MONTRÉAL JUG · LIGHTNING PRÉSENTATIONS
ÉCLAIRS, ÉPISODE 2 · 29 MARS 2023

Sablier

Scale to zero

github.com/sablierapp/sablier



SABLIER, SCALE TO ZERO

Alexis Couvreur

Fullstack Software Engineer

Working at the National Bank of Canada for SII Canada

SABLIER, SCALE TO ZERO

Reducing infrastructure costs boils down to turning apps off when you don’t use them

Scaling to zero

Scaling to zero allows workloads to be run only when there is demand

They scale to zero

- Google Serverless (App Engine, Function)
- AWS Lambda
- Heroku
- Github Codespace
- And more!



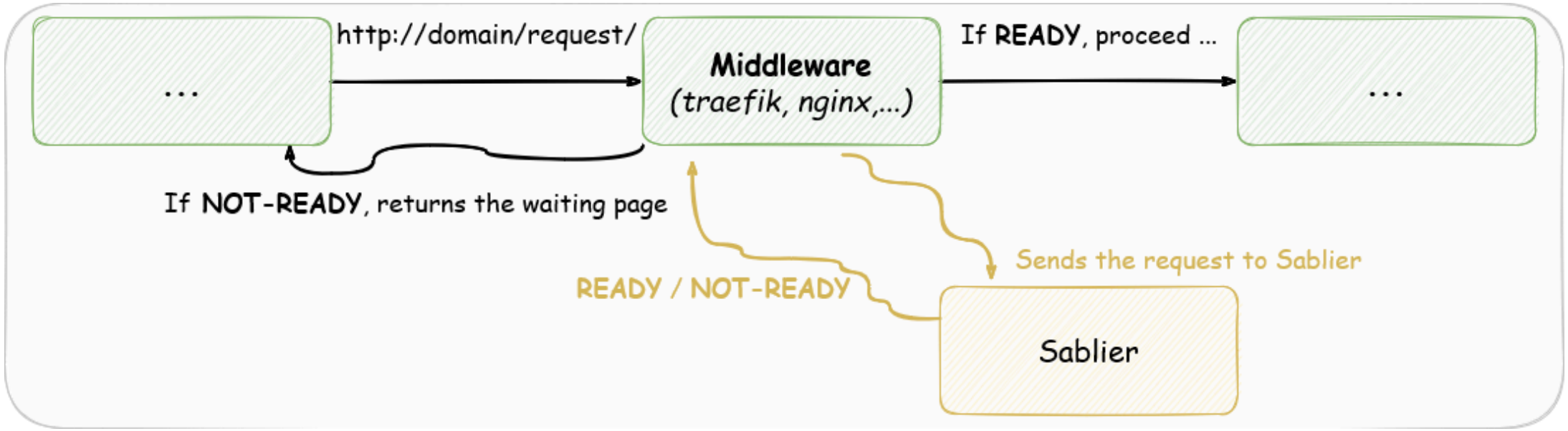
SABLIER, SCALE TO ZERO

Sablier

A free and open source software scaling to zero solution

How does it work?

Reverse proxy integration



How to use Sablier?

Let’s see its integration with Caddy!

Without Sablier

docker-compose.yaml

```yaml
services:
  proxy:
    image: caddy:2.6.4
    ports:
      - "8080:80"
    volumes:
      - ./Caddyfile:/etc/caddy/Caddyfile:ro

  whoami:
    image: traefik/whoami:v1.12.0
```

Caddyfile

```
:80 {
  route /whoami {
    reverse_proxy whoami:80
  }
}
```

With Sablier

docker-compose.yaml

```yaml
services:
  proxy:
    image: caddy:local
    ports:
      - "8080:80"
    volumes:
      - ./Caddyfile:/etc/caddy/Caddyfile:ro

  whoami:
    image: traefik/whoami:v1.12.0
    labels:
      - sablier.enable=true
      - sablier.group=demo

  sablier:
    image: sablierapp/sablier:1.19.1
    volumes:
      - '/var/run/docker.sock:/var/run/docker.sock'
```

Caddyfile

```
:80 {
  route /whoami {
    sablier http://sablier:10000 {
      group demo
      session_duration 1m
      dynamic {
        display_name Dynamic-Whoami
      }
    }
    reverse_proxy whoami:80
  }
}
```

SABLIER, SCALE TO ZERO

Demo

Sablier features

Providers

- Docker
- Docker Swarm
- Kubernetes
- Podman
- AWS EC2



Label filtering

- Docker labels
- Docker swarm service labels
- Kubernetes deployments labels

Docker labels

```yaml
services:
  whoami:
    image: traefik/whoami:v1.12.0
    labels:
      - sablier.enable=true
      - sablier.group=mygroup
```

Kubernetes deployments labels

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: whoami
  labels:
    app: whoami
    sablier.enable: "true"
    sablier.group: mygroup
```

Reverse proxies

- Traefik (Go)
- Caddy (Go)
- Nginx (Javascript/Lua/C)
- more to come...



Integration nightmare

Develop and maintain an integration for each reverse proxy

- Different APIs
- Different languages (Go, C, Javascript, Lua)
  - Runtime plugin execution
  - Compilation plugin (recompile the proxy)
- Breaking changes
- Provider compatibility
- Cloud native

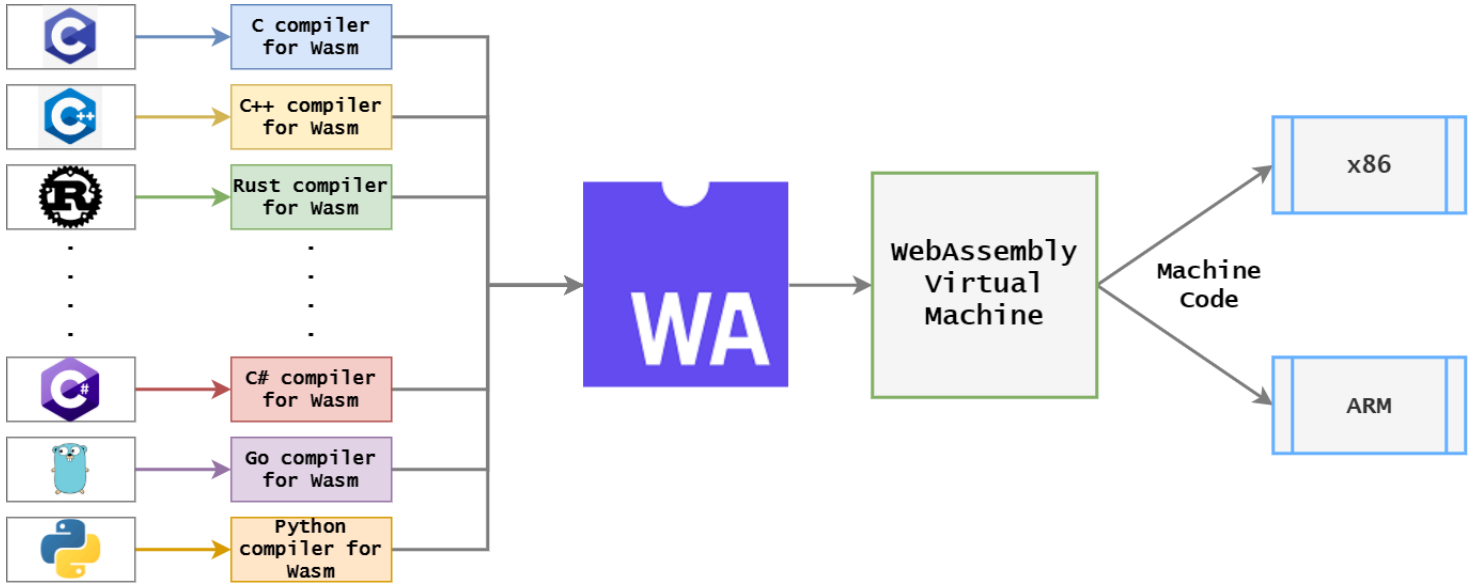


WebAssembly to the rescue

> WebAssembly (abbreviated Wasm) is a binary instruction format designed as a portable compilation target for programming languages, enabling deployment on the web for client and server applications.

What is Web Assembly

- Bytecode (almost native speed)
- Compilation target for higher level languages (Rust, Go, C++...)
- Runs in a sandboxed environment

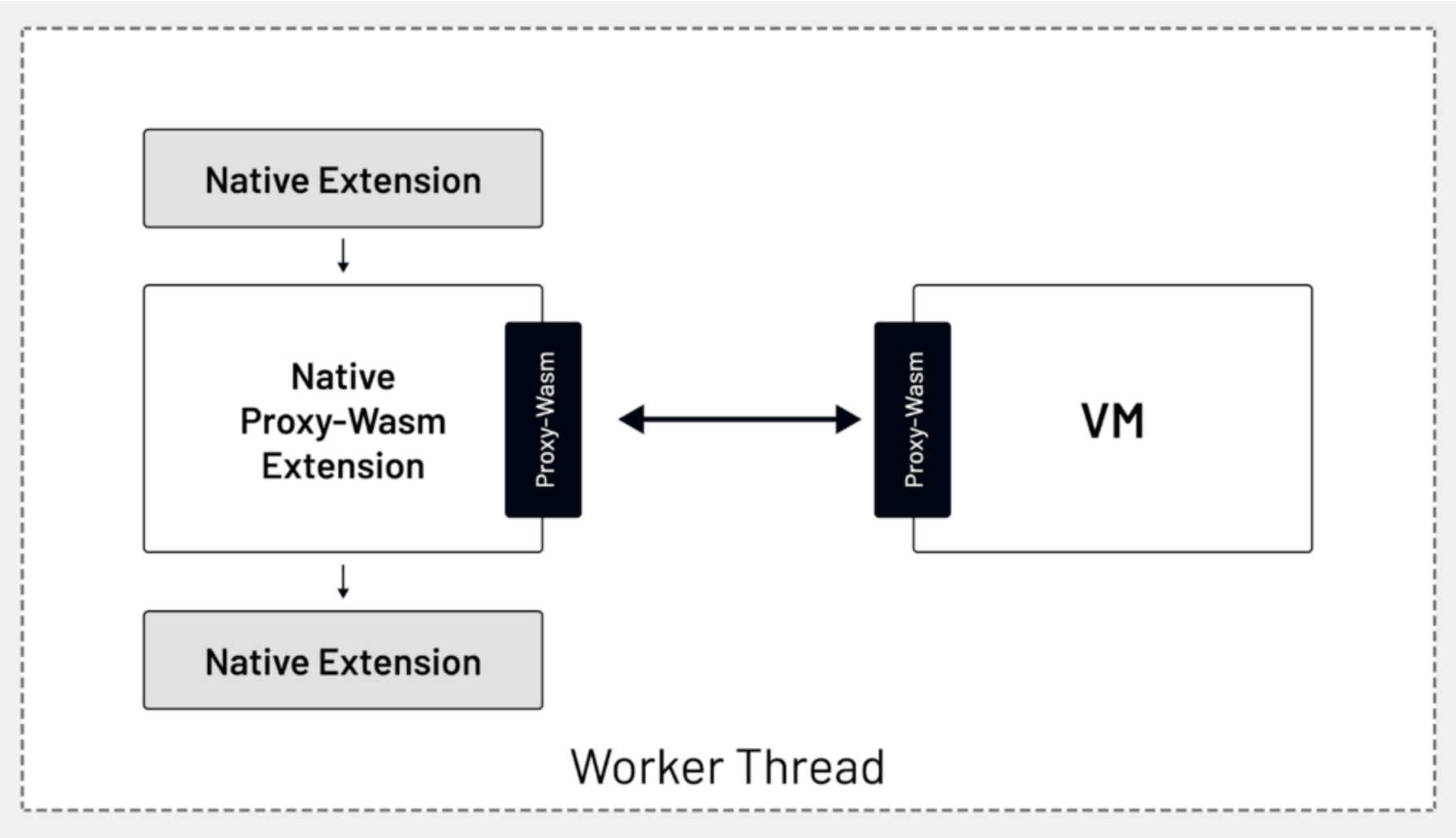


SABLIER, SCALE TO ZERO

ProxyWasm

WebAssembly modules for reverse proxies

SABLIER, SCALE TO ZERO



No more integration nightmare

Create one WASM Module to be used by all reverse proxies

Reverse proxy maintainers will have to develop and maintain one ProxyWasm extension

Thank you!

Questions?