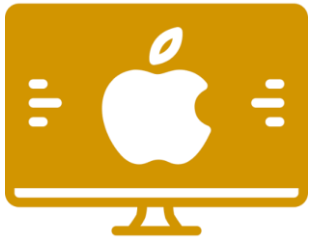
Cloud Native compatible

The way to go for legacy apps

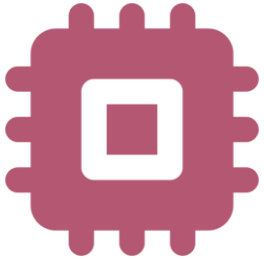
.NET is platform independent

Windows

Linux

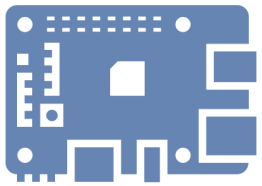


Mac



x86, x64, arm64

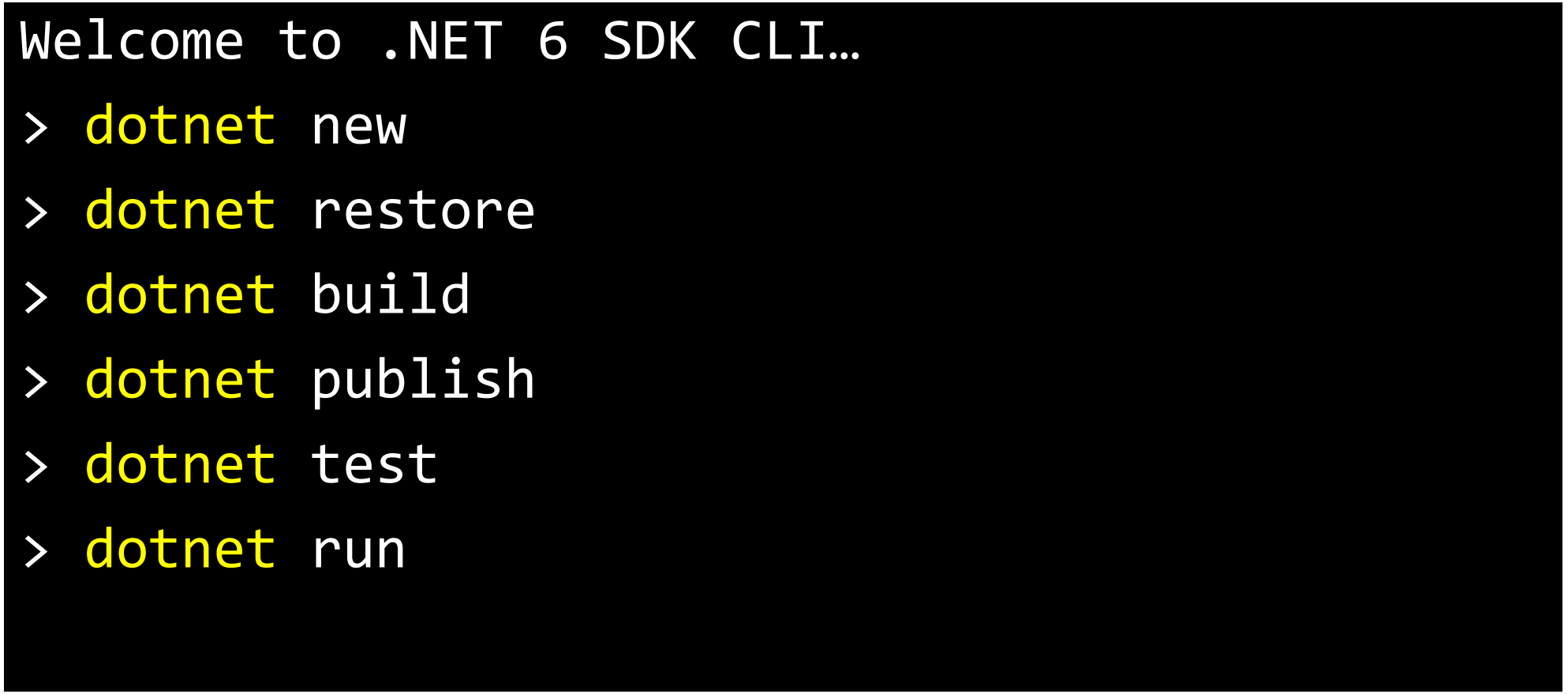
Cloud



IoT

Docker

.NET SDK is NOT bound by tools



Built-in Dependency Injection


```
[Route("[controller]")]
[ApiController]
public class ArenaController : ApiController<IArenaLogic>
{
    public ArenaController(IArenaLogic arenaLogic) : base(arenaLogic) { }

    // GET api/values
    [HttpGet]
    public Task<IActionResult> GetArena()
    {
        return Success(l => l.GetArena());
    }
}
```

Configuration

```
public class ArenaLogic : IArenaLogic
{
    private readonly IConfiguration _configuration;

    public ArenaLogic(IConfiguration configuration) {
        _configuration = configuration;
    }

    public Task<ArenaDto> GetArena() {
        return Task.FromResult(new ArenaDto {
            Width = _configuration.GetValue<int>("ARENA_SIZE"),
            Height = _configuration.GetValue<int>("ARENA_SIZE")
        });
    }
}
```

Logging

```
try {
    using var sw = new SimpleStopwatch();
    var middleware = scopedServiceProvider.GetService<IMiddleware>();
    await middleware.Process();
    _logger.LogInformation(
        "[ CSharpWars Script Processor - PROCESSING {ElapsedMilliseconds}ms! ]“
        , sw.ElapsedMilliseconds);
} catch (Exception ex) {
    _logger.LogError(ex,
        $"[ CSharpWars Script Processor - EXCEPTION - '{ex.Message}'! ]");
}
```

Microservices

A collection of loosely coupled services, where services are fine-grained, and protocols are lightweight

What are Microservices?

- Architectural style
- Distributed system
- Divide monolithical application into smaller applications
- Adds a communication layer in between these smaller applications
- Multiple (micro)services deliver the same functionality of the monolith
- End-user should not notice any difference


SAY MICROSERVICE
ONE MORE TIME
memegenerator.net

Why Microservices

- Increased performance
- Easier to pinpoint a performance bottleneck in the system
- Easier to scale out

Why Microservices

- Increased manageability
- Easier to upgrade part of the system in isolation
- Easier to do feature-updates
- Easier to pinpoint the culprit

Why Microservices

- Increased velocity
- Easier to scale out teams
- Teams can work in their technology or language of choice

Why Microservices

- Increased flexibility
- Easier to use different technologies
- Easier to use different programming languages

Should you use Microservices?

Large applications

- That (can) have clear defined boundaries
- That should be scalable and flexible

Large teams

- Than can work on different parts of the application in parallel
- To increase flexibility and velocity

Should you use Microservices?

Large applications

- That (can) have clear defined boundaries
- That should be scalable and flexible

Large teams

- Than can work on different parts of the application in parallel
- To increase flexibility and velocity

PROBABLY YES

Should you use Microservices?

- Small or lightweight applications
- Distributed applications are hard
- More chances of failing parts
- Harder to work together as a team
- More communication needed between different teams
- Difficult to define boundaries

Should you use Microservices?

- Small or lightweight applications
- Distributed applications are hard
- More chances of failing parts
- Harder to work together as a team
- More communication needed between different teams
- Difficult to define boundaries

PROBABLY NO


YOU CALLED MY SYSTEM A
MONOLITH

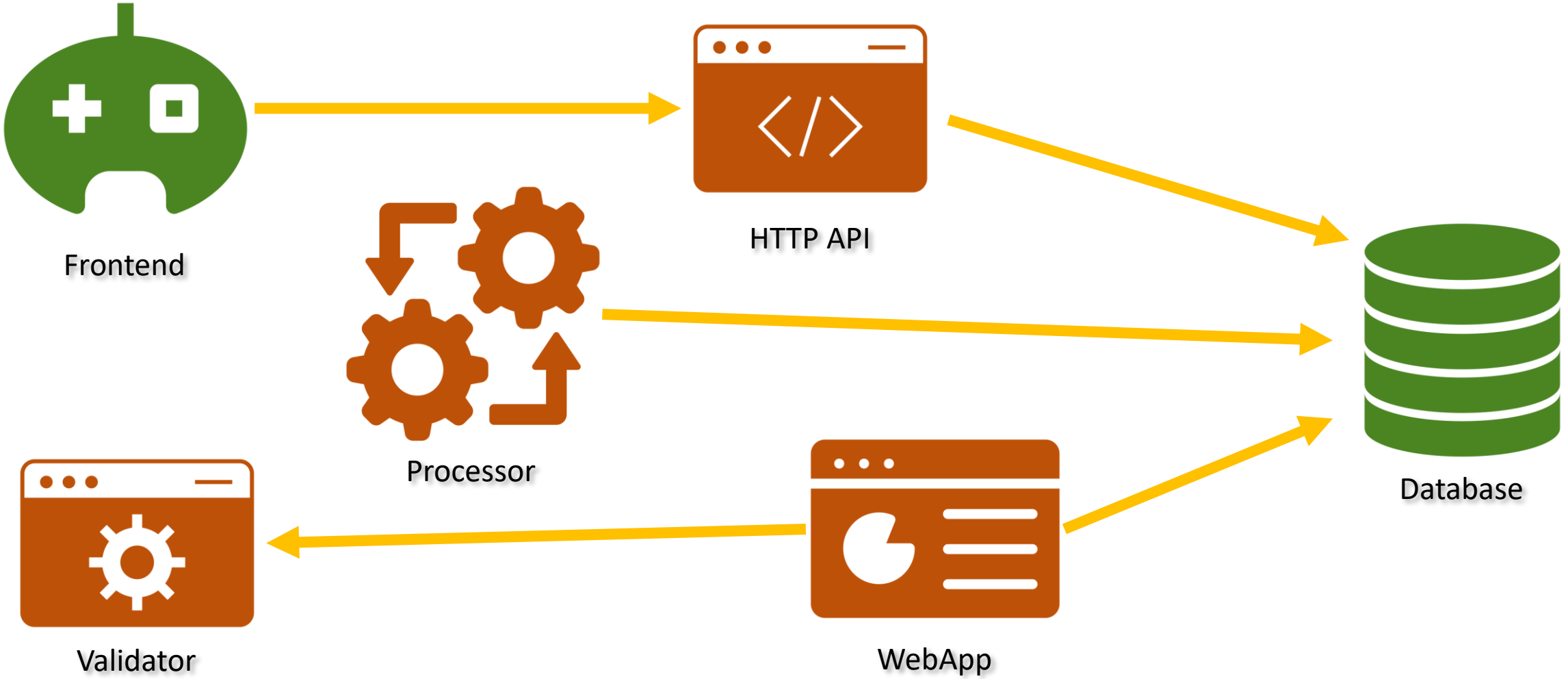
.NET to develop Microservices

- ASP.NET WebApi for HTTP & JSON based communication
- ASP.NET gRPC for HTTP/2 & binary based communication
- .NET Worker Services for background processing
- External messaging and pub/sub infrastructure & frameworks
- Azure Service Bus
- NServiceBus
- Dapr
- Orleans
- …

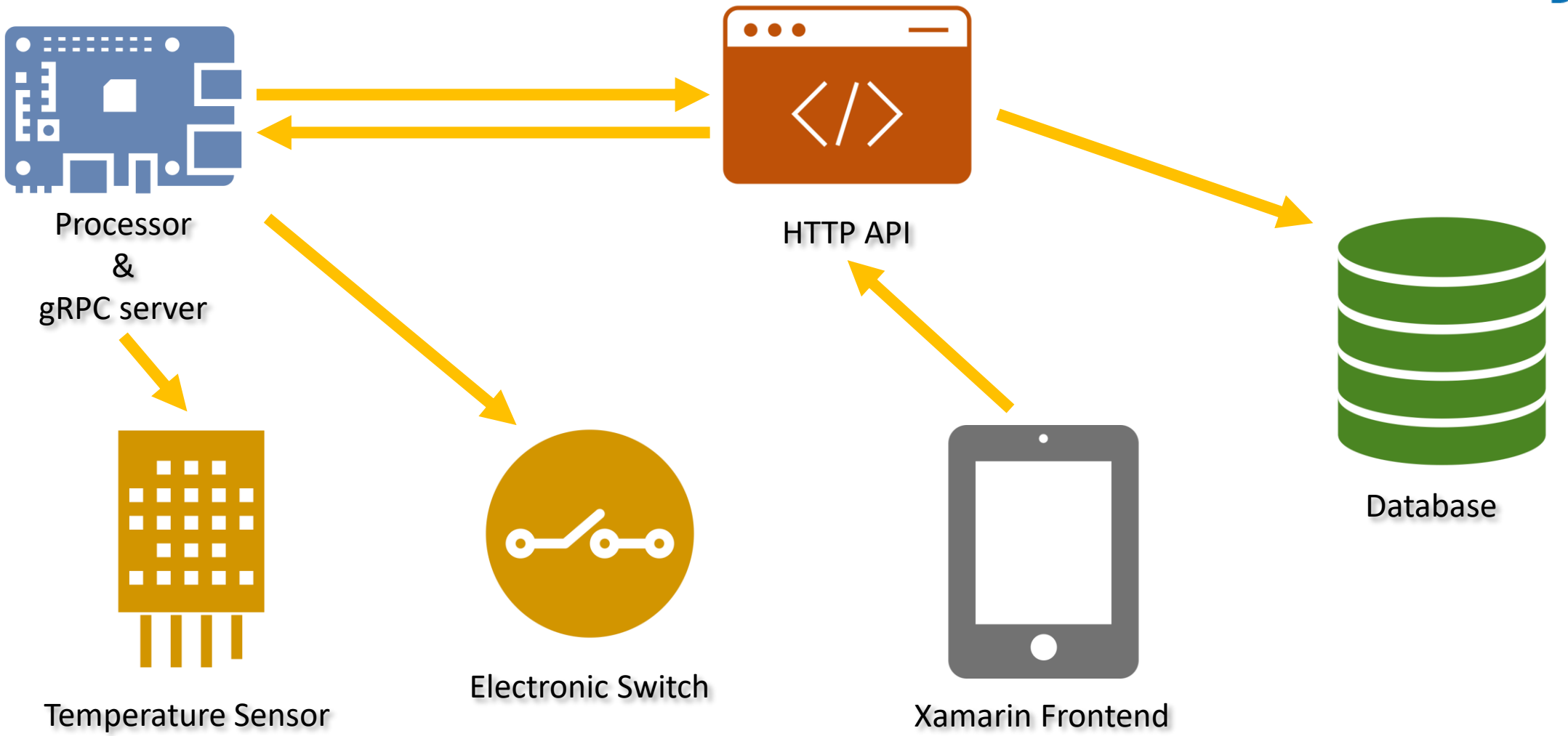
.NET 6 to host Microservices

- ASP.NET 5 has a built-in webserver called Kestrel
- Runs on Windows, Linux and Mac
- Runs on Raspberry Pi
- Runs on a potato
- Host in IIS
- Host in Azure App Service
- Host inside a (Docker) container (Windows or Linux)
- …

CSharpWars: Architecture



MySauna: Architecture



Containers

A standardized unit for developing, shipping and deploying a software package to run quickly and reliably, independently of computing environment

What are containers?

- OS-level virtualization
- Software packages
- Includes dependencies, libraries and configuration
- Isolated from one another
- Communication via well defined channels
- More lightweight than Virtual Machines
- Single operating system kernel, multiple containers
- Resource limiting


BONAVENTURE
KENDAL


tex

Building containers

```
FROM mcr.microsoft.com/dotnet/core/aspnet:3.1
WORKDIR /app
COPY bin/Release/publish /app
EXPOSE 5000
ENV TZ=Europe/Brussels
ENV KEY_VAULT=...
ENV CLIENT_ID=...
ENV CLIENT_SECRET=...
ENV CERTIFICATE_KEY=...
ENV ARENA_SIZE=10
ENTRYPOINT ["dotnet", "CSharpWars.Web.Api.dll"]
```

What are containers?

- Containers should not hold state!
- Use environment variables or volume mapping for configuration
- Use external caching services like Redis
- Use external storage services like databases

Serverless

Thanks to cloud computing, scaling, capacity planning and maintenance can be hidden from the developer or operator, focus on your application, not the infrastructure

What is Serverless?

- The cloud provider is responsible to execute your piece of code
- Resources can be allocated dynamically
- You are charged for the resources you need (have consumed)
- Run as stateless containers
- Triggered by a variety of events (http, queueing, jobs, …)
- Latency due to cold starts


SERVERLESS
IS MADE OF SERVERS

.NET and Serverless

- You can have a serverless compute experience and invoke a piece of .NET code by a trigger

- .NET is supported by:
- Azure Functions
- AWS Lambda

Multi-cloud

The use of multiple cloud computing and storage services in a single network architecture and the ability to be cloud-agnostic

Why Multi-cloud?

- Build apps that work across multiple cloud providers
- Avoid vendor lock-in
- Each provider has strenghts and weaknesses
- Achieve a level of resiliency that is not available on a single provider


CLOUD COMPUTING
CLOUD COMPUTING EVERYWHERE
memegenerator.net

Configuration

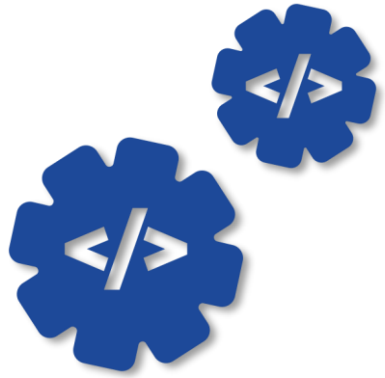
```
public static IHostBuilder CreateHostBuilder(string[] args) =>
  Host.CreateDefaultBuilder(args)
    .ConfigureWebHostDefaults(webBuilder => {
      webBuilder.ConfigureAppConfiguration(configBuilder => {
        var keyVault = GetEnvironmentVariable("KEY_VAULT");
        var clientId = GetEnvironmentVariable("CLIENT_ID");
        var clientSecret = GetEnvironmentVariable("CLIENT_SECRET");
        configBuilder.AddAzureKeyVault(keyVault, clientId, clientSecret);
      });
      webBuilder.ConfigureKestrel((ctx, options) => {
        var key = GetEnvironmentVariable("CERTIFICATE_KEY");
        var data = ctx.Configuration.GetValue<string>(key);
        var certificate = new X509Certificate2(Convert.FromBase64String(data));
        options.Listen(IPAddress.Any, 5000, listenOptions => {
          listenOptions.UseHttps(certificate); });
        }
      });
      webBuilder.UseStartup<Startup>();
});
```

Logging

```
public static IHostBuilder CreateHostBuilder(string[] args) =>
  Host.CreateDefaultBuilder(args)
       ConfigureLogging((hostContext, logging) => {
         var elasticUri = hostContext.Configuration.GetValue<string>("elastic-uri");
         if (!string.IsNullOrEmpty(elasticUri)) {
           Log.Logger = new LoggerConfiguration()
              .Enrich.FromLogContext()
              .Enrich.WithExceptionDetails()
              .WriteTo.Elasticsearch(new ElasticsearchSinkOptions(new Uri(elasticUri))
                {
                   AutoRegisterTemplate = true
                }).CreateLogger();
              logging.AddSerilog();
         }
       });
});
```

DevOps

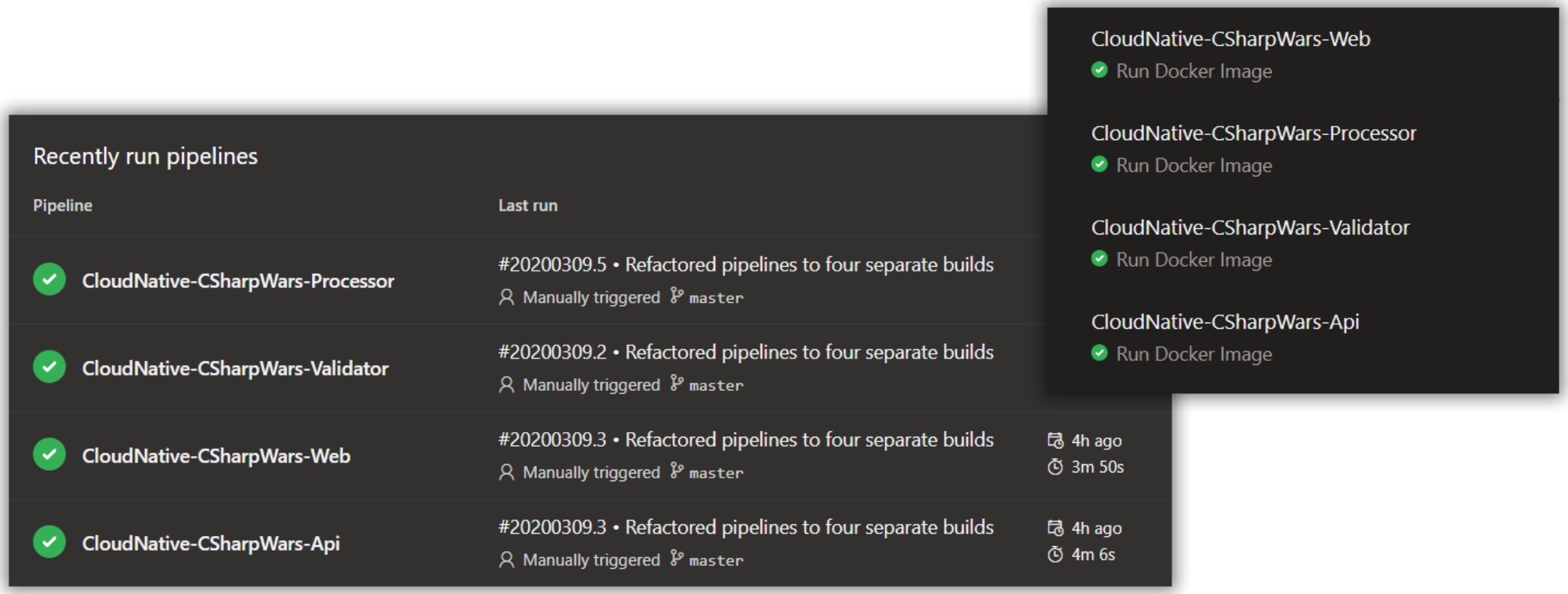
A set of practices that combines software development and IT-operations to shorten systems development lifecycle and provides continues delivery



Why DevOps?

- Working together
- Automation (with tools)
- Building
- Testing
- Deploying
- Updating and upgrading
- Scaling
- Monitoring
- Scripting (with tools)
- Configuration as code
- Source control!


WORKED FINE IN DEV
OPS PROBLEM NOW
memegenerator.net


Recently run pipelines
Pipeline
Last run
CloudNative-CSharpWars-Processor
#20200309.5 • Refactored pipelines to four separate builds
Manually triggered master
CloudNative-CSharpWars-Validator
#20200309.2 • Refactored pipelines to four separate builds
Manually triggered master
CloudNative-CSharpWars-Web
#20200309.3 • Refactored pipelines to four separate builds
Manually triggered master
4h ago
3m 50s
CloudNative-CSharpWars-Api
#20200309.3 • Refactored pipelines to four separate builds
Manually triggered master
4h ago
4m 6s
CloudNative-CSharpWars-Web
Run Docker Image
CloudNative-CSharpWars-Processor
Run Docker Image
CloudNative-CSharpWars-Validator
Run Docker Image
CloudNative-CSharpWars-Api
Run Docker Image

Thank you!

Questions?

- Johnny Hooyberghs
- @djohnnieke
- johnny.hooyberghs@involved.be

Please rate this session using

.NET DeveloperDays mobile app

(available on Google Play and AppStore)

Event Sponsors

Strategic Sponsors

avanade

Relativity®

Gold Sponsors

N-iX

<epam>

KMD
An NEC Company

HITACHI
Inspire the Next

Hitachi Energy

<vpTech/>

SCALO

Silver Sponsors

Softesco

grupa pracuj

Point72