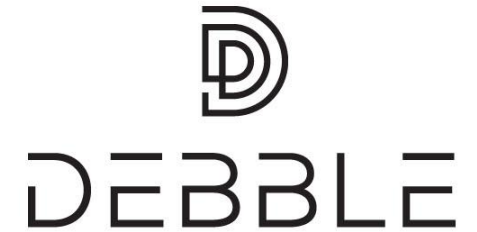





COMMUNITY

Automate your tasks through Azure Functions

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



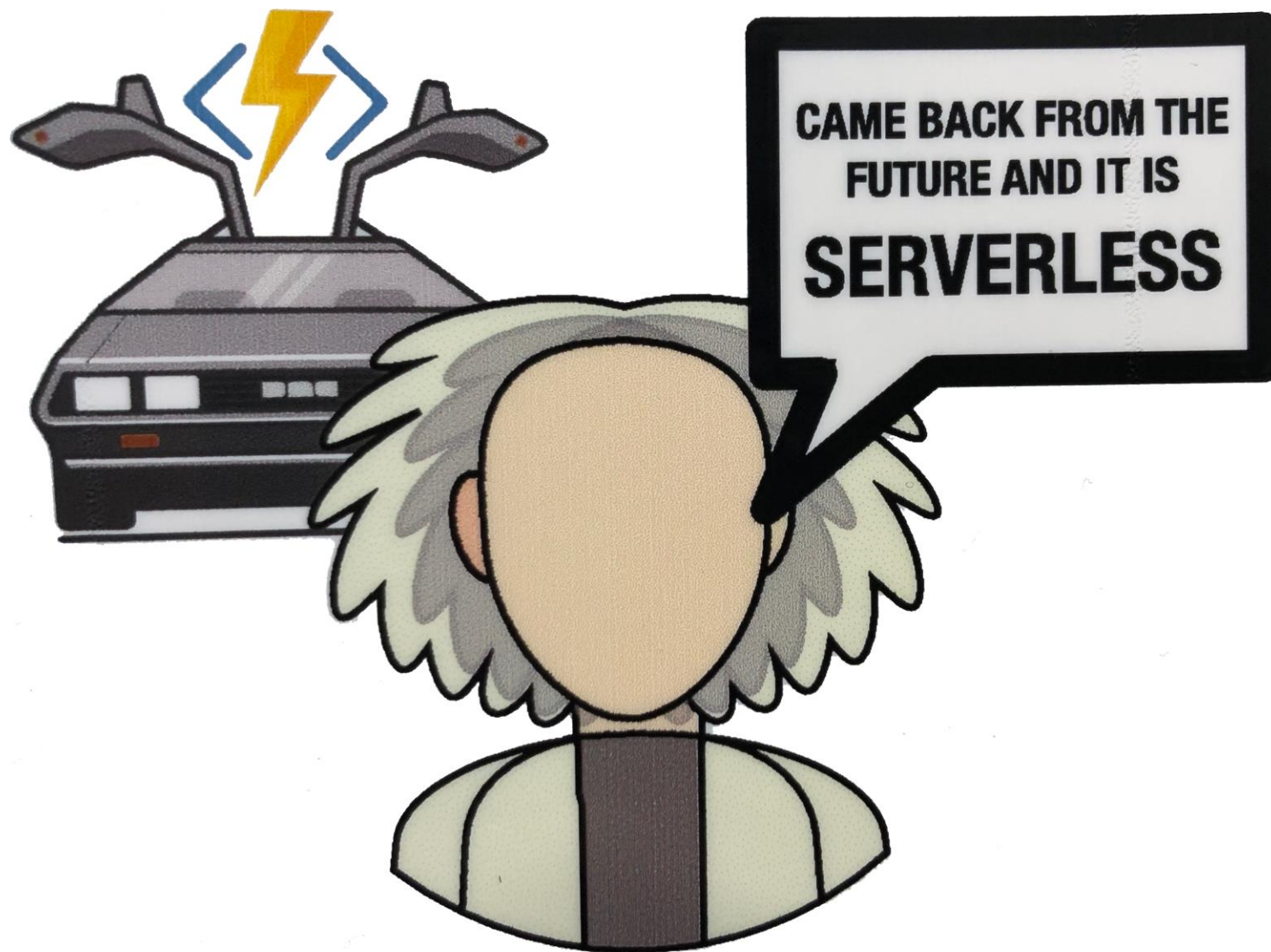
SERVER  *LESS*

A close-up photograph of a small, light-colored hamster, possibly a dwarf hamster, with its paws near its mouth. The hamster has a pinkish nose and dark eyes. The background is blurred, showing a light-colored surface and some indistinct shapes.

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



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

A dark brick wall is covered in a grid of security cameras. The cameras are arranged in a regular pattern, with some pointing upwards and others downwards. A vertical rectangular window is located in the upper center of the wall. At the bottom of the wall, two women are standing on a concrete ledge, looking up at the wall. The woman on the left is wearing a dark jacket and black pants, while the woman on the right is wearing a brown jacket and dark pants. The word "Security" is written in white text across the middle of the wall.

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`*

A blurred background image of a laptop screen displaying PHP code and a keyboard. The code on the screen includes comments and function calls like `wp_get_attachment_image` and `esc_attr`. The text "Local development" is overlaid in the center in a white, sans-serif font.

Local development

Local development options

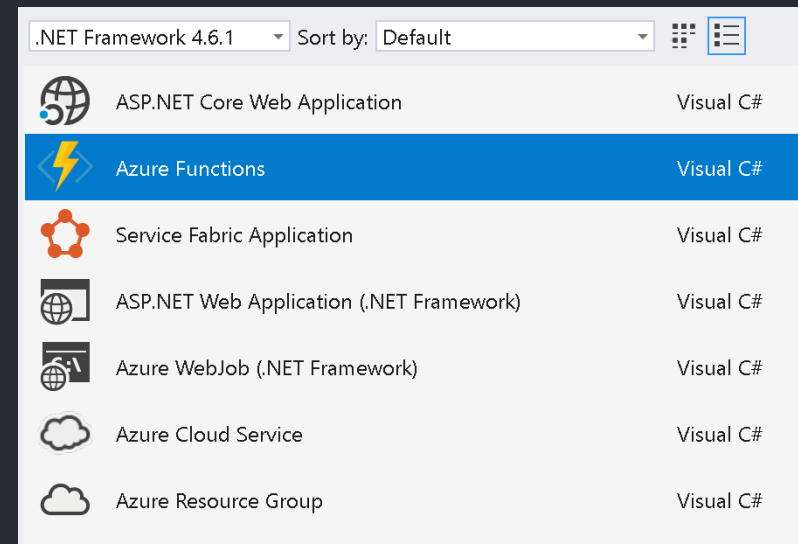
Any stack / editor / platform

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

[illegible]

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
```

- Press F5 in Visual Studio Code and execute your function

Demo: Writing your first Azure Function

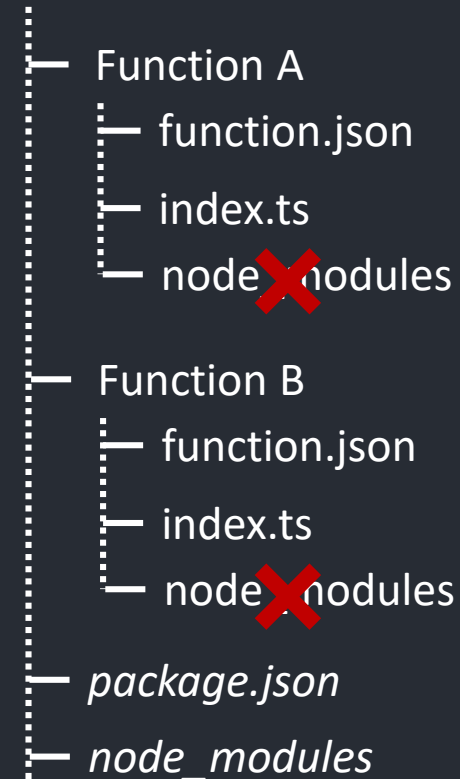
```
21718 scope.$watch(watchExpr, function ngSwitchMatchAction(value) {
21719     var i, ii;
21720     for (i = 0, ii = previousElements.length; i < ii; ++i) {
21721         previousElements[i].remove();
21722     }
21723     previousElements.length = 0;
21724
21725     for (i = 0, ii = selectedScopes.length; i < ii; ++i) {
21726         var selected = selectedElements[i];
21727         selectedScopes[i].$destroy();
21728         previousElements[i] = selected;
21729         $animate.leave(selected, function() {
21730             previousElements.splice(i, 1);
21731         });
21732     }
21733 }
```

Dependencies

```
21734 selectedElements.length = 0;
21735 selectedScopes.length = 0;
21736
21737 if ((selectedTranscludes = ngSwitchController.transcludes) != null) {
21738     scope.$eval(attr.change);
21739     forEach(selectedTranscludes, function(selectedTransclude) {
21740         var selectedScope = scope.$new();
21741         selectedScopes.push(selectedScope);
21742         selectedTransclude.transclude(selectedScope);
21743         var anchor = selectedScope.$anchor();
21744     });
21745 }
```

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



A low-angle, upward-looking photograph of several modern skyscrapers with glass facades. The buildings are dark and reflective, with some interior lights visible. The perspective creates a sense of height and architectural complexity. The text "Working with environment variables" is overlaid in the center in a white, sans-serif font.

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>")`

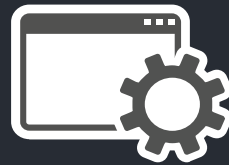


Bindings

Working with bindings

Blob storage
Cosmos DB
Storage queue
Service bus
HTTP
...

Input



Function

Output

Blob storage
Cosmos DB
Storage queue
SendGrid
Twilio
...

Queues are great for...

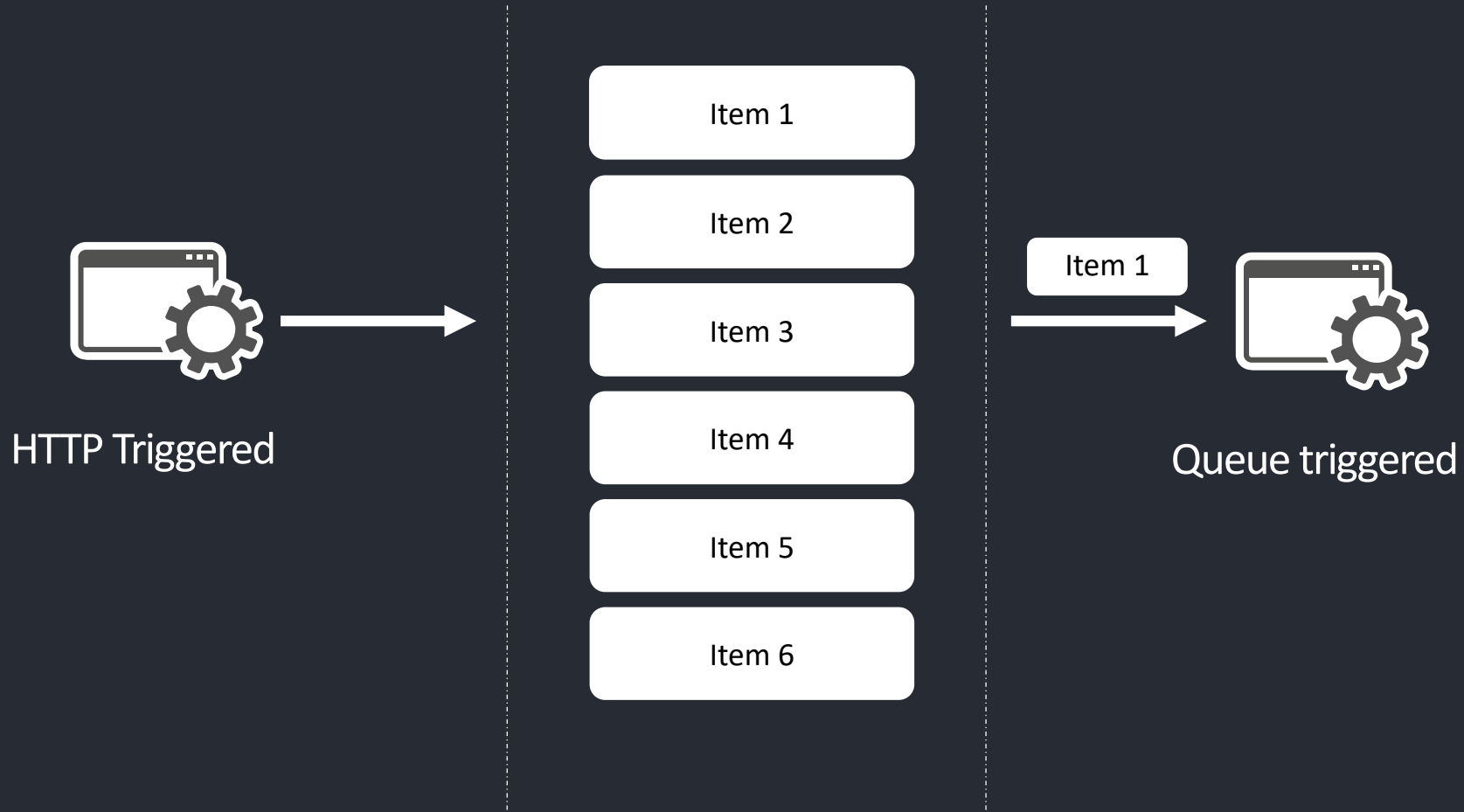
Anything that should be picked up by a longer running task

- 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

A photograph of a network cable with a damaged RJ45 connector. The white plastic housing of the connector is cracked and broken, revealing the internal wiring. Several colored wires (blue, green, brown) are exposed and frayed. The cable is set against a plain, light-colored background.

Deployment

Deployment options

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

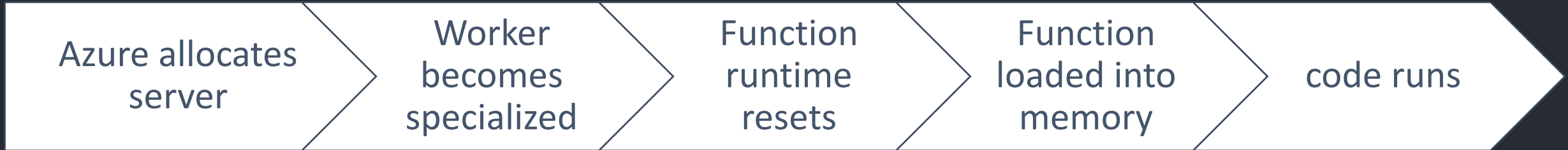
A red and white warning sign is attached to a chain-link fence. The sign features a large white exclamation mark inside a red triangle on the left, and the word 'DANGER' in bold white capital letters on a red background to the right. The sign is slightly tilted and appears to be secured with a metal fastener. The background is a blurred chain-link fence.

Do not forget to configure
your settings

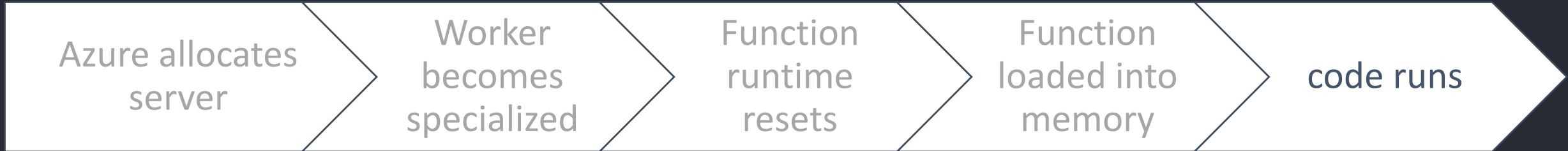
A dark, high-contrast photograph of a snow-capped mountain peak, likely Mount Everest, with the text "Cold start" overlaid in white. The image is in grayscale, emphasizing the textures of the snow and the jaggedness of the rock. The sky is dark and cloudy, adding to the cold and dramatic atmosphere.

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

```
1 <?php language=
2
3 <?php bloginfo( 'charset'
4
5 <?php bloginfo( 'charset'
6
7 <?php bloginfo( 'charset'
8
9 <?php bloginfo( 'charset'
10
11 <?php bloginfo( 'charset'
12
13 <?php bloginfo( 'charset'
14
15 <?php bloginfo( 'charset'
16
17 <?php bloginfo( 'charset'
18
19 <?php bloginfo( 'charset'
20
21 <?php bloginfo( 'charset'
22
23 <?php bloginfo( 'charset'
24
25 <?php bloginfo( 'charset'
26
27 <?php bloginfo( 'charset'
28
29 <?php bloginfo( 'charset'
30
31 <?php bloginfo( 'charset'
32
33 <?php bloginfo( 'charset'
34
35 <?php bloginfo( 'charset'
36
37 <?php bloginfo( 'charset'
38
39 <?php bloginfo( 'charset'
40
41 <?php bloginfo( 'charset'
42
43 <?php bloginfo( 'charset'
44
45 <?php bloginfo( 'charset'
46
47 <?php bloginfo( 'charset'
48
49 <?php bloginfo( 'charset'
50
51 <?php bloginfo( 'charset'
52
53 <?php bloginfo( 'charset'
54
55 <?php bloginfo( 'charset'
56
57 <?php bloginfo( 'charset'
58
59 <?php bloginfo( 'charset'
60
61 <?php bloginfo( 'charset'
62
63 <?php bloginfo( 'charset'
64
65 <?php bloginfo( 'charset'
66
67 <?php bloginfo( 'charset'
68
69 <?php bloginfo( 'charset'
70
71 <?php bloginfo( 'charset'
72
73 <?php bloginfo( 'charset'
74
75 <?php bloginfo( 'charset'
76
77 <?php bloginfo( 'charset'
78
79 <?php bloginfo( 'charset'
80
81 <?php bloginfo( 'charset'
82
83 <?php bloginfo( 'charset'
84
85 <?php bloginfo( 'charset'
86
87 <?php bloginfo( 'charset'
88
89 <?php bloginfo( 'charset'
90
91 <?php bloginfo( 'charset'
92
93 <?php bloginfo( 'charset'
94
95 <?php bloginfo( 'charset'
96
97 <?php bloginfo( 'charset'
98
99 <?php bloginfo( 'charset'
100
```

A person with dark, curly hair and black-rimmed glasses is shown in a close-up, looking down with a distressed expression. Their right hand is pressed against their forehead, and their face shows signs of crying. The background is a dimly lit room with a white sheet and a red patterned blanket visible. The word "Troubleshooting" is written in a white, sans-serif font across the center of the image.

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

The background is a dark, monochromatic image featuring a complex grid of thin, intersecting lines that create a sense of depth and perspective. A large, circular frame is centered in the image, acting as a focal point. Inside this frame, the grid pattern continues, but with some lines appearing slightly blurred or out of focus, suggesting a three-dimensional space. The overall tone is mysterious and contemplative.

What's next?

Durable functions

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



THANK YOU

Questions?



Elio Struyf

Developer & Architect



@eliostruyf



www.eliostruyf.com



info@estruyf.be

*Office Servers & Services MVP
Azure / Office 365 / SharePoint*

Slides

<https://www.eliostruyf.com/speaking>

