

Serverless & PHP

Where did the server go?

PHP

Hi!

@matthieunapoli

Sponsors



I am not a sysadmin!

Serverless

Do less server stuff

Not just about code

- Serverless file storage
- Serverless databases
- Serverless message queues
- Serverless search engines
- ...

Manage less

Scale more

Pay per use

Serverless for code = FaaS

Function as a Service

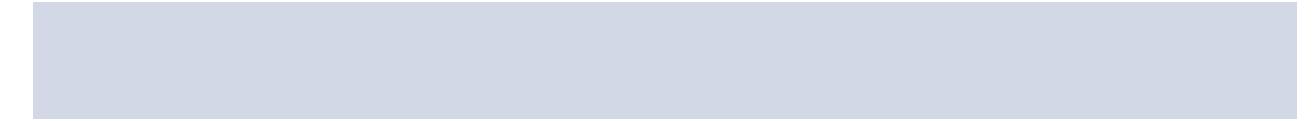


ZIP

Code

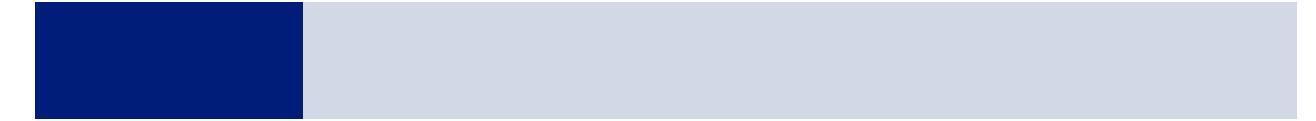








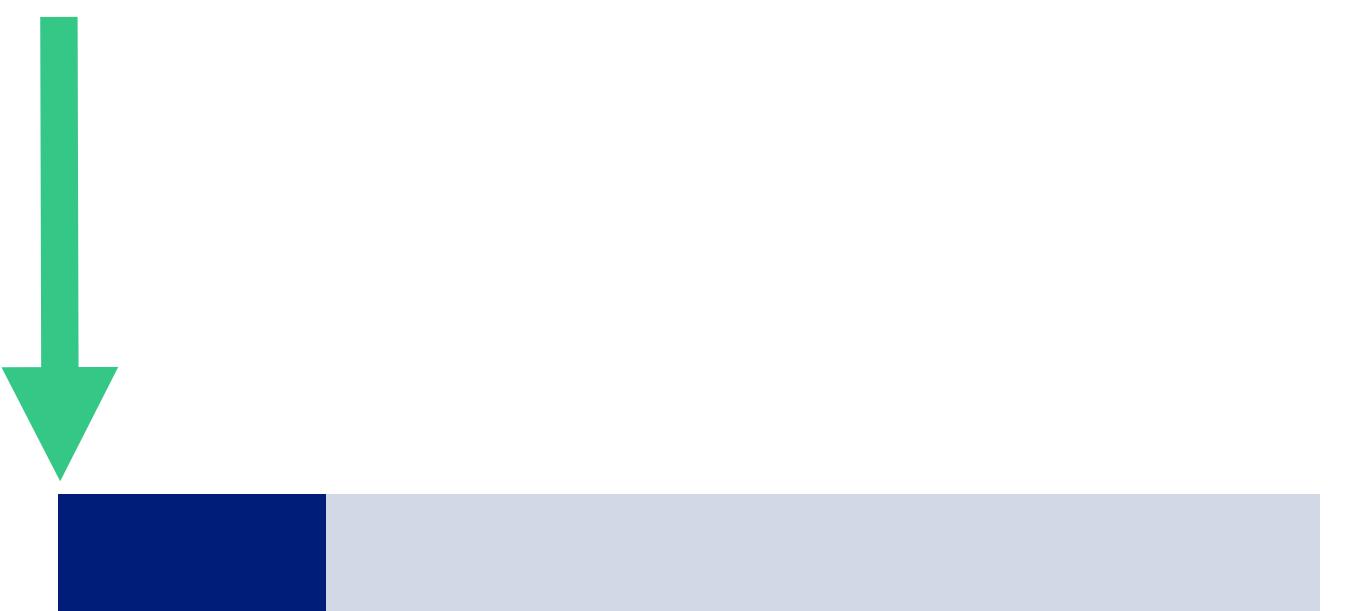
index.php



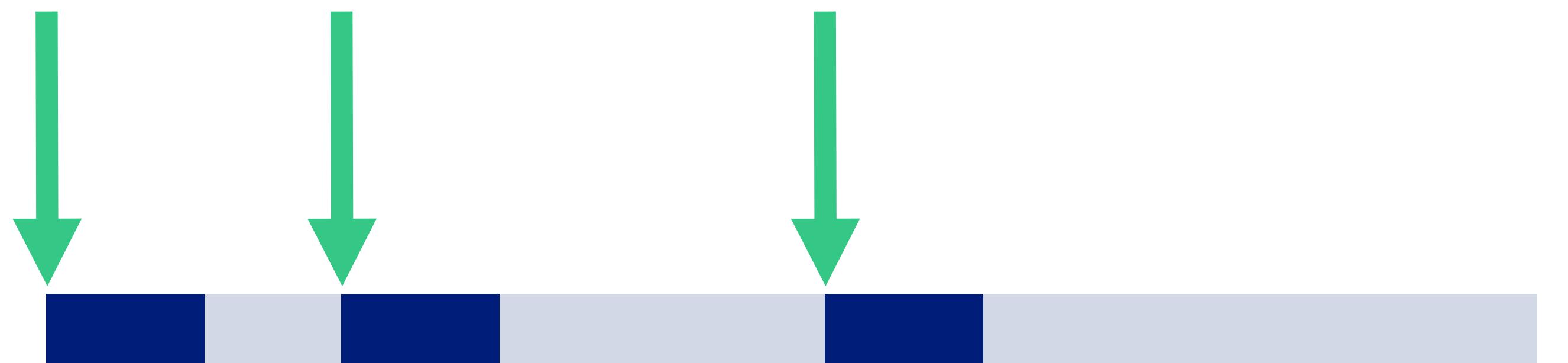
10 minutes













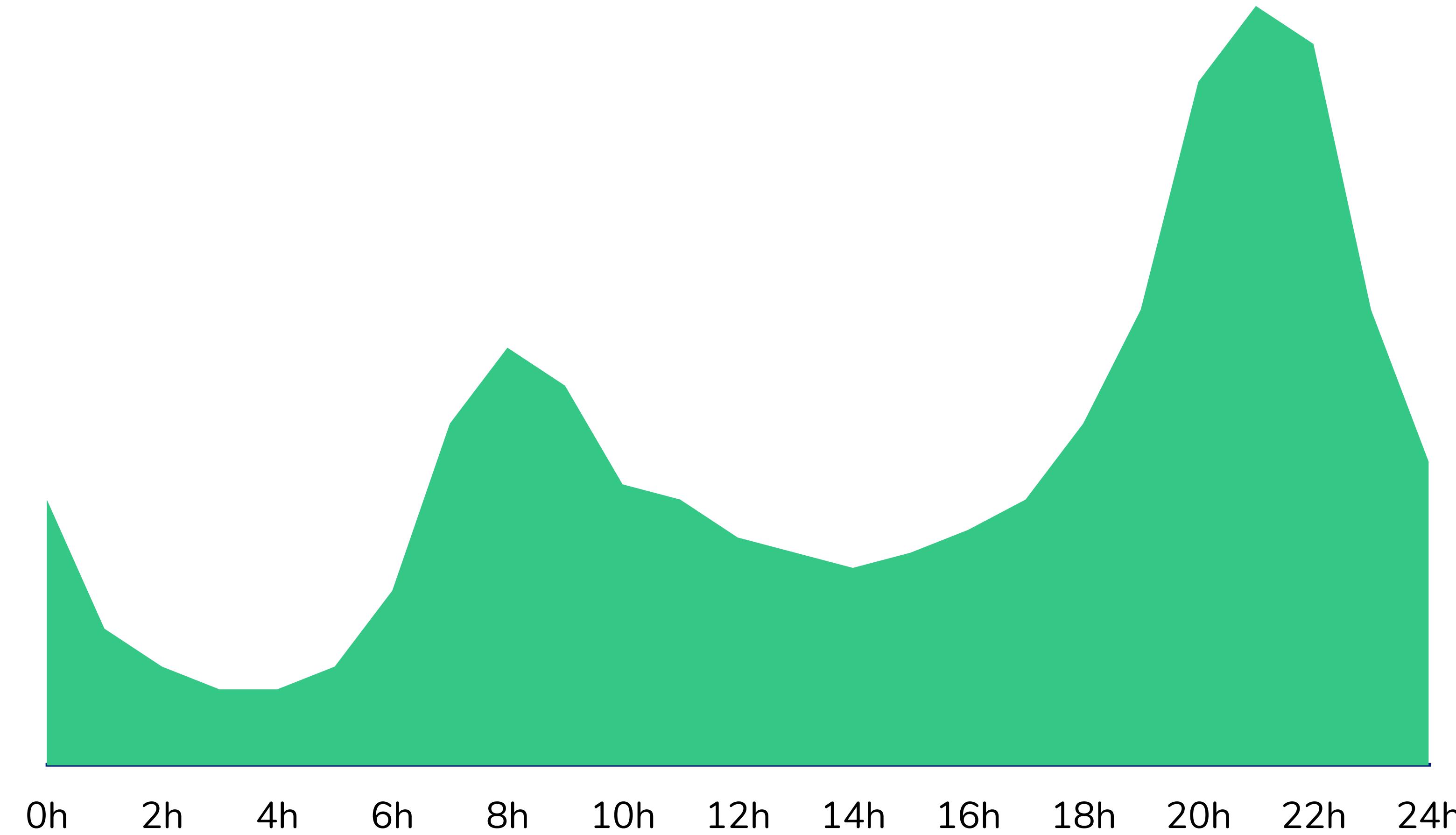


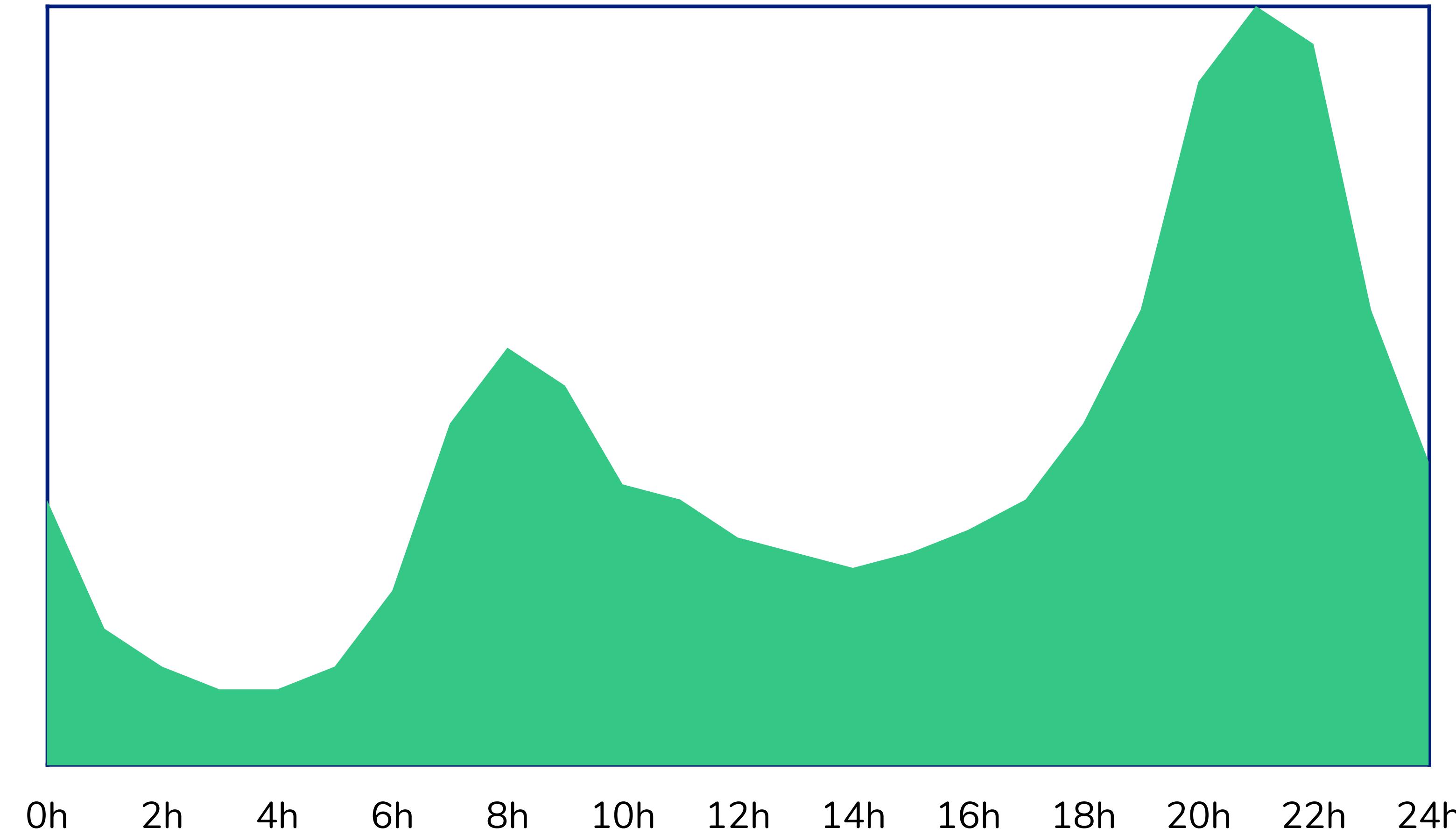


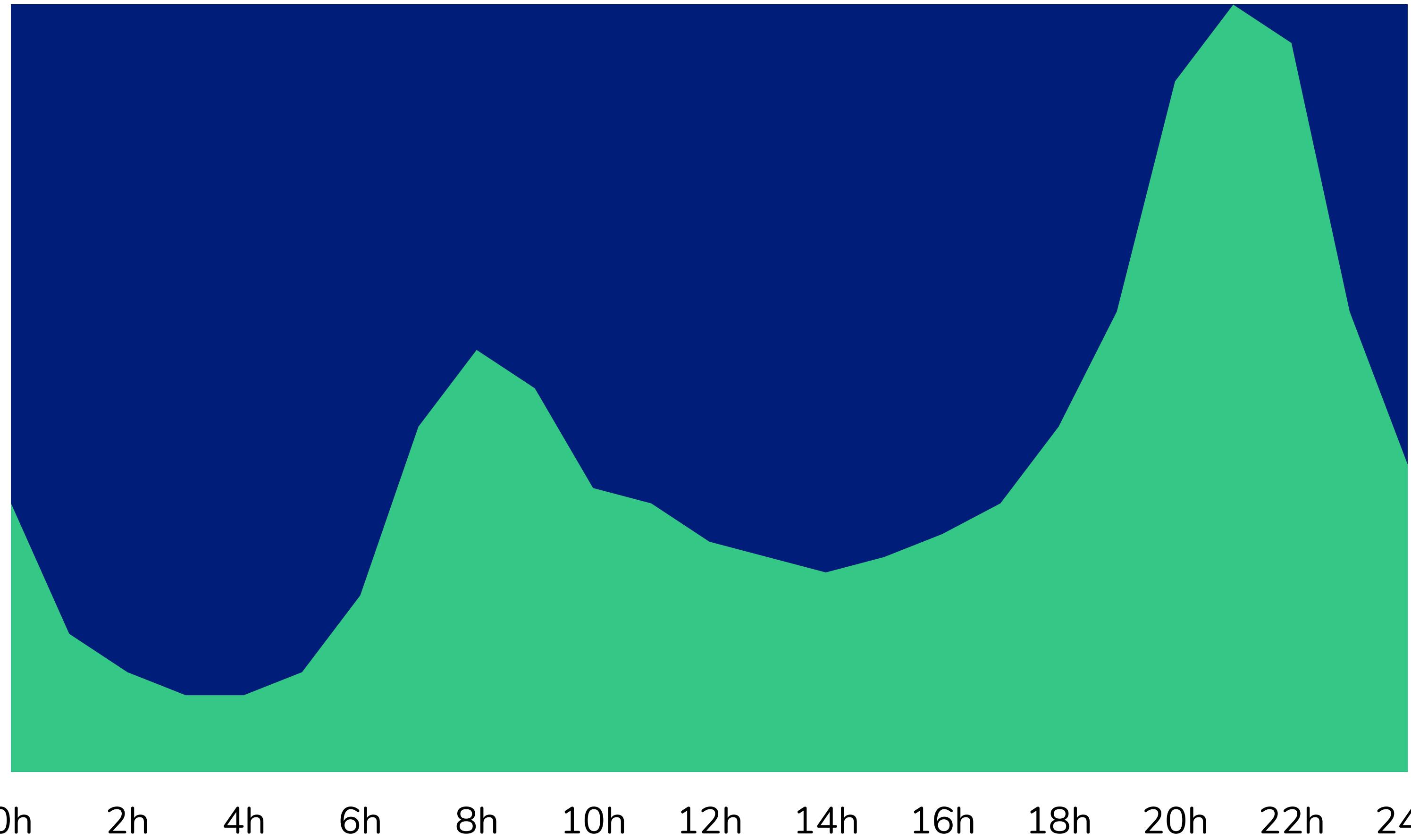


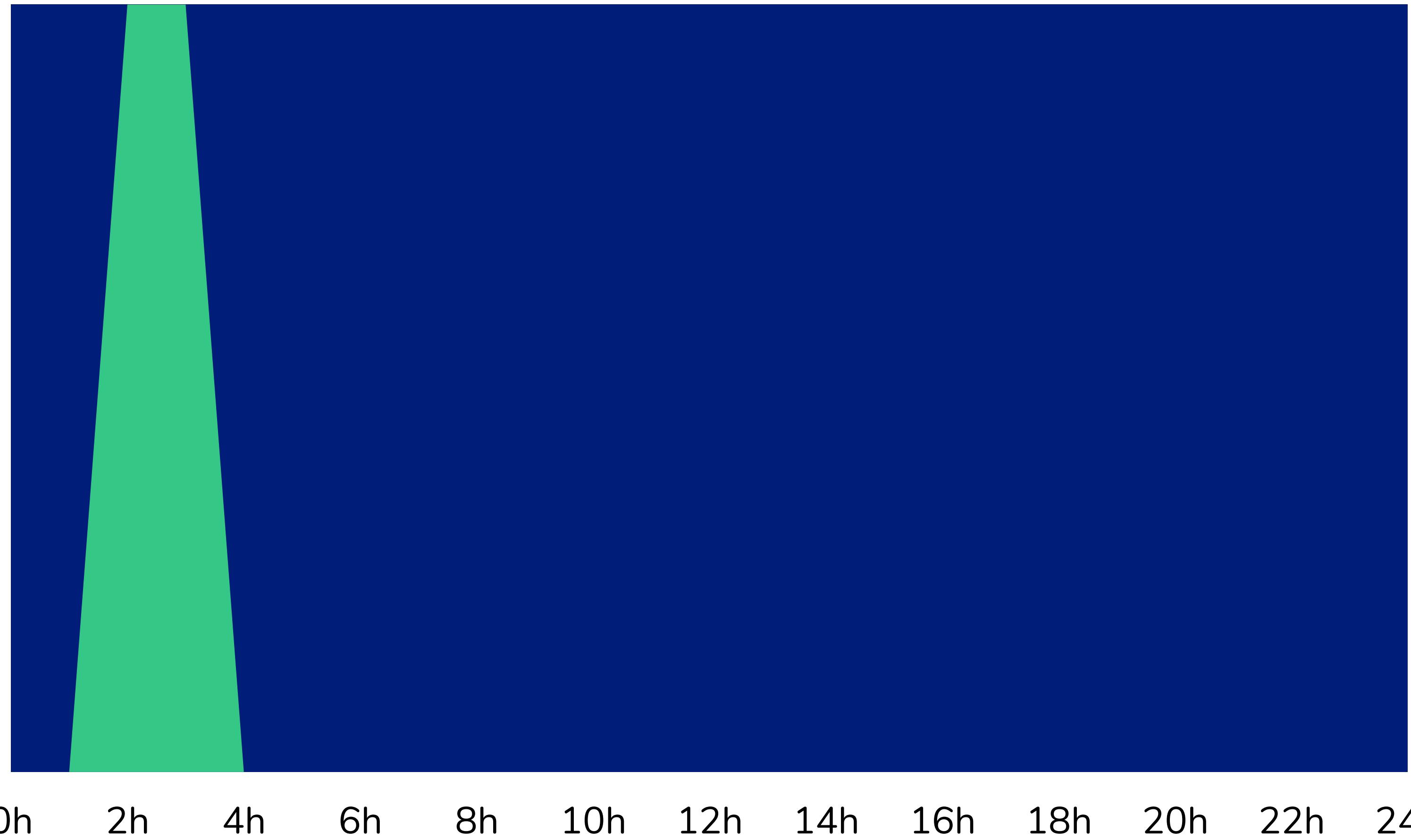
300ms

$$\$0.0000002 + 300 \times \$0.00000000208$$









Manage less

Scale more

Pay per use

- AWS Lambda
- Microsoft Azure Functions
- Google Cloud Functions
- CloudFlare Workers
- IBM OpenWhisk
- Alibaba Cloud Function
- ...

PHP

bref

running PHP made simple

Everything you need to easily
deploy and run serverless PHP applications.

Choices

Documentation

Tools

Symfony

Laravel

...

Symfony Messenger

Laravel Queues

Serverless framework

serverless.com

serverless.yml

```
service: my-app

provider:
  name: aws

functions:
  web:
    handler: public/index.php
    runtime: provided
    layers:
      - ${bref:layer.php-74-fpm}
  events:
    - httpApi: '*'

plugins:
  - ./vendor/bref/bref
```

```
$ composer require bref/bref
```

```
$ vendor/bin/bref init
```

```
$ serverless deploy
```

```
...
```

```
Service deployed
```

```
https://wyn0qrz.execute-api.us-east-1.amazonaws.com/
```

Demo!

Case studies

#externals

Opening PHP's #internals to the outside.

Search with  algolia

Search...

- [↑ 0 Adding return types to internal methods \(PHP 8\)](#)
🕒 4 hours ago  Nikita Popov  1

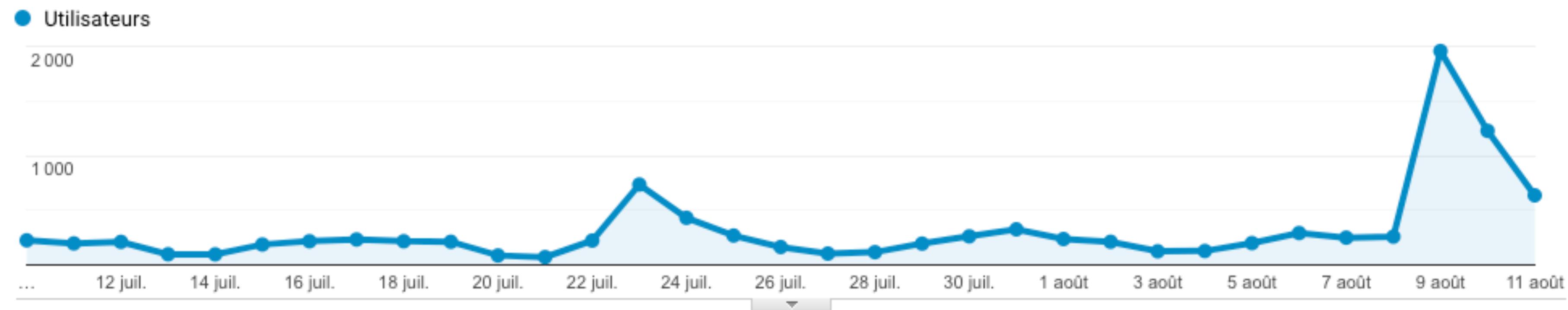
- [↑ 0 P++: FAQ](#)
🕒 1 day ago  Zeev Suraski  24

- [↑ 2 Call for participation: Annotating internal function argument and return types](#)
🕒 1 day ago  Nikita Popov  8

- [↑ 9 Bringing Peace to the Galaxy](#)
🕒 2 days ago  Zeev Suraski  44

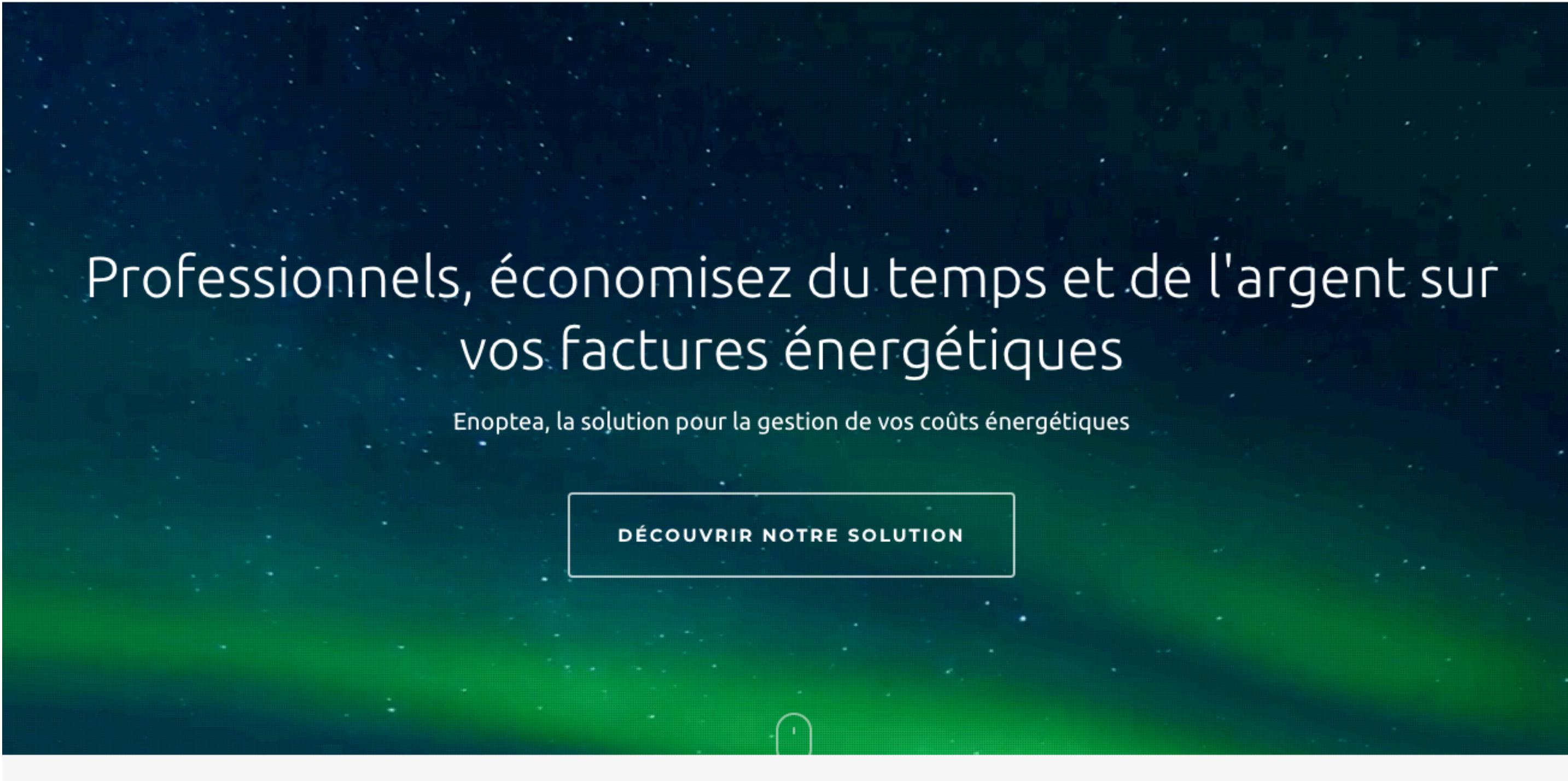
externals.io

- Website (55ms) + cron
- Custom framework, MySQL
- 100,000 requests/month
- 50\$/month -> \$17/month



\$0.04

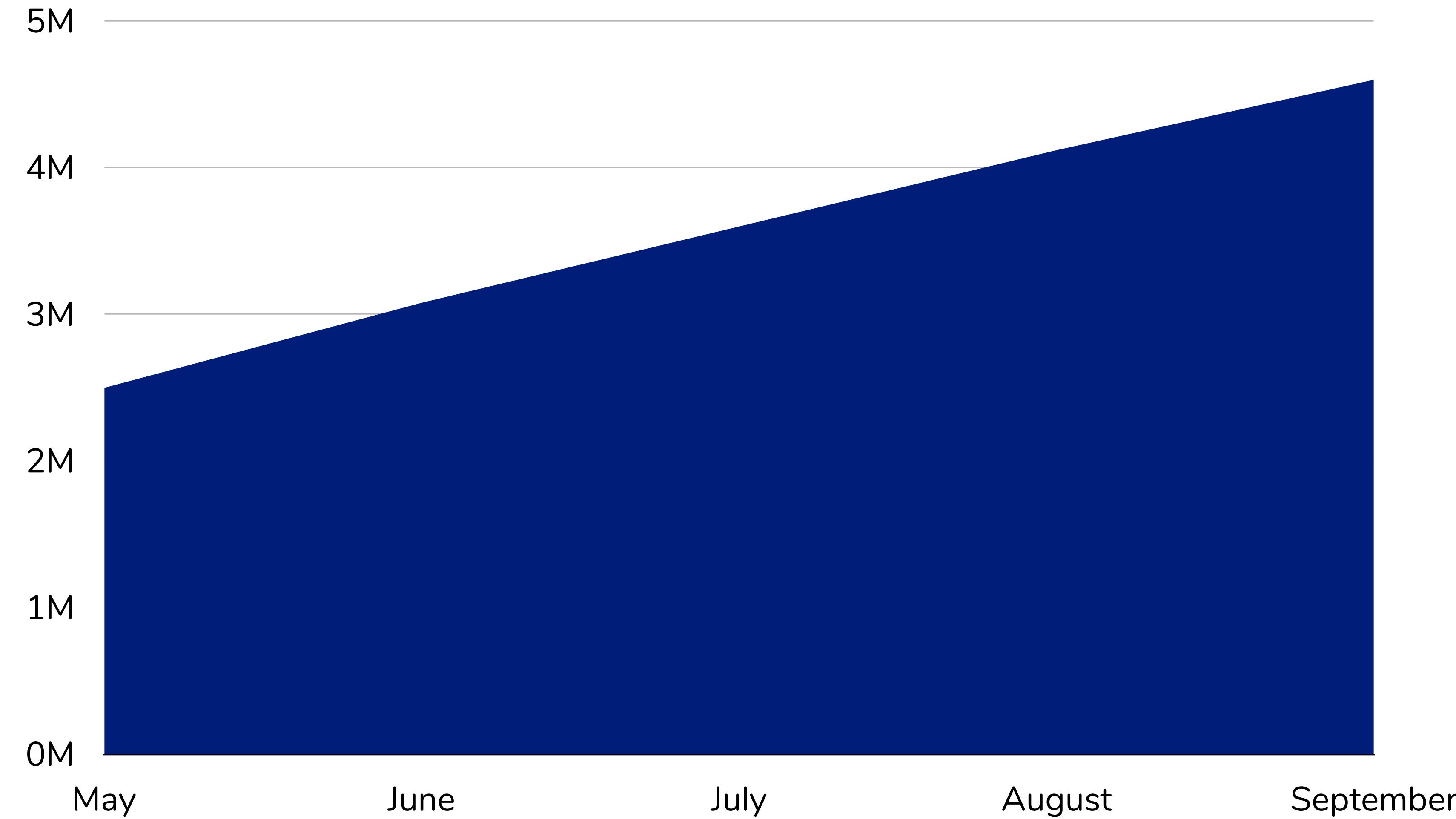
- 100k r/month -> 2 million r/month
- 17\$/month -> \$39/month

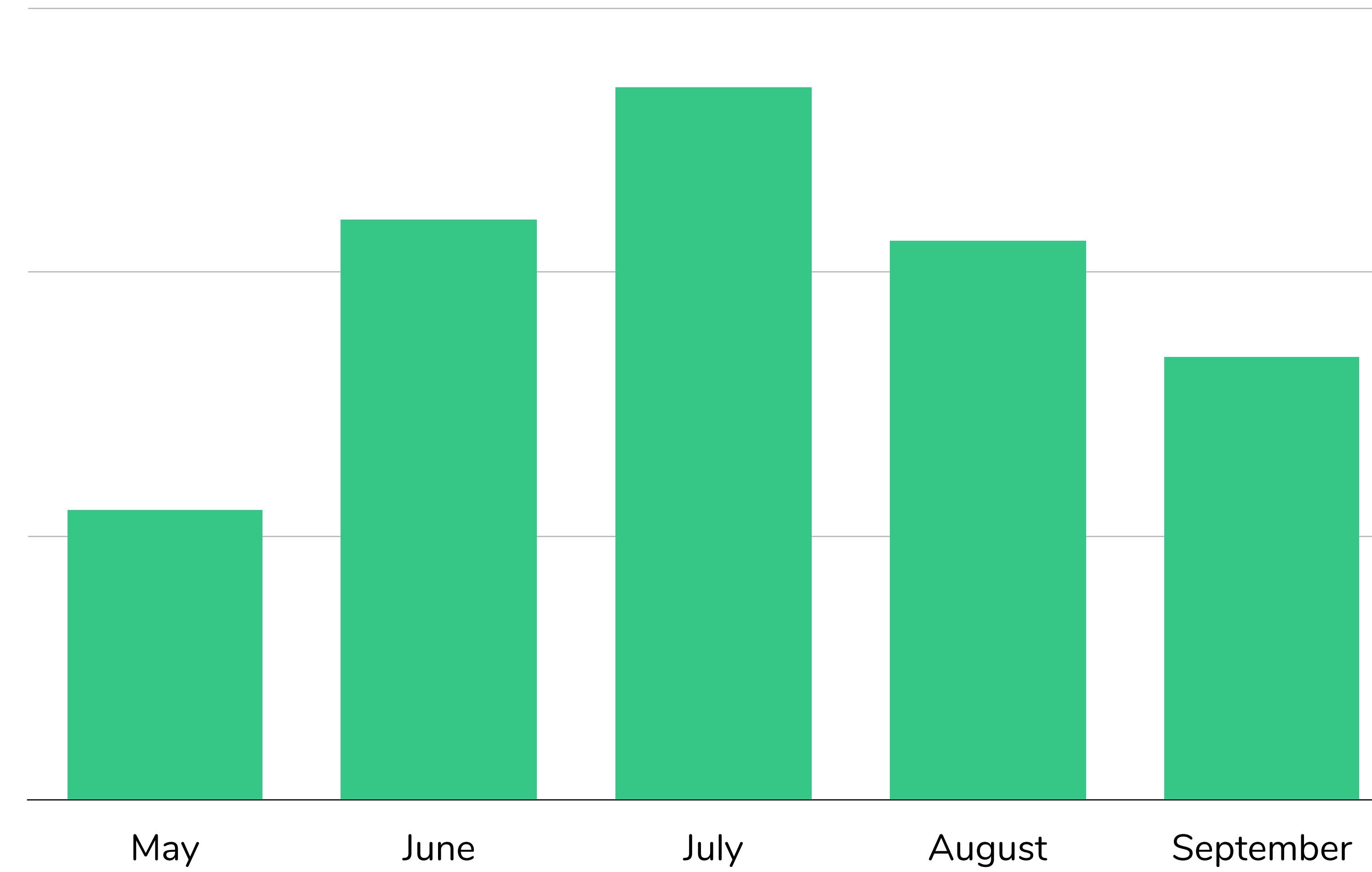


Enoptea

enoptea.fr

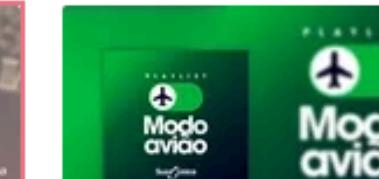
170k jobs/day





\$800 -> \$90

LANÇAMENTOS EM DESTAQUE



TV SUA MÚSICA

VER OUTROS



YouTube INSCREVA-SE NO TV SUA MÚSICA

TENHA O SUA MÚSICA NO SEU
SMARTPHONE ANDROID OU IOS

Sua Música

suamusica.com.br

- API (90ms) + website (130ms)
- Phalcon, MySQL
- 40M requests/day (Alexa's top 3000)
- 2500 concurrent lambdas
- ECS (EC2) -> Lambda: 25% savings

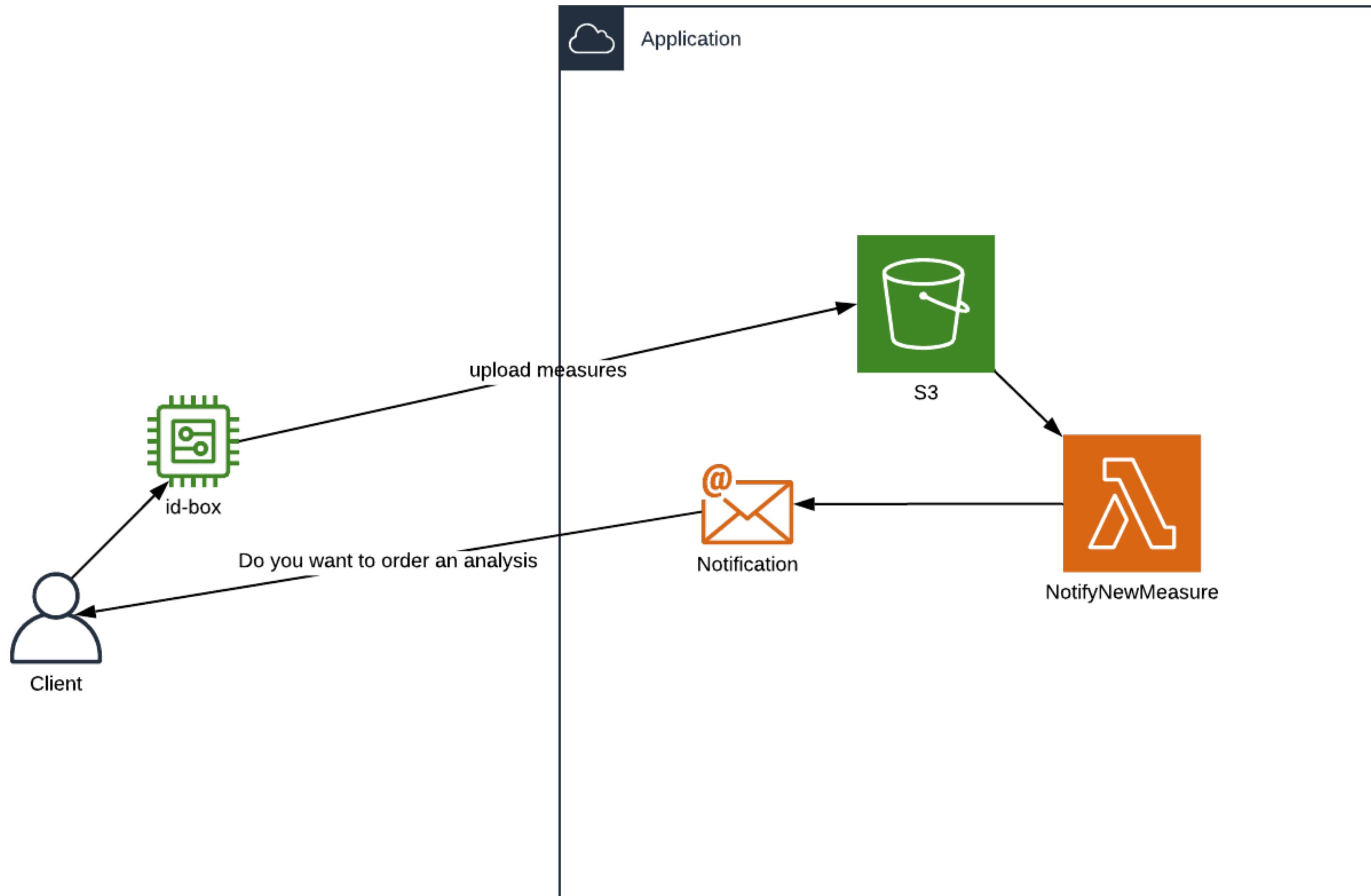


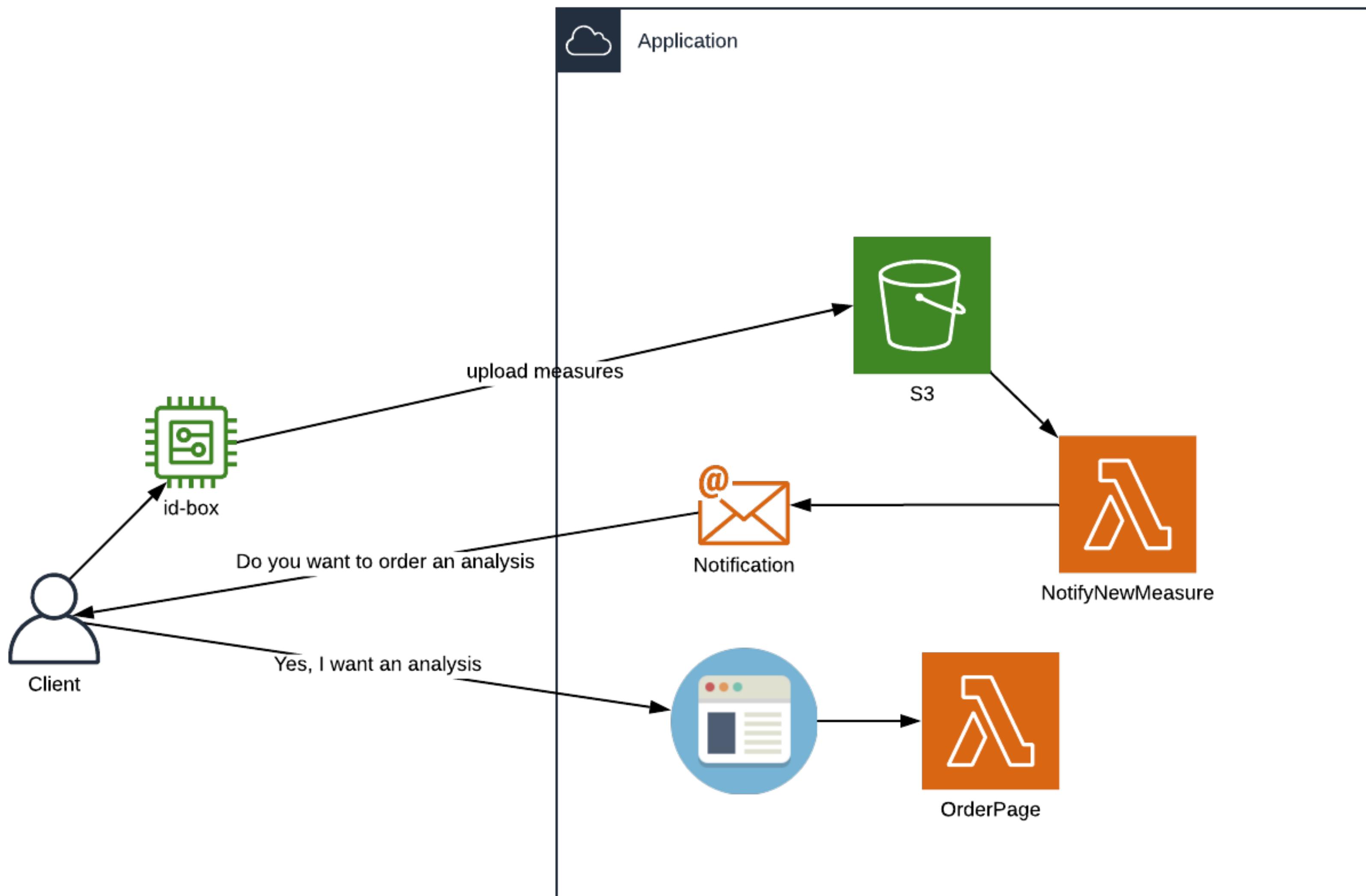
id-box

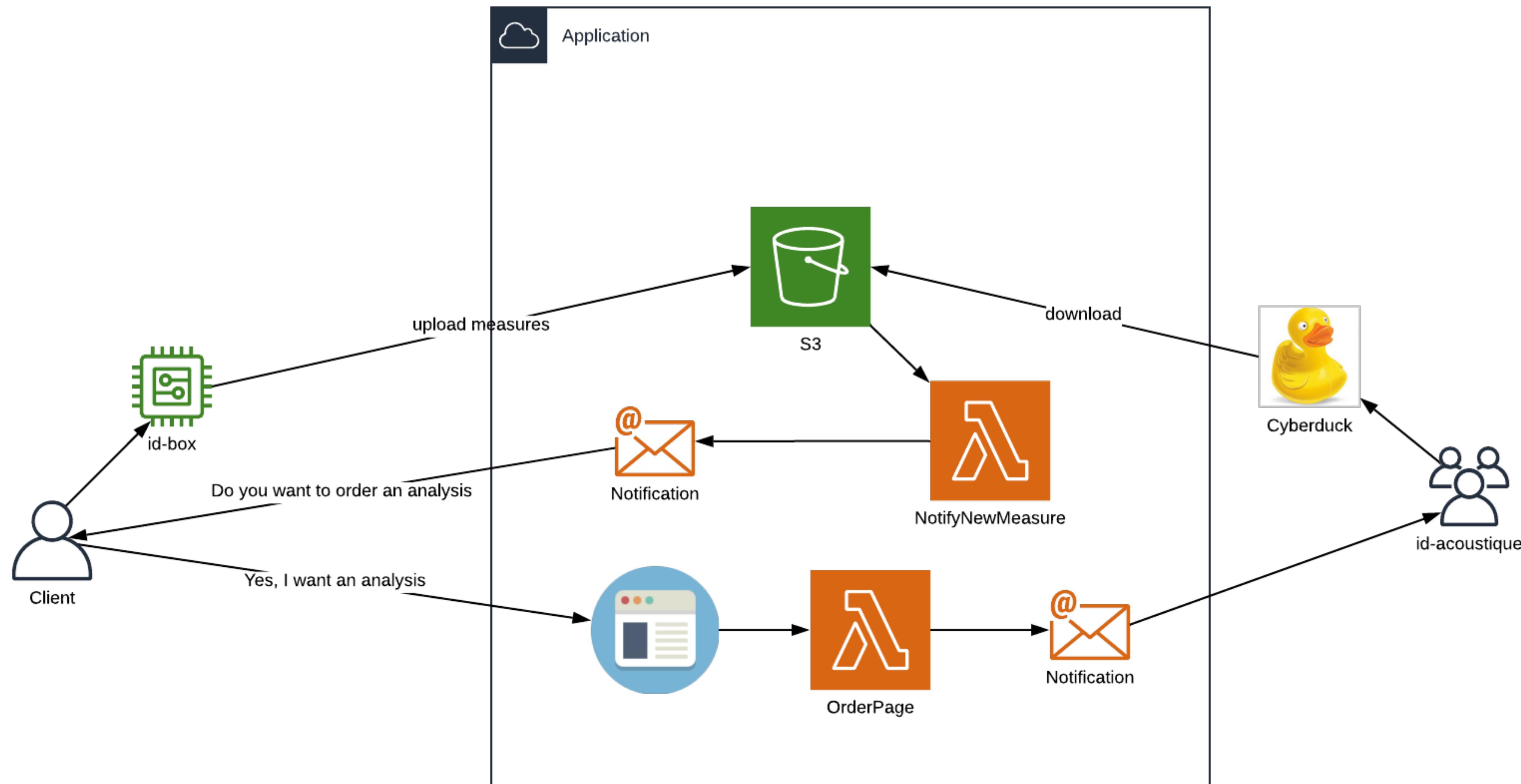
id-acoustique.com

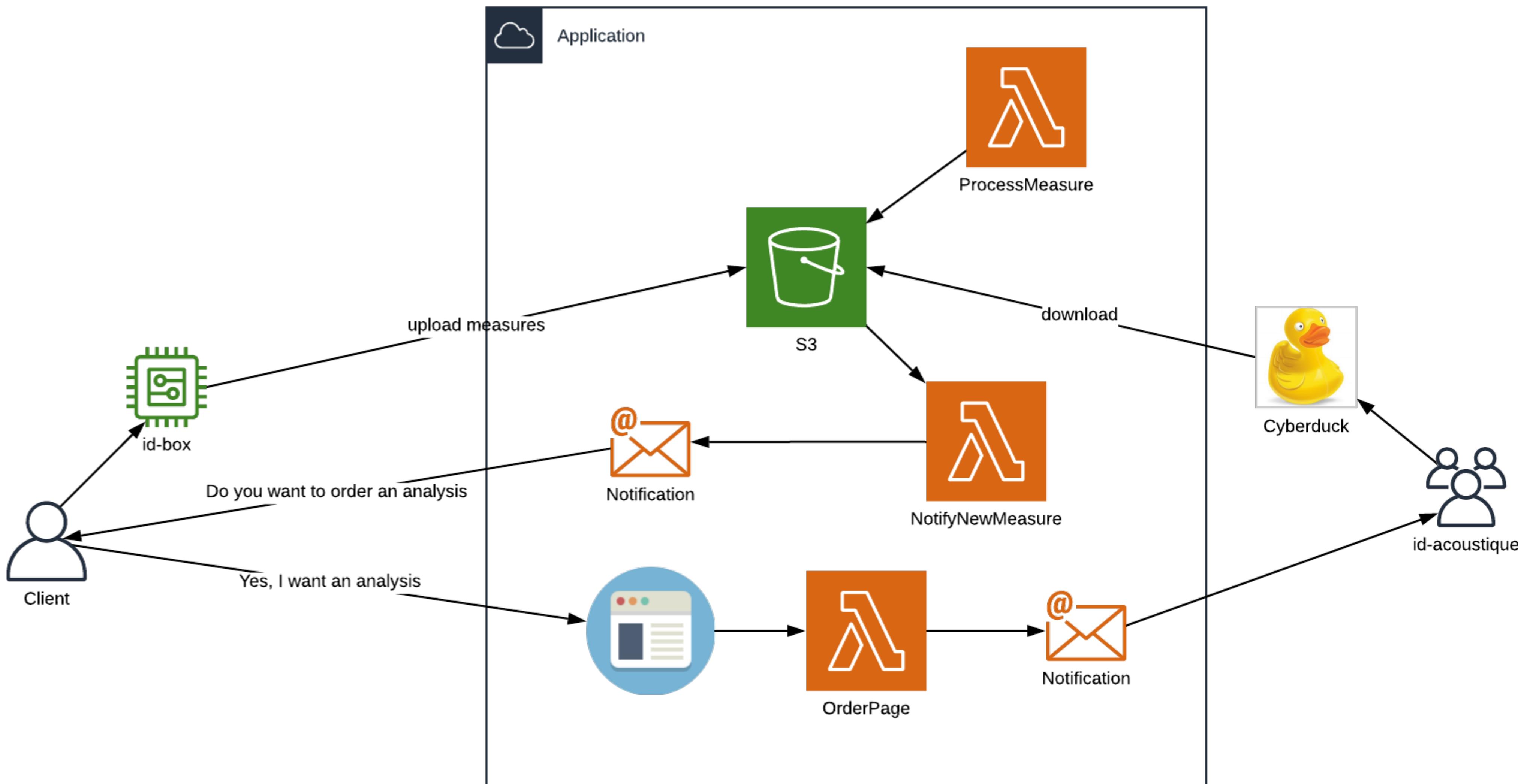












Use cases

- Small websites/APIs -> cheap, effortless hosting
- Large websites/APIs -> scalable, lower TCO
- Back-offices -> better performances for lower costs
- Many websites (agencies, multi-tenancy) -> costs, effort
- Queue workers with SQS -> scalable, cost-efficient
- Micro-services messaging -> decoupled, scalable architecture
- Micro-functions: webhook, image resizing, PDF generation...

?

is
serverless
mature?

is
the future
serverless?

is
PHP's future
serverless?

try!

learn!

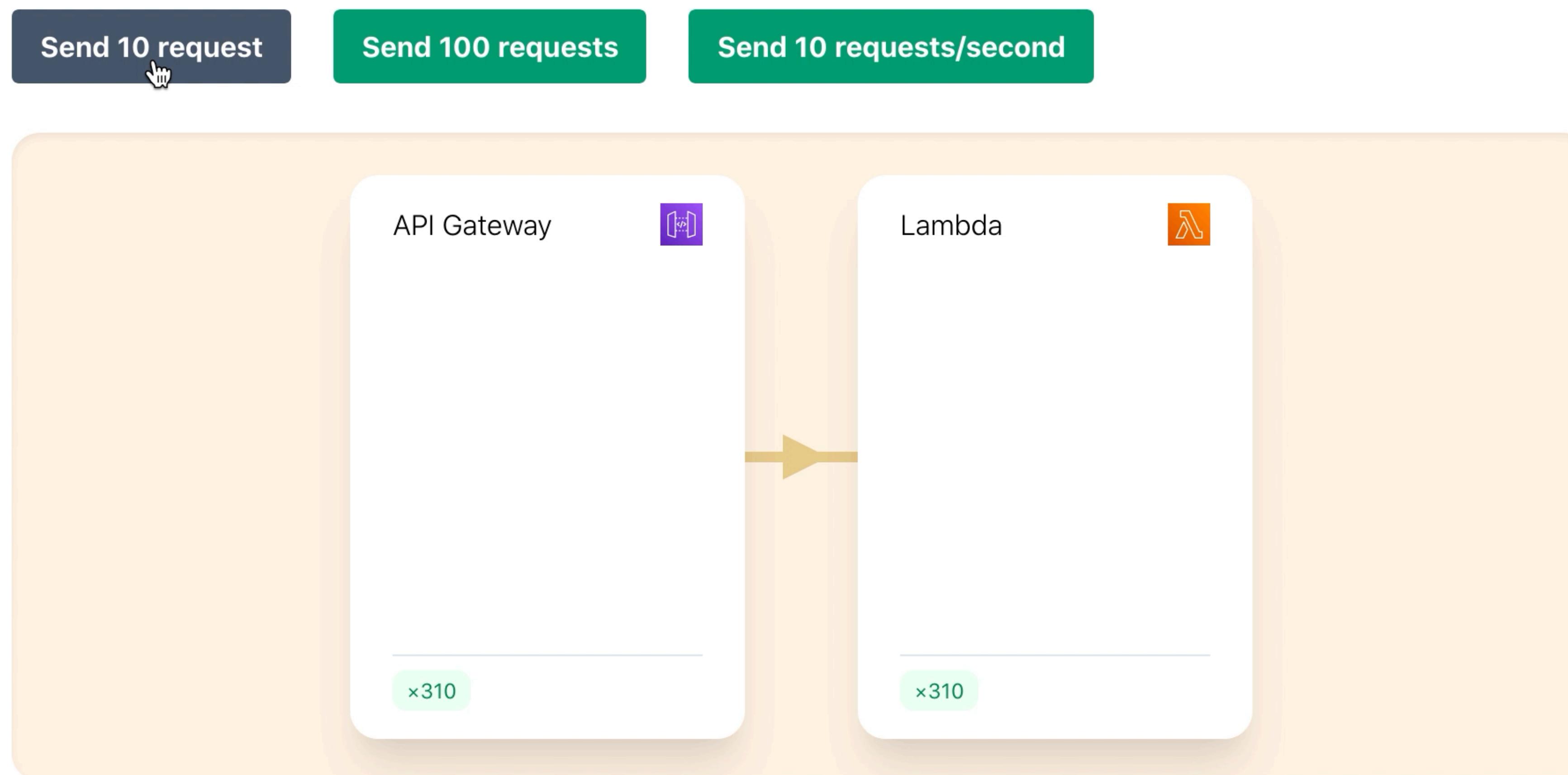
share!

bref.sh

@matthieunapoli

- i New Lambda instances will be booted instantly when the running instances are already busy handling requests.

Let's illustrate that: try sending 10 requests, then 100, then 10 requests per second.



As we can see, no matter the HTTP traffic, our application will scale instantly to handle it.

The serverless visual course
where you **learn, see, and do.**

Serverless leaves you dazed and confused?

Forget the marketing BS and verbose documentation. This course is straight to the point and teaches you **what matters**.

This is not the ultimate reference. This is how you finally piece everything together and **get started!**

Like a book,
but a bit more than a book:

- Practical use cases
- Interactive simulations
- Deployable code examples

serverless-visually-explained.com