How Openfit optimized Lambda with Datadog

Daniel Maher
Developer Relations

Table of Contents

Who?

Modern monitoring & analytics

DATADOG

Digital fitness streaming platform

openfit.

Openfit: some numbers

➔ Active subscribers: 50,000
➔ Plays per month: 400,000
➔ Hours streamed per month: 500,000
➔ Service calls per month: 65 million
➔ 206 Lambdas across 26 API gateway endpoints

Openfit:
100% serverless

What?

"As with all transformations, you have to make sure it fits your model."

– Reza Javidi

Consider the possibilities

Servers (EC2)

- ➔ Provisioning of instances, ASG, sec groups, VPC, NAT…
- ➔ Maintenance of OS, frameworks, code…
- ➔ Security and logistics…

Consider the possibilities

Servers (EC2)

- ➔ Provisioning of instances, ASG, sec groups, VPC, NAT…
- ➔ Maintenance of OS, frameworks, code…
- ➔ Security and logistics…

Containers

- ➔ Provision and deploy the container images, VPC, NAT…
- ➔ Maintenance of code
- ➔ Security and logistics…

Consider the possibilities

Servers (EC2)

- ➔ Provisioning of instances, ASG, sec groups, VPC, NAT…
- ➔ Maintenance of OS, frameworks, code…
- ➔ Security and logistics…

Containers

- ➔ Provision and deploy the container images, VPC, NAT…
- ➔ Maintenance of code
- ➔ Security and logistics…

Serverless (Lambda)

- ➔ Maintenance of code and dependencies
- ➔ Code-level security

Serverless isn’t magic; your workload needs to fit the model

- ➔ Discrete actions that aren’t long-running.
- ➔ Actions that can deal with—or benefit from—high concurrency.
- ➔ Actions that have predictable and stable resource requirements.

" RESTful APIs that do cloud actions are perfect candidates for serverless."

– Reza Javidi

Sure, but…

A tale of two models

Servers (EC2)

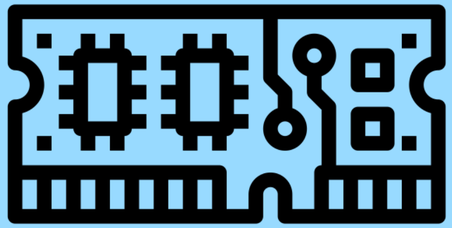
- ➔ Pay for each running instance, whether it's being used or not
- ➔ Have to manage number of instances depending on load
- ➔ Have to pay salaries for humans to figure those two things out

Serverless (Lambda)

- ➔ Pay-per-use
- ➔ Have to consider the resource requirements…

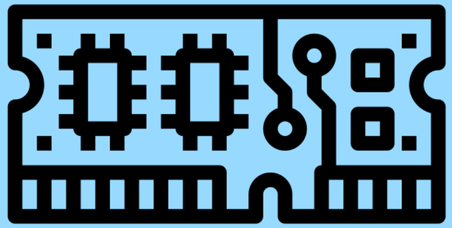
Resource requirements

What are you actually paying for?

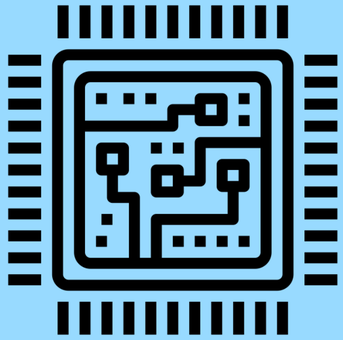


Memory
(RAM)

What are you actually paying for?

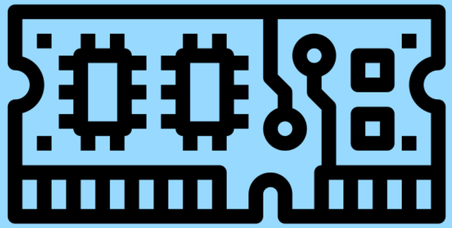


Memory
(RAM)

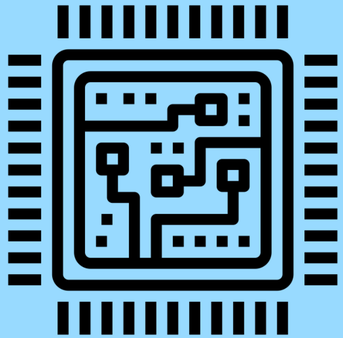


Processor
(CPU)

What are you actually paying for?



Memory
(RAM)



Processor
(CPU)

Time
(Execution)

Don’t forget cold-starts…

In the context of Lambda…

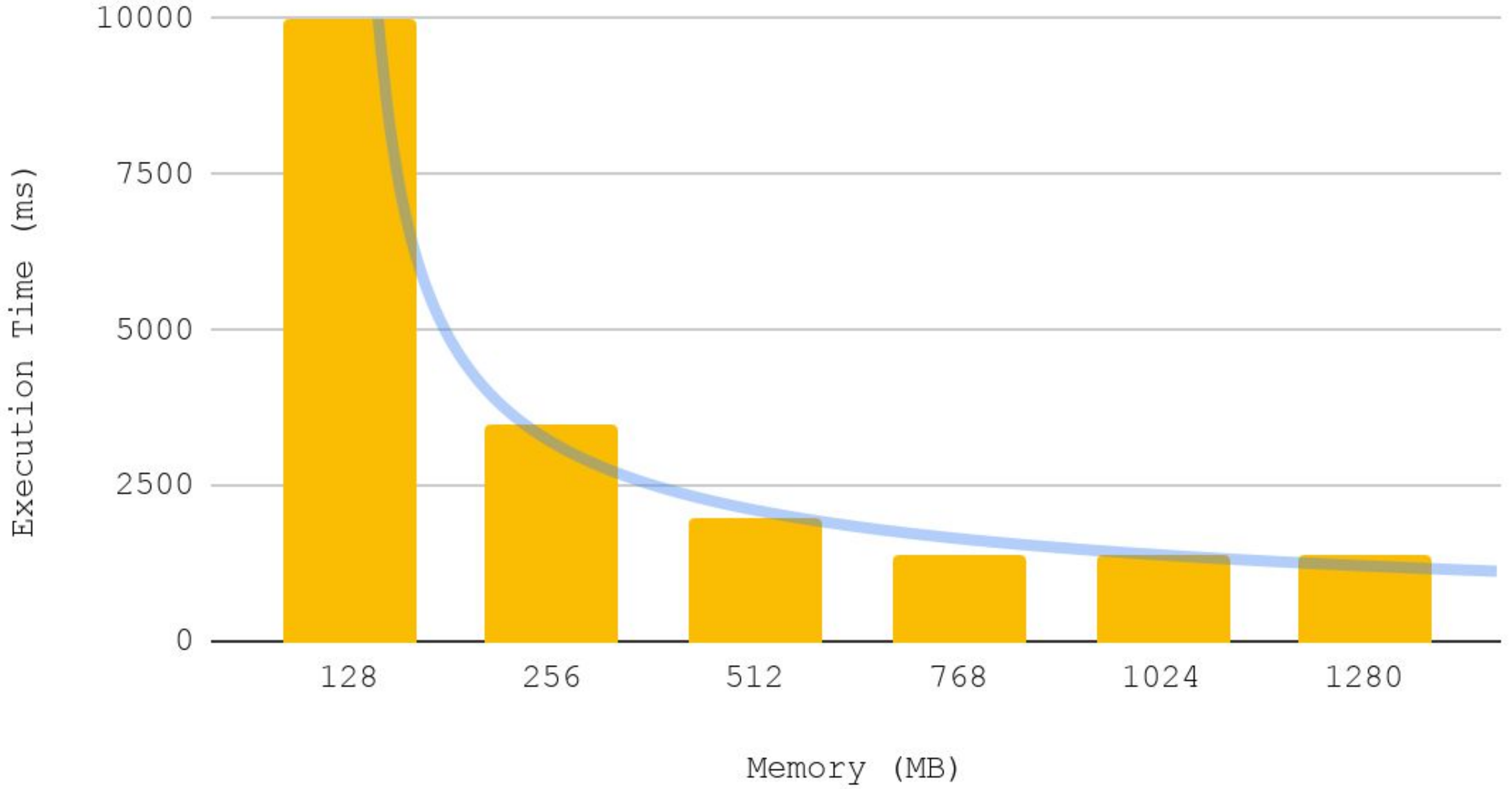
Couplets of memory and processor, per 100ms of time consumed.

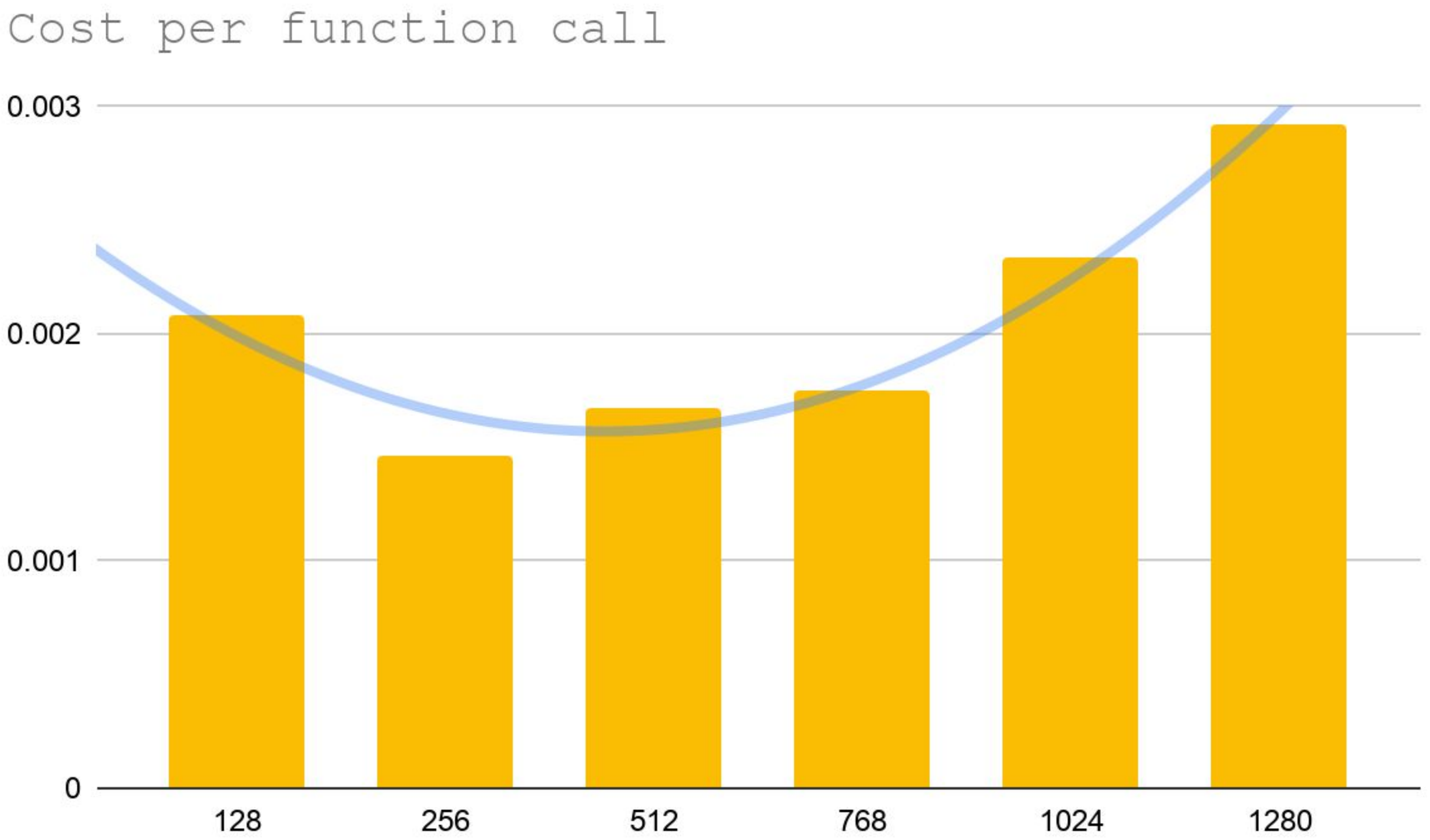
" It's very affordable. If your functions are short and optimized, it's a cost-savings."

– Reza Javidi

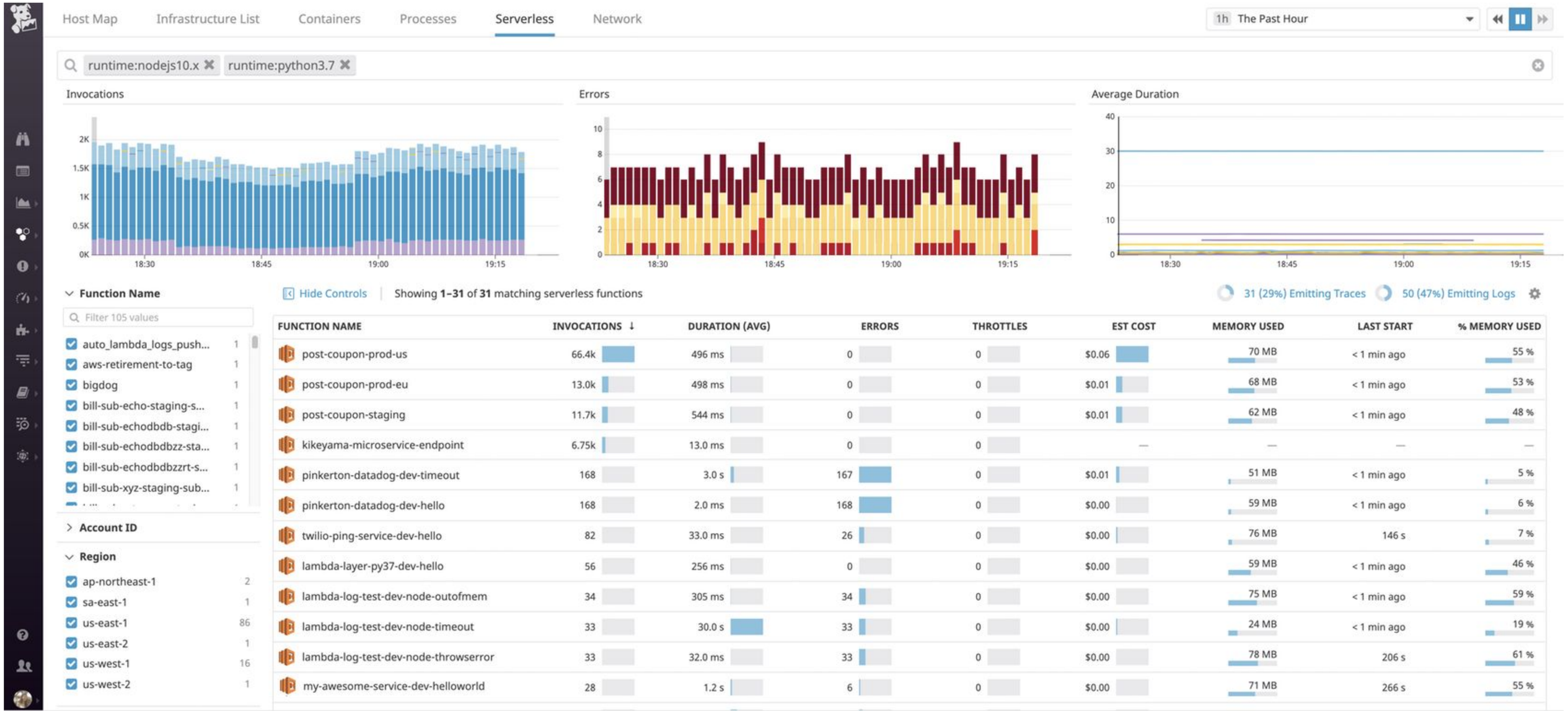
Details, please.

Lambda Memory vs. Execution Time (cost)

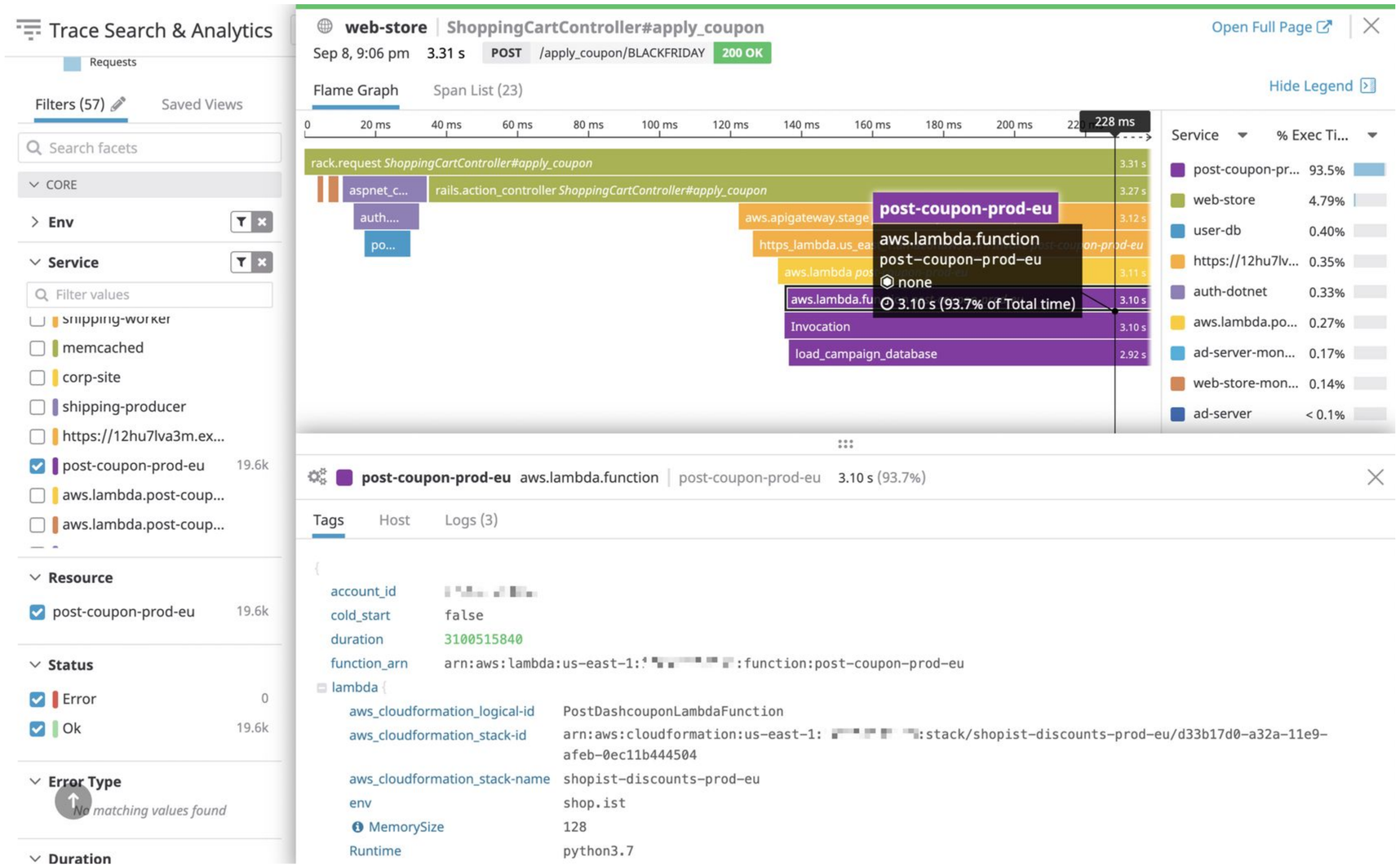



Cost per function call
0.003
0.002
0.001
0
128
256
512
768
1024
1280

Serverless Framework



Serverless Framework



"[We] do weekly reviews of our functions to make sure they are behaving correctly."

– Reza Javidi

That’s neat!

Come chat with
us at our booth!

DATADOG