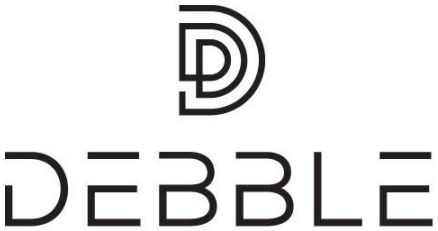
Automate your tasks through Azure Functions

Elio Struyf
Architect @ Valo – MVP
November 15th, 2018

Office 365 & SharePoint Connect

#O365Connect

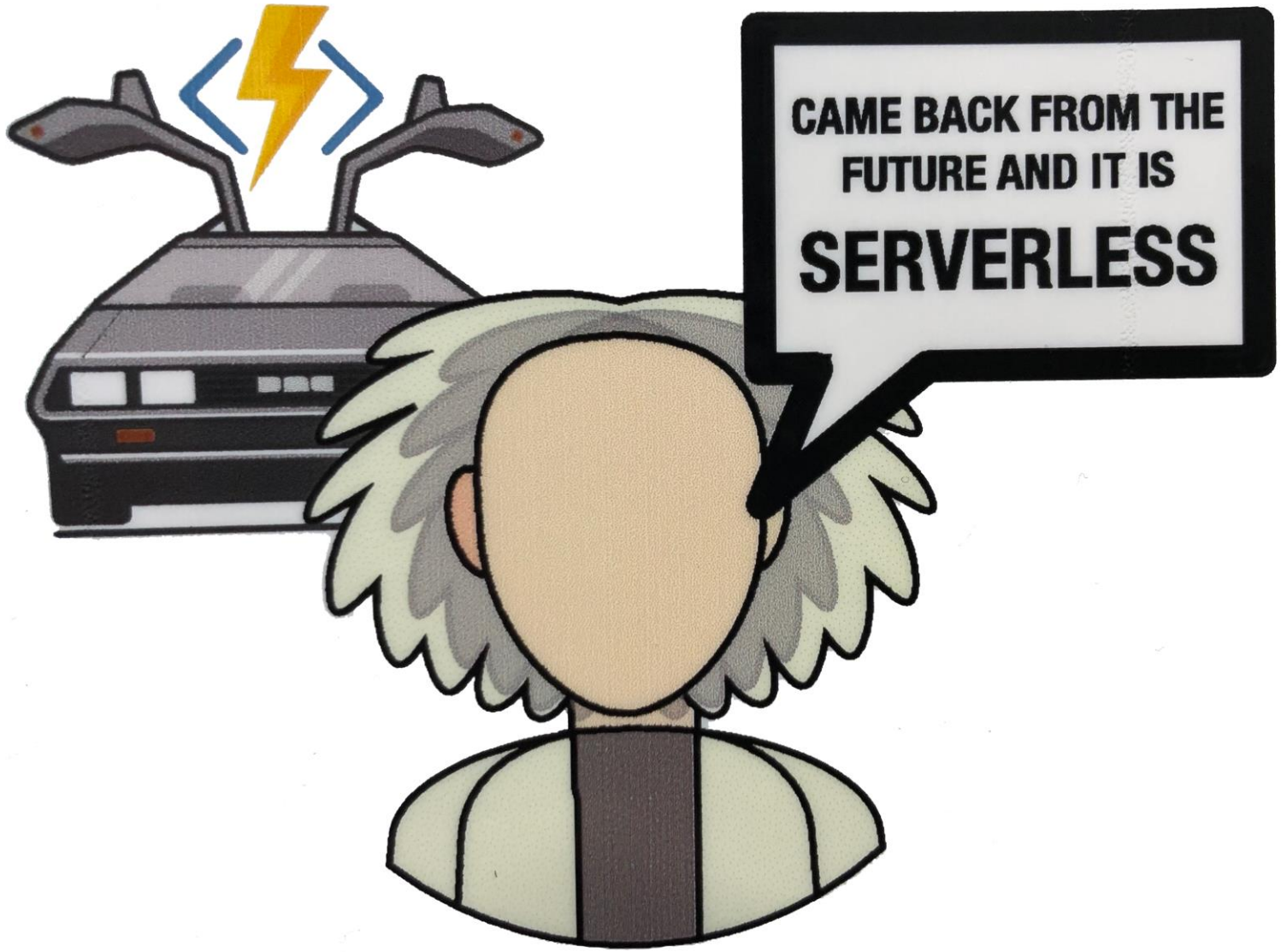



SERVERLESS


No hamsters are involved in the process of running your functions

What is Serverless?

- **Abstraction of servers**
- AKA: FaaS (Function as a Service)
- Event-driven / instant scale
- Pay-per-use
- Micro-services → decouple monolithic applications
- No permanent storage, or it has to live somewhere else


CAME BACK FROM THE
FUTURE AND IT IS
SERVERLESS

Azure Functions

- Running pieces of code / scripts in the cloud
- Use the language of your preference
- Runs on Azure App Services

Use cases

- Timer function
- Call from SPFx web part
- Call from site design / flow
- …

Azure Functions vs Web Jobs

Similarities

- Both can be scheduled or use a trigger (queue/blob/etc.)
- Both support C#, JavaScript, PS

Differences

- Pricing
- Development flexibility
- Easier to call Azure Functions (HTTP Trigger)
- It has its own UI in Azure
- Restrictions: 10 minutes timeout

Azure Functions – Pricing plans

Consumption plan

- € 0,17 per one million requests
- Scales up automatically
- Useful for smaller workloads
- Max. timeout of 10 minutes

host.json

```
{
  "functionTimeout": "00:10:00"
}
```

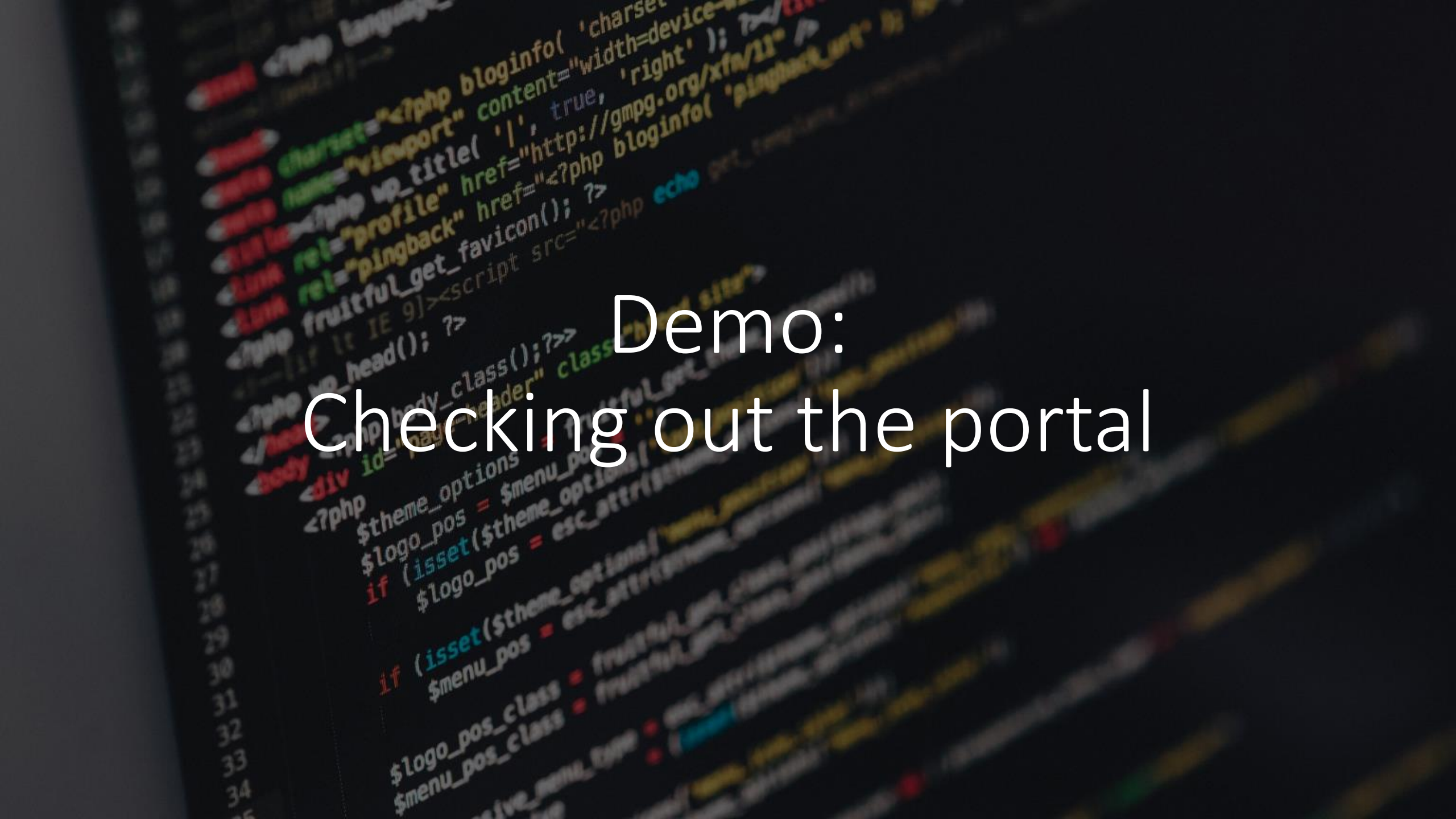
App Service Plan

- Depends on the chosen plan
- You’re in control of scaling
- Faster / higher workload
- Timeout: 30 min. (can be unlimited)

Turn on: always on!

What to know before you start creating them

- Functions should be **stateless**
- Function app can only use a **single language**
- Think about what **triggers** are required for your functions
- Need to call your functions from an application, try using **ngrok**
- Avoid **long** running operations / split them in multiple ones



Demo:
Checking out the portal


Security

Security options

Simple function authentication

- A code passed to the function
- On function level or app/admin level

https://<app-name>.azurewebsites.net/api/<function>?code=<code>

Azure AD Authentication

- Use OAuth token to call the function
- Great use in combination with SPFx solutions: *AadHttpClient*


Local development

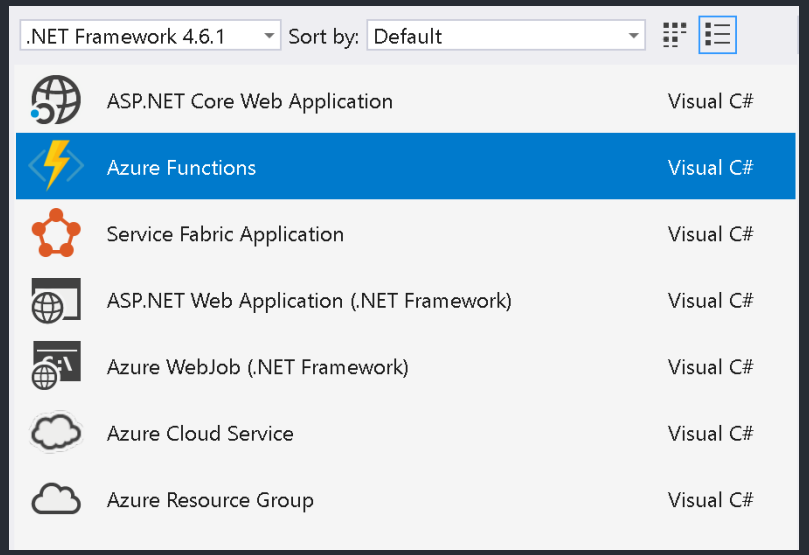
Local development options

Any stack / editor / platform

- azure-functions-core-tools
- Node.js

Visual Studio

- True C# functions
- Azure development workload



Useful CLI commands

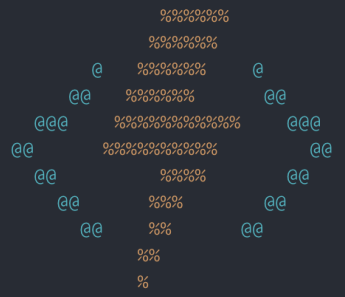
- func init → initialize a new project
- func new → create a new function
- func host start → start up the functions runtime

- func azure functionapp fetch-app-settings <function-name>
- func azure storage fetch-connection-string <storage-name>

Debugging your functions

- Run: *func host start*

```
11:18:33 ~/nodejs/training/azure-functions master ? 8.5.0 func host start
```



```
[27/10/2017 09:19:03] Reading host configuration file '/Users/eliostruyf/nodejs/training/azure-functions/host.json'
[27/10/2017 09:19:03] Host configuration file read:
[27/10/2017 09:19:03] { }
info: Worker.Node.1d30543d-3db7-4b60-b405-2d636d04d4ab[0]
      Start Process: node  --inspect=5858 "/Users/eliostruyf/.azurefunctions/bin/workers/Node/dist/src/nodejsWorker.js" --ho
st 127.0.0.1 --port 62410 --workerId 1d30543d-3db7-4b60-b405-2d636d04d4ab --requestId eed6f605-b33f-4960-84f6-4aff99ead374
info: Worker.Node.1d30543d-3db7-4b60-b405-2d636d04d4ab[0]
      Debugger listening on ws://127.0.0.1:5858/c0f1ab97-f7c2-4013-88be-6fda1189fcf4
info: Worker.Node.1d30543d-3db7-4b60-b405-2d636d04d4ab[0]
      For help see https://nodejs.org/en/docs/inspector
```

- Press F5 in Visual Studio Code and execute your function

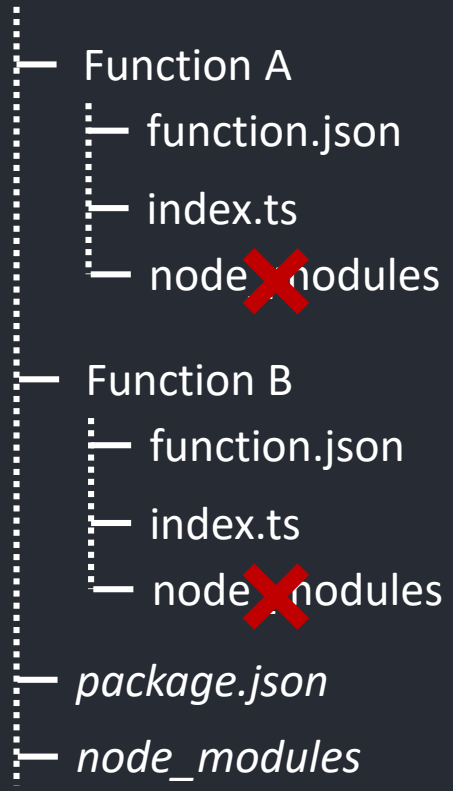
Demo:
Writing your first Azure Function


Dependencies

Importing dependencies

- Node.js → npm install
  - Make sure you create a **package.json** file (npm init)
  - Best to be placed on the root level

- C#
  - Use **NuGet** like you are used using it



Working with environment variables

Working with environment variables

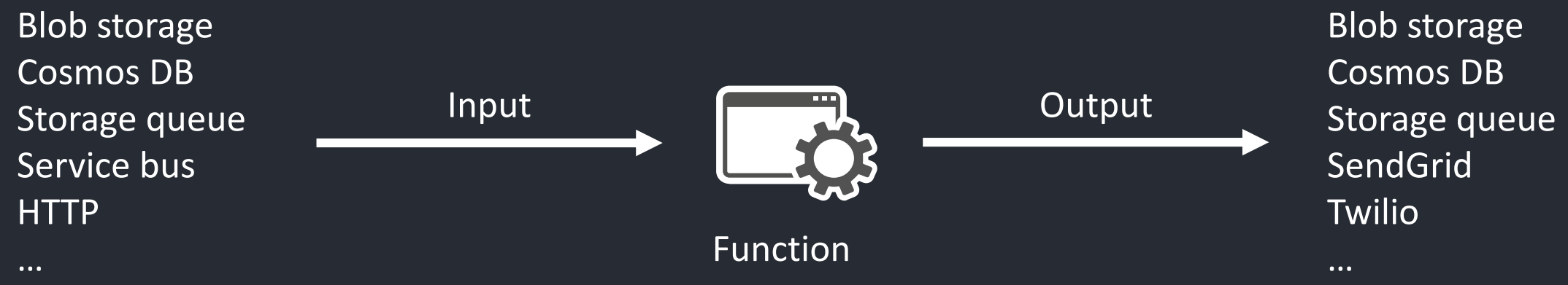
- Local development: local.settings.json
- On Azure: Application settings

In code

- Node.js
  - process.env.<setting-name>
- C#
  - GetEnvironmentVariable("<setting-name>")


Bindings

Working with bindings



Queues are great for…

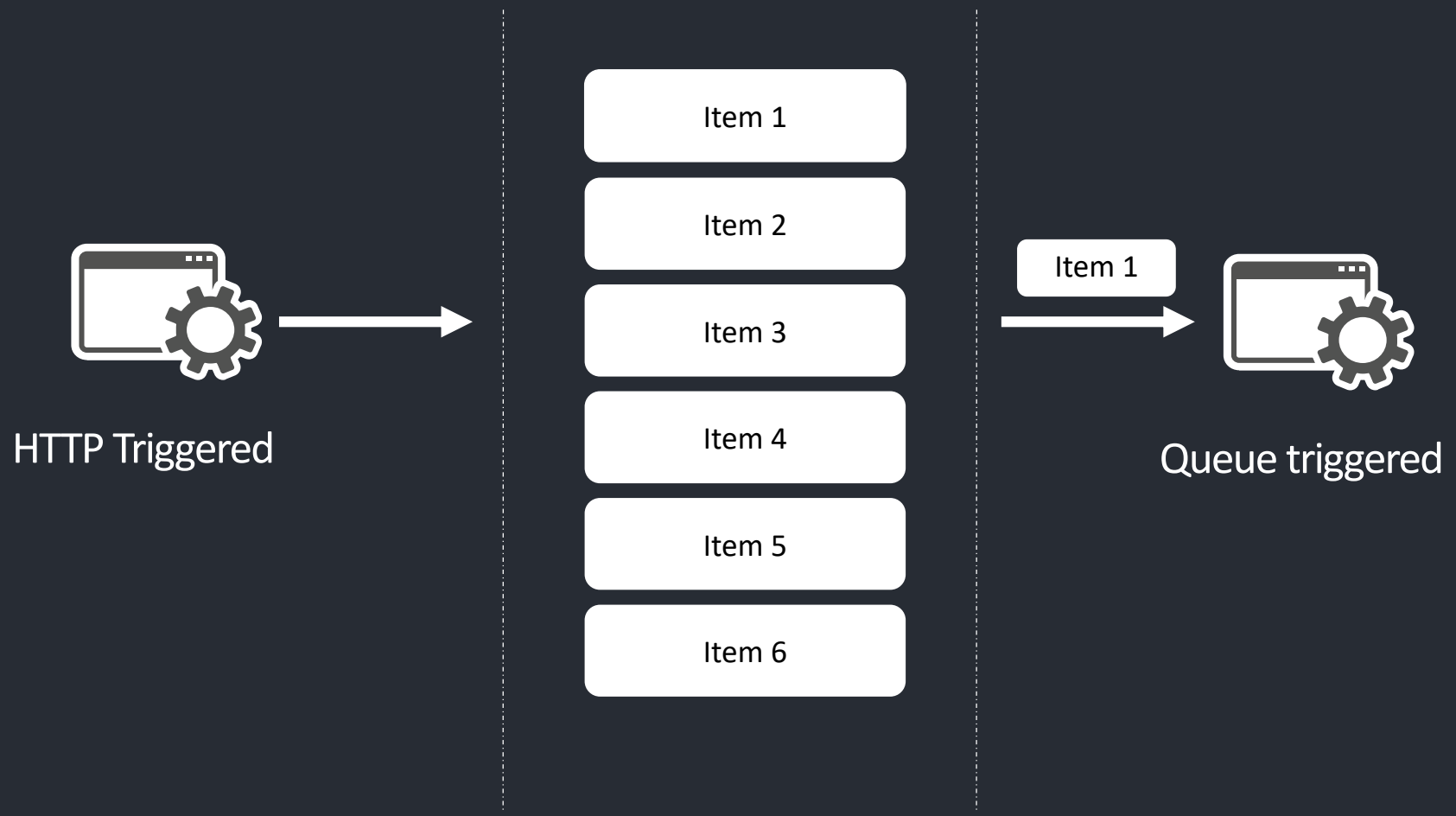
Anything that should be picked up by a longer running task

- SharePoint site provisioning
- SharePoint web hooks
- File processing (e.g. image size optimization)

QueueTrigger function

- Auto-runs your code when a message gets added
- If fails, it retries

Queues are great for…



func extensions install

Working with bindings

Add your bindings to the **function.json** file

```
{
  "disabled": false,
  "bindings": [{
    "authLevel": "function",
    "type": "httpTrigger",
    "direction": "in",
    "name": "req"
  },
  {
    "type": "queue",
    "name": "spfxversions",
    "direction": "out",
    "queueName": "versions",
    "connection": "AzureWebJobsStorage"
  }
}
```

Demo:
Creating settings and bindings

Deployment

Deployment options

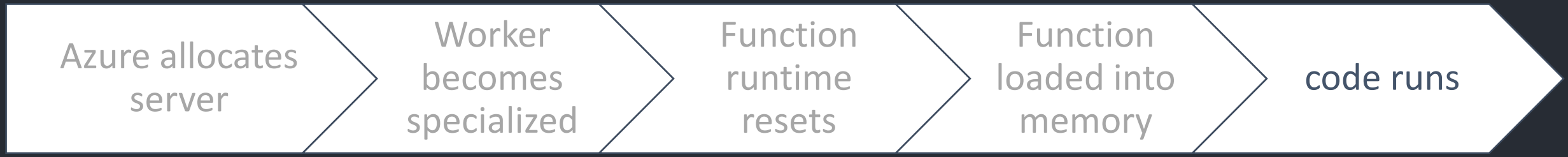
- Publish from Visual Studio
- CLI: func azure functionapp publish <function-name>
- Use local GIT repository
- Continuous deployment via Azure DevOps

Do not forget to configure your settings


Cold start

What happens during a cold start

When function is already warmed-up



Optimizing the cold start

- KISS
- Precompile C# created functions
  - Not use the *.csx files
  - Use Visual Studio + Azure development workload
- Use Azure Functions Pack for Node.js
  - https://github.com/Azure/azure-functions-pack
  - Uses Webpack
  - Add your dependencies at root level of the host!

Demo:
Deployment via local GIT repo


Troubleshooting

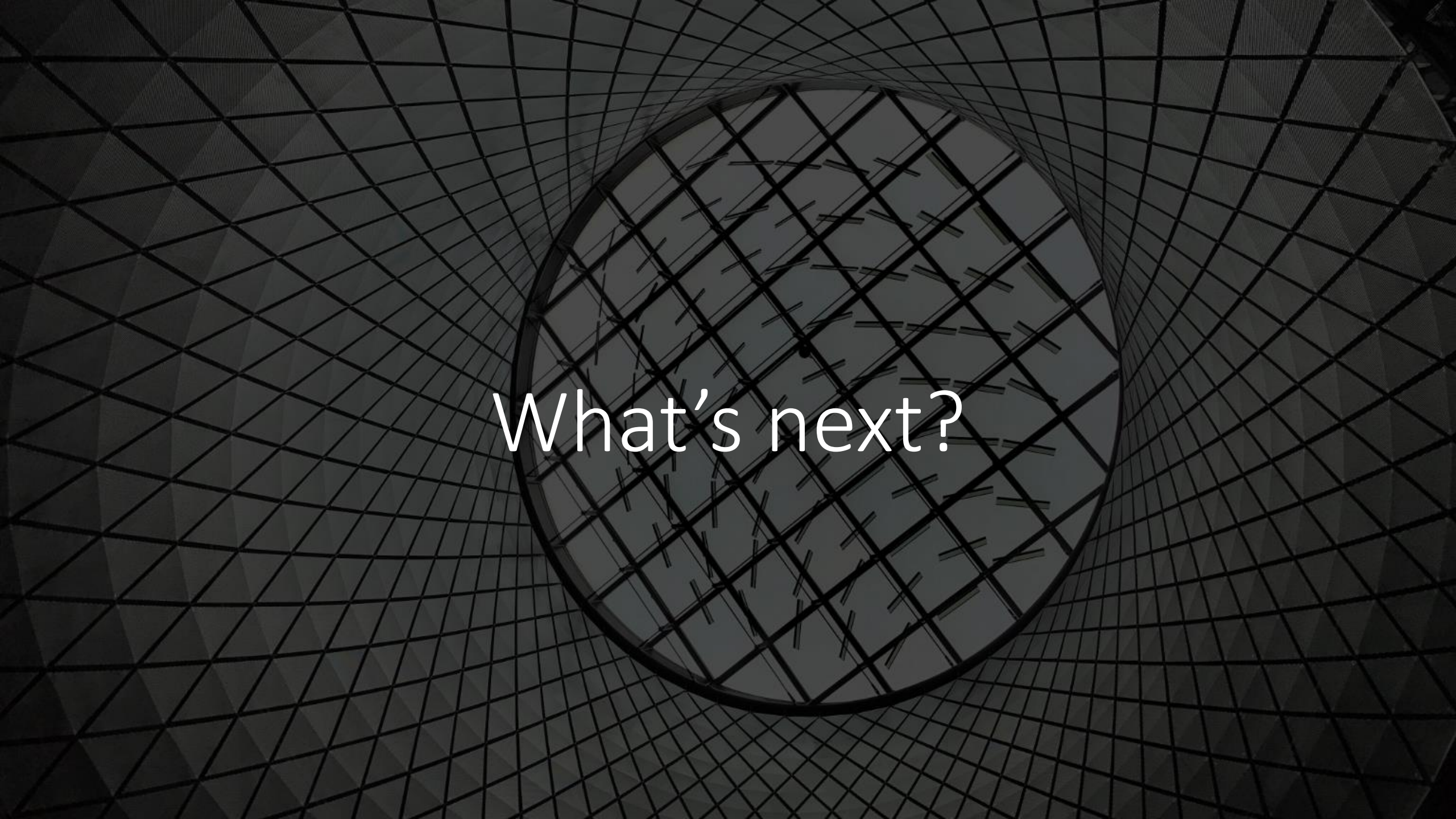
Troubleshooting

Check the linked Application Insights

Check Kudu (advanced tools)

Push debug build (only in **DEV** - slots) & attach the VS remote debugger

Try to reproduce it in your local version


What’s next?

Durable functions

- Extension on top of Azure Functions and WebJobs
- Write stateful functions
- Waiting for human interaction

Questions?

Elio Struyf

Developer & Architect

BIWUG

MVP Microsoft® Most Valuable Professional

PnP

@eliostruyf

www.eliostruyf.com

info@estruyf.be

Office Servers & Services MVP
Azure / Office 365 / SharePoint


Slides
https://www.eliostruyf.com/speaking
YEOMAN
Visual Studio