Swiss Cloud Native Day
2025

Supercharging Edge Event-Driven Architecture on Kubernetes with Wasm

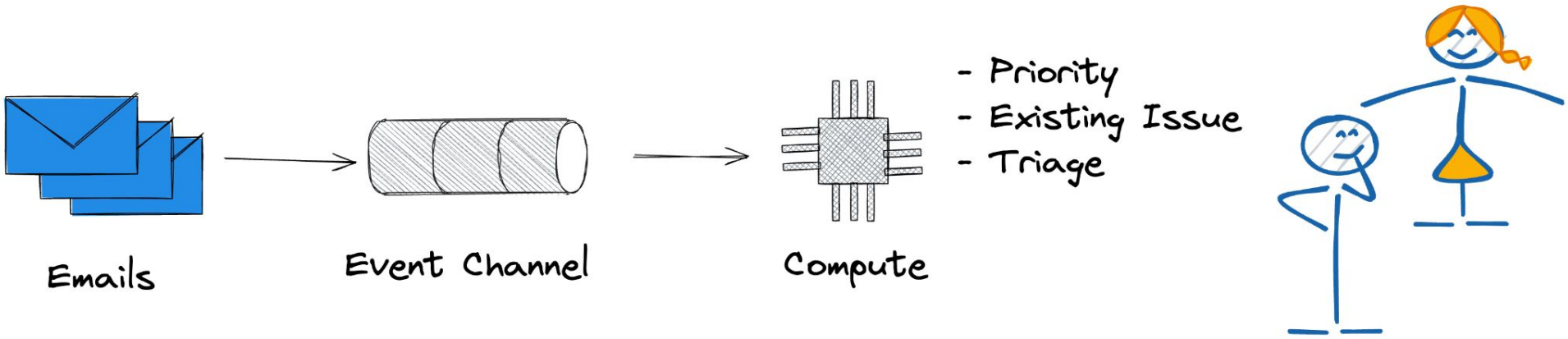
PRESENTER

Fabrizio Lazzaretti
Managing Consultant,
Wavestone

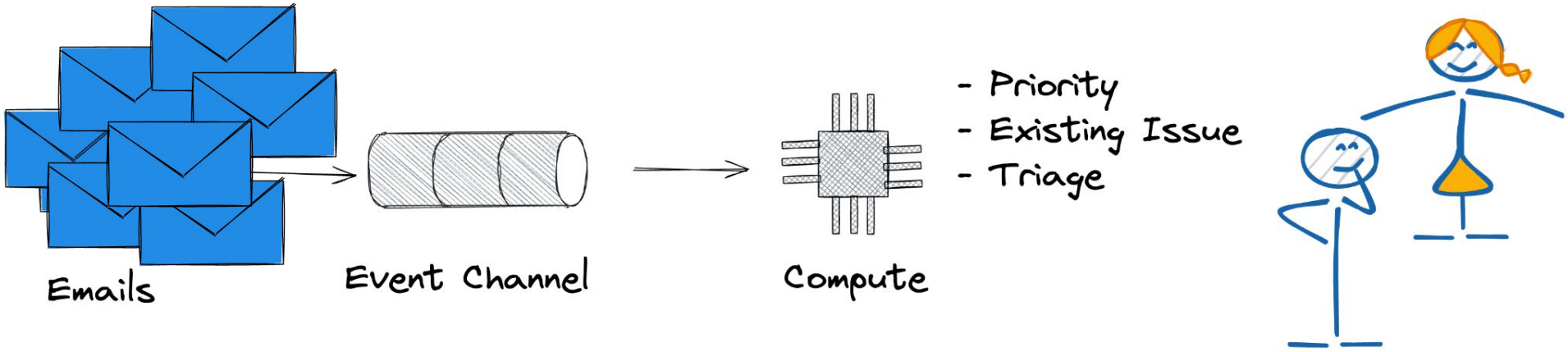
PRESENTER

Linus Basig
Head of Engineering,
CARU AG

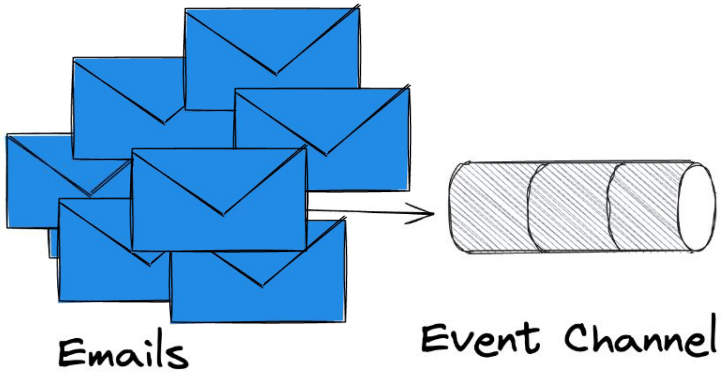
You got Mail!



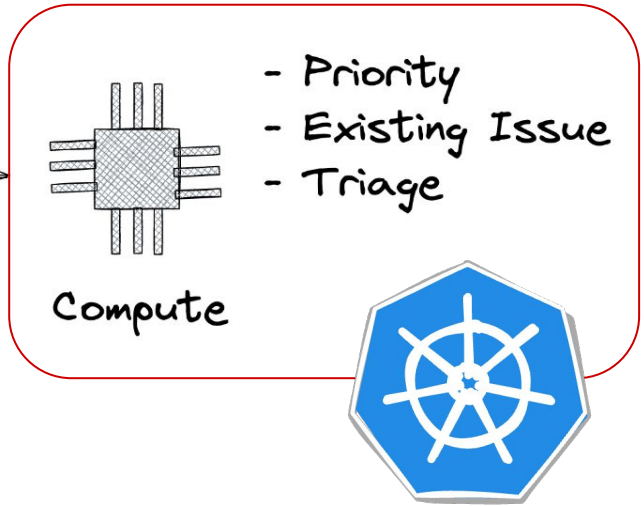
You got Mail!

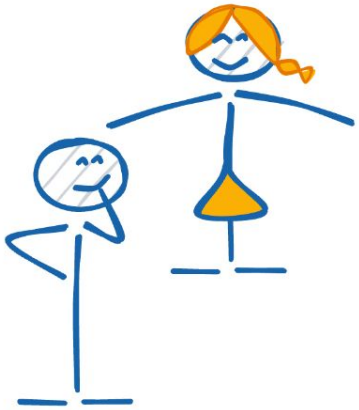


You got Mail!

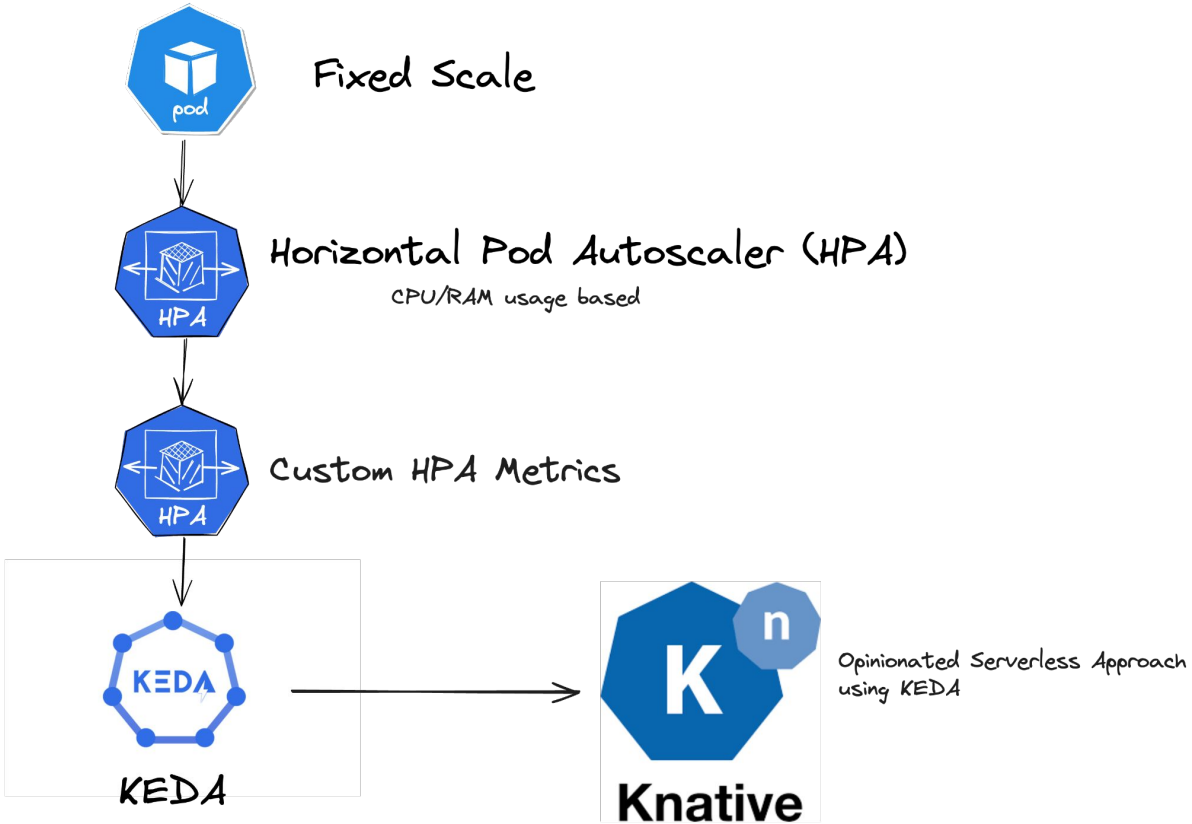


How do we scale?



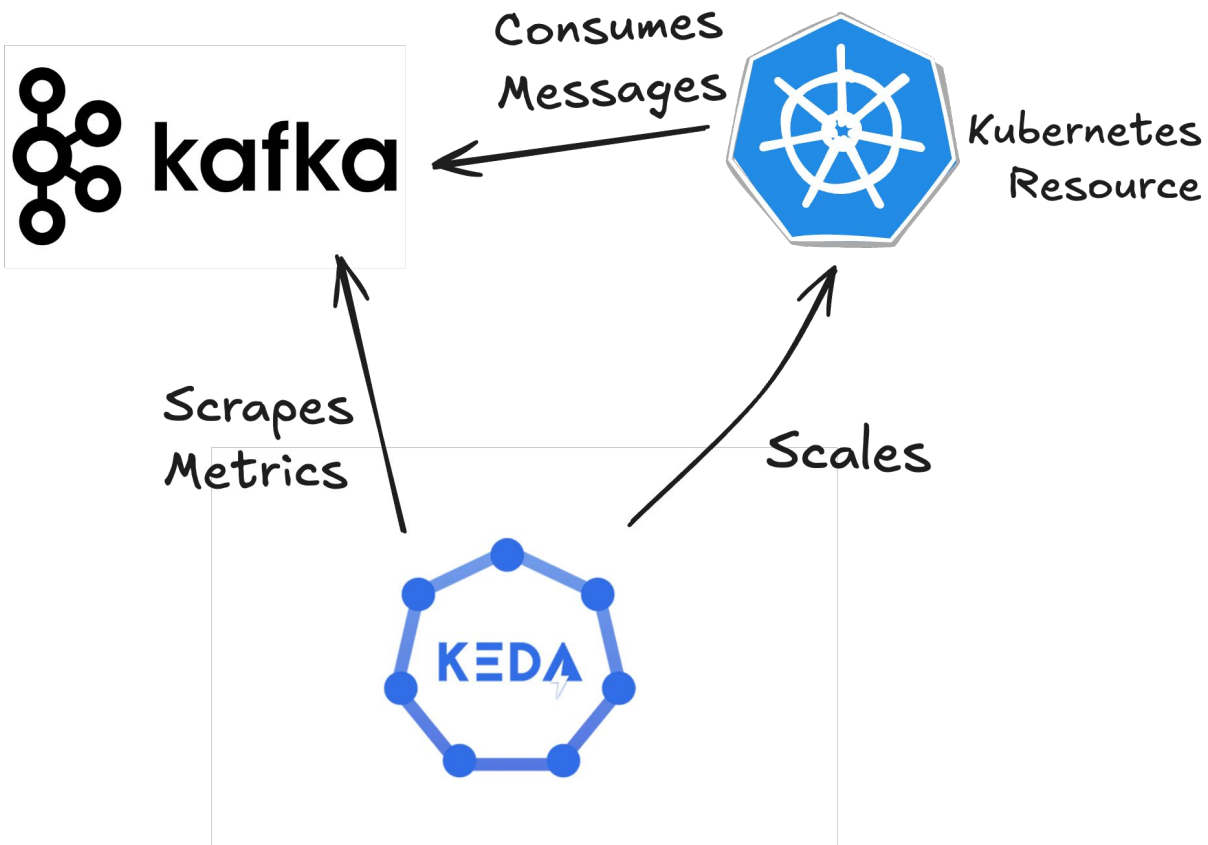


Scaling Event-Driven Architecture on Kubernetes



SWISS CLOUD NATIVE DAY

How does KEDA work?



Now that we have scaling

How do we make it fast?

DISCLAIMER: Cutting Edge

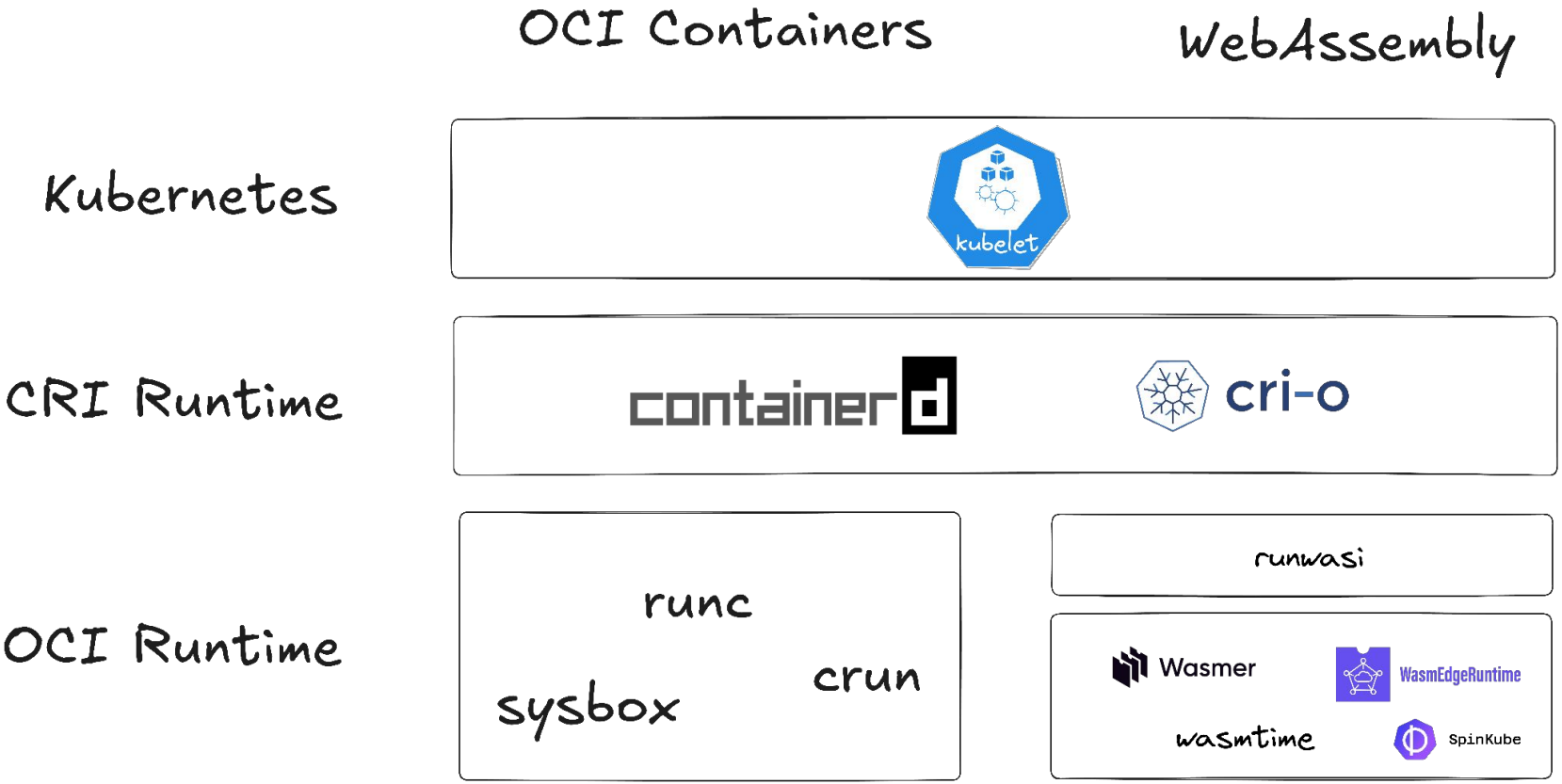
Working with WebAssembly on Kubernetes may lead to unexpected emotional distress. Side effects may include, but are not limited to:

- Frustration from complex configurations
- Anxiety due to compilation issues
- Despair when debugging across multiple layers of abstraction
- Euphoria upon successful deployment (followed by crashes)

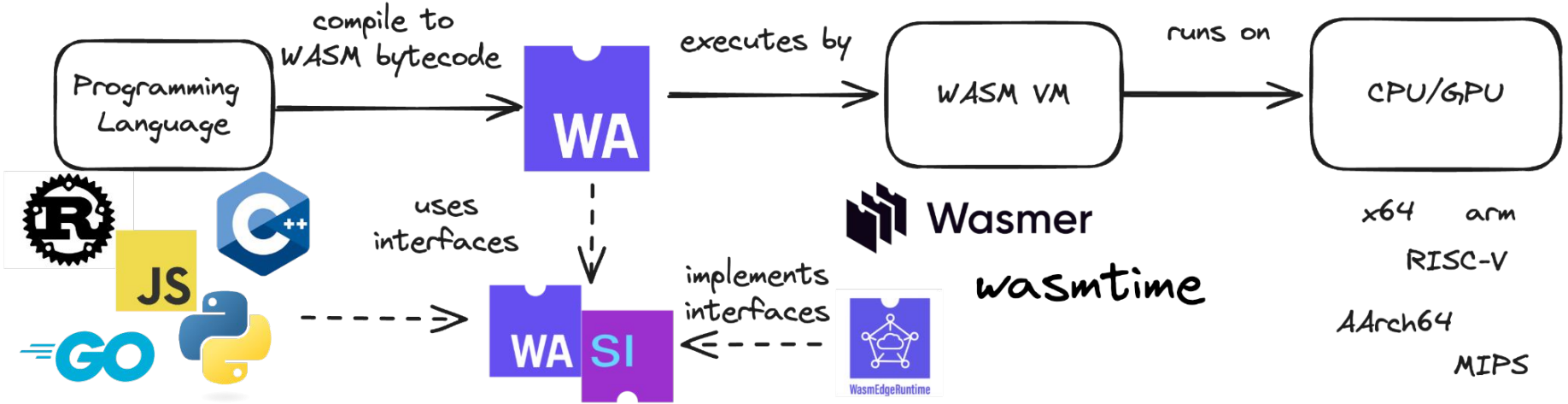
Users are advised to proceed with caution and maintain a sense of humor. Remember: it's just code, not a reflection of your worth as a human being.

Consult your nearest DevOps therapist if symptoms persist for more than two sprints.

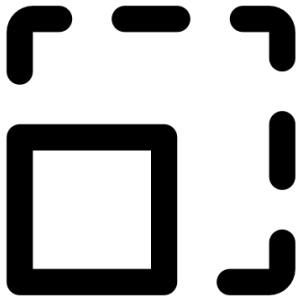
WebAssembly on Kubernetes



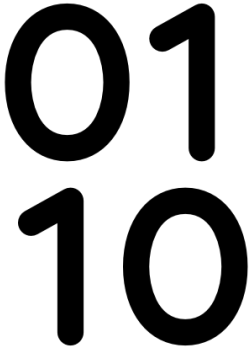
WebAssembly in a nutshell



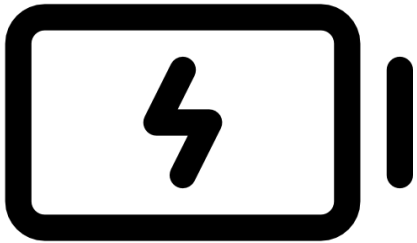
How does WebAssembly (Wasm) help with speed?



Tiny Images

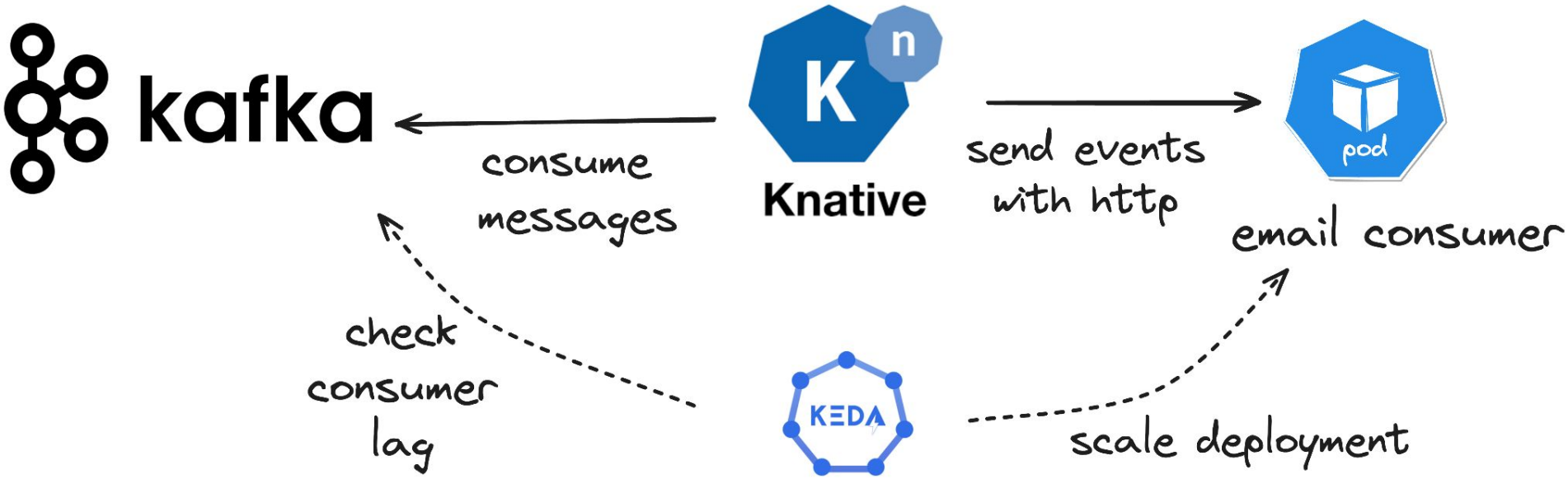


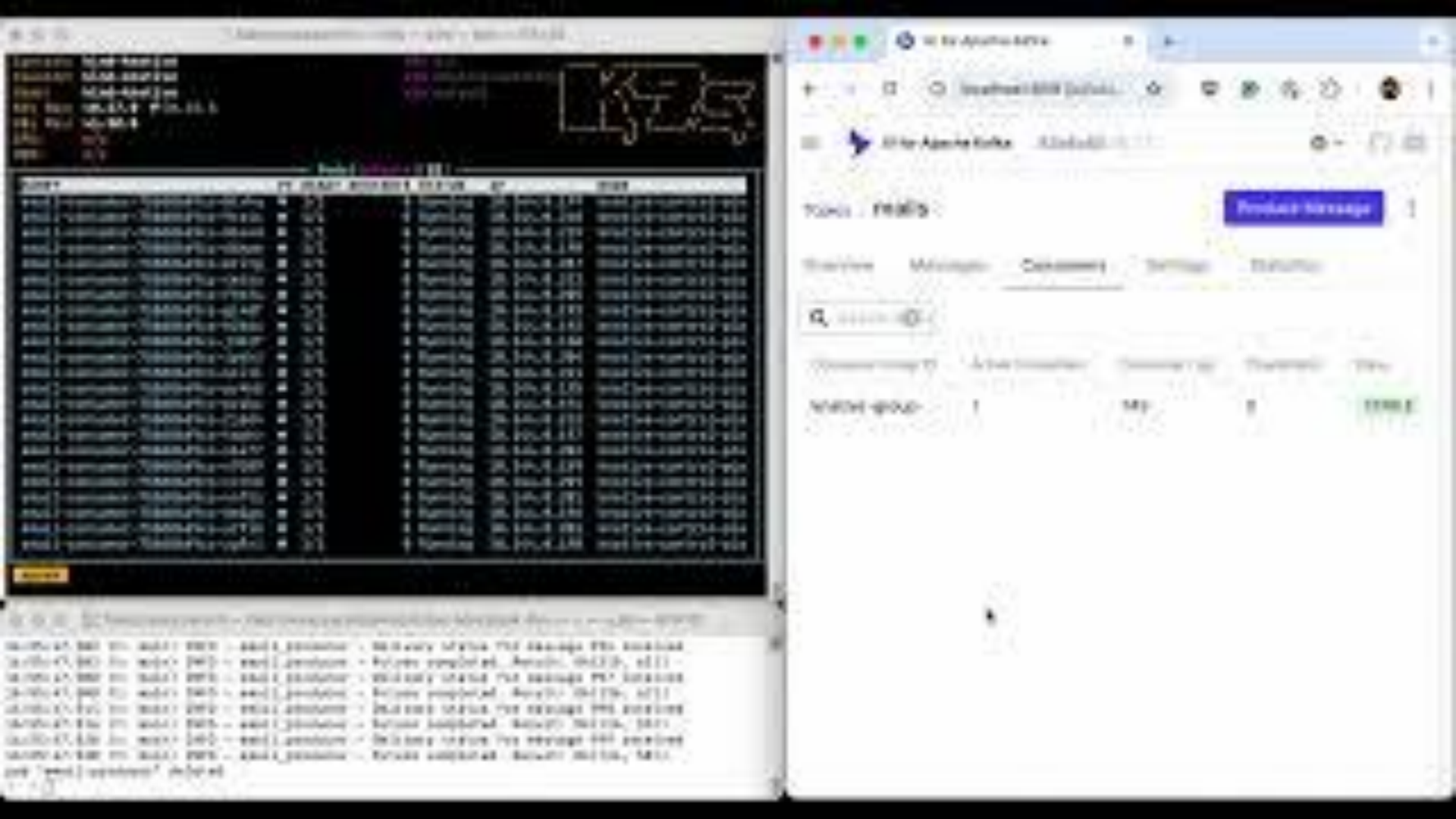
Bytecode



Batteries Included

Demo Architecture





Productive usage

WASI is in v0.2

- v0.3: expected this year
- v1: expected in 2026

CNCF Usage of WASM

allowes extendions with custom logic compiled to WASM using the http-handler

allowes filtering of requests with the Proxy-Wasm sandbox API

SWISS CLOUD NATIVE DAY

Dapr usage of WASI (simplified)

Wasm http-handler in TinyGo

```go
package main

import (
    "strings"

    "github.com/http-wasm/http-wasm-guest-tinygo/handler"
    "github.com/http-wasm/http-wasm-guest-tinygo/handler/api"
)

func main() {
    handler.HandleRequestFn = handleRequest
}

// handleRequest implements a simple HTTP router.
func handleRequest(req api.Request, resp api.Response) (next bool, reqCtx uint32) {
    // If the URI starts with /host, trim it and dispatch to the next handler.
    if uri := req.GetURI(); strings.HasPrefix(uri, "/host") {
        req.SetURI(uri[5:])
        next = true // proceed to the next handler on the host.
        return
    }

    // Serve a static response
    resp.Headers().Set("Content-Type", "text/plain")
    resp.Body().WriteString("hello")
    return // skip the next handler, as we wrote a response.
}
```

https://docs.dapr.io/reference/components-reference/supported-middleware/middleware-wasm/#generating-wasm

Run WASM

```go
import "github.com/tetratelabs/wazero"

// ...

r := wazero.NewRuntime(ctx)
defer r.Close(ctx)
mod, _ := r.Instantiate(ctx, wasmAdd)
res, _ := mod.ExportedFunction("add").Call(ctx, 1, 2)
```

https://wazero.io/

SWISS CLOUD NATIVE DAY

How to get started?

1. Familiarize yourself with the WebAssembly Component Model.
2. Familiarize yourself with the WebAssembly System Interface.
3. Implement a Hello World in Rust
4. Choose a WebAssembly Runtime
   a. Wasmer
   b. WasmEdge
   c. WAMR
   d. Wasmtime
5. Prepare Kubernetes Nodes
   a. runwasi
   b. crun

Swiss Cloud Native Day
2025

Supercharging Edge Event-Driven Architecture on Kubernetes with Wasm

PRESENTER

Fabrizio Lazzaretti
Managing Consultant,
Wavestone

PRESENTER

Linus Basig
Head of Engineering,
CARU AG