



# Lambda Power Tuning

Una comparativa entre arquitecturas  
x86\_64 y arm64

Oscar Cortes



Hoy hablamos de...

### Contenidos de la charla

- AWS Lambda
- Lambda Power Tuning
- Laboratorio
- Resultados

### Sobre mí



**Oscar Cortes**

Cloud Architect

**Amazon community builder**

**AWS User Group Madrid Co-organizador**

[oscar.cortes@ackstorm.com](mailto:oscar.cortes@ackstorm.com)

[www.linkedin.com/in/linkedindelpONENTE](http://www.linkedin.com/in/linkedindelpONENTE)

[www.ackstorm.com](http://www.ackstorm.com)



# Amazon User Group Madrid



## Detalles

¡Feliz año nuevo! Os invitamos a participar en el próximo Meetup de AWS que tendrá lugar el 13 de febrero de 2025, a las 19 Horas. Los ponentes invitados realizarán **re:Cap AWS del AWS re:Invent 2024** repasando las principales novedades en IA y DATA

📍 Dirección: NTT DATA - Sala Madrid Río Novus Building, Cam. de la Fuente de la Mora, 1, Hortaleza, 28050 Madrid



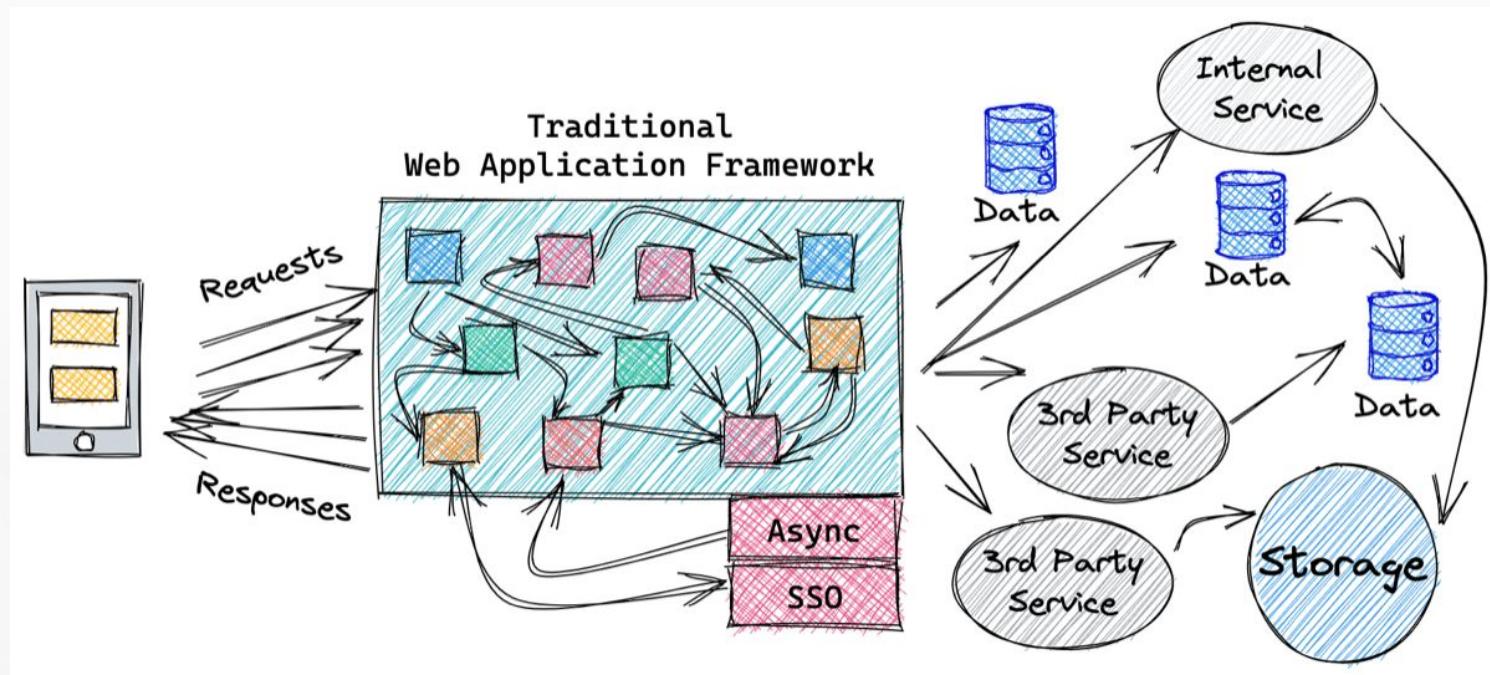
Hoy hablamos de...

## Contenidos de la charla

- AWS Lambda
- Lambda Power Tuning
- Laboratorio
- Resultados

- Quien utiliza AWS Lambda?
- En qué proyectos lo utilizan?
  - API
  - IA
  - Microservicios
  - ...

# AWS Lambda ?



<https://aws.amazon.com/es/what-is/serverless-computing/>

## AWS Lambda

API



Events



## AWS Lambda



# AWS Lambda

API



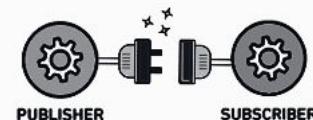
Events



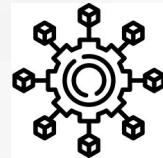
Request/  
Response



Publisher/  
Subscriber



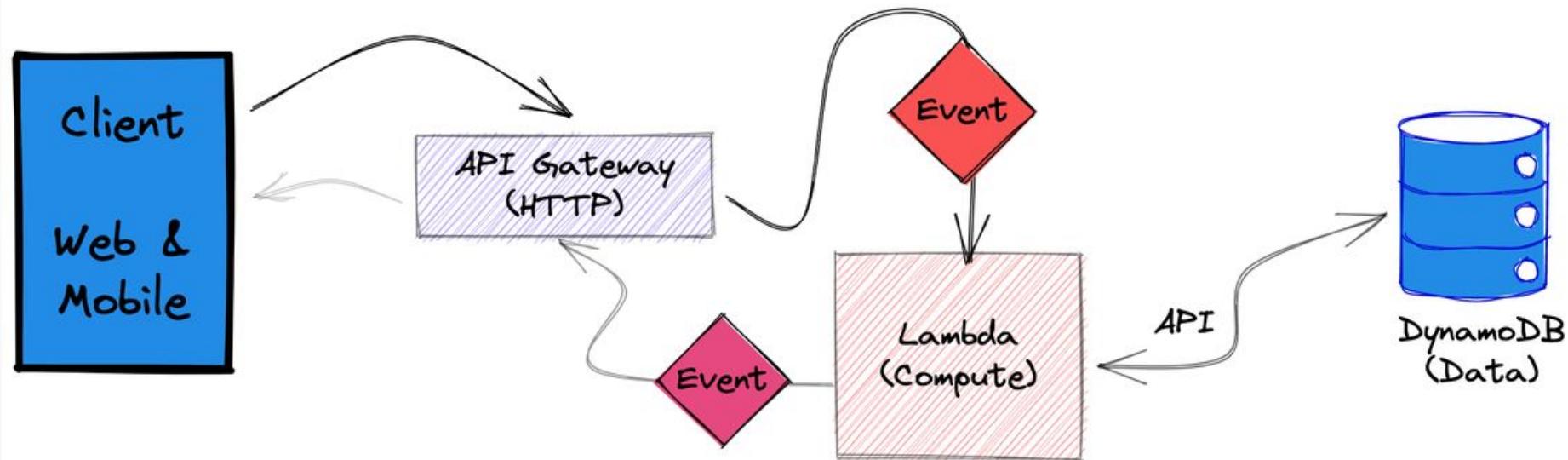
Microservicios



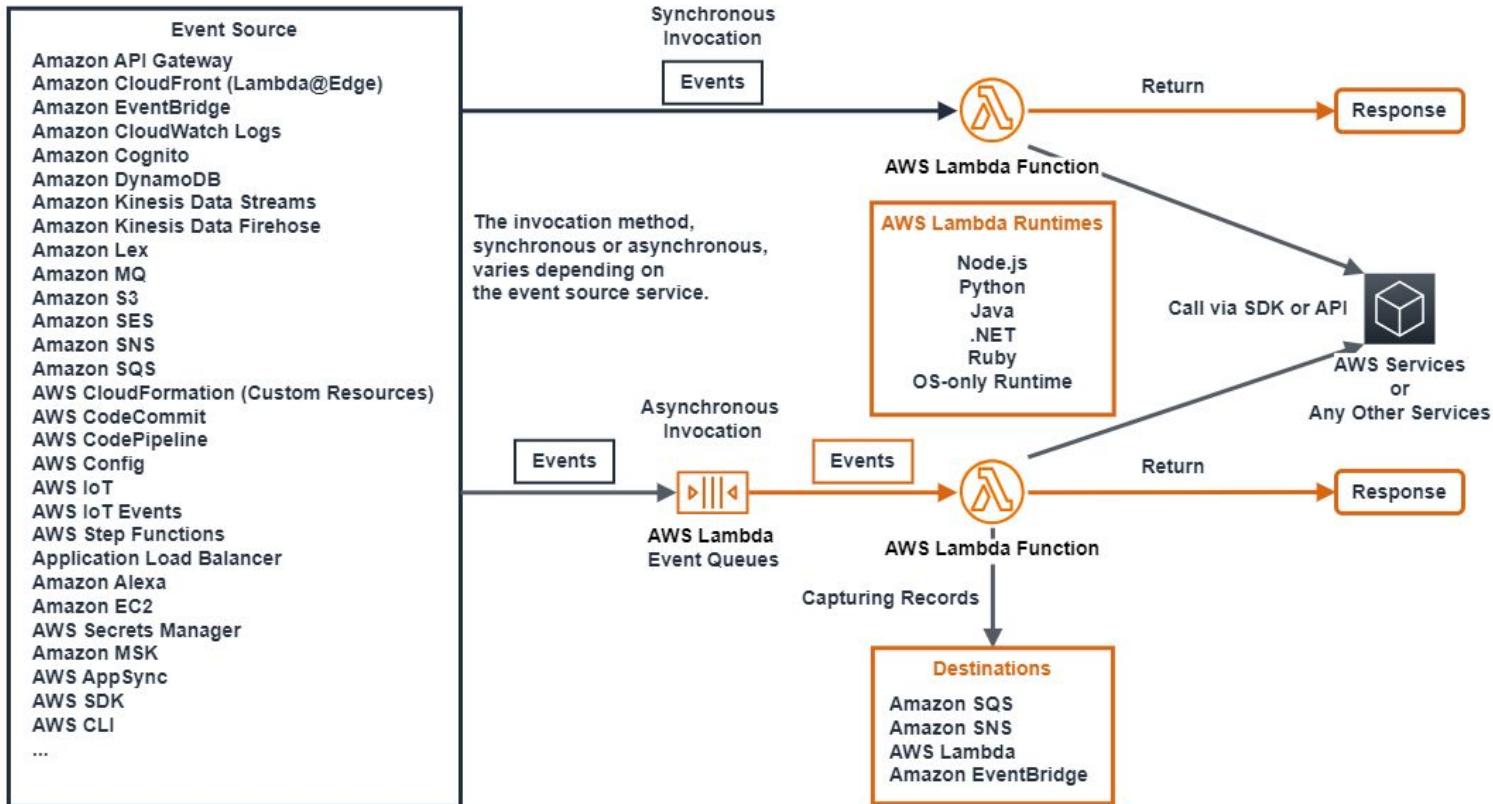
Functions



## AWS Lambda



<https://aws.amazon.com/es/what-is/serverless-computing/>



# 10 Years of Serverless with AWS Lambda and Amazon ECS

Move faster from idea to business value

[Learn more about AWS Lambda](#)

[Learn more about Amazon ECS](#)

## Serverless adoption today

### 1.5 Million

Over 1.5 million customers use AWS Lambda each month.

### Tens of Trillions

AWS Lambda processes tens of trillions of requests each month.

### 2.4 Billion

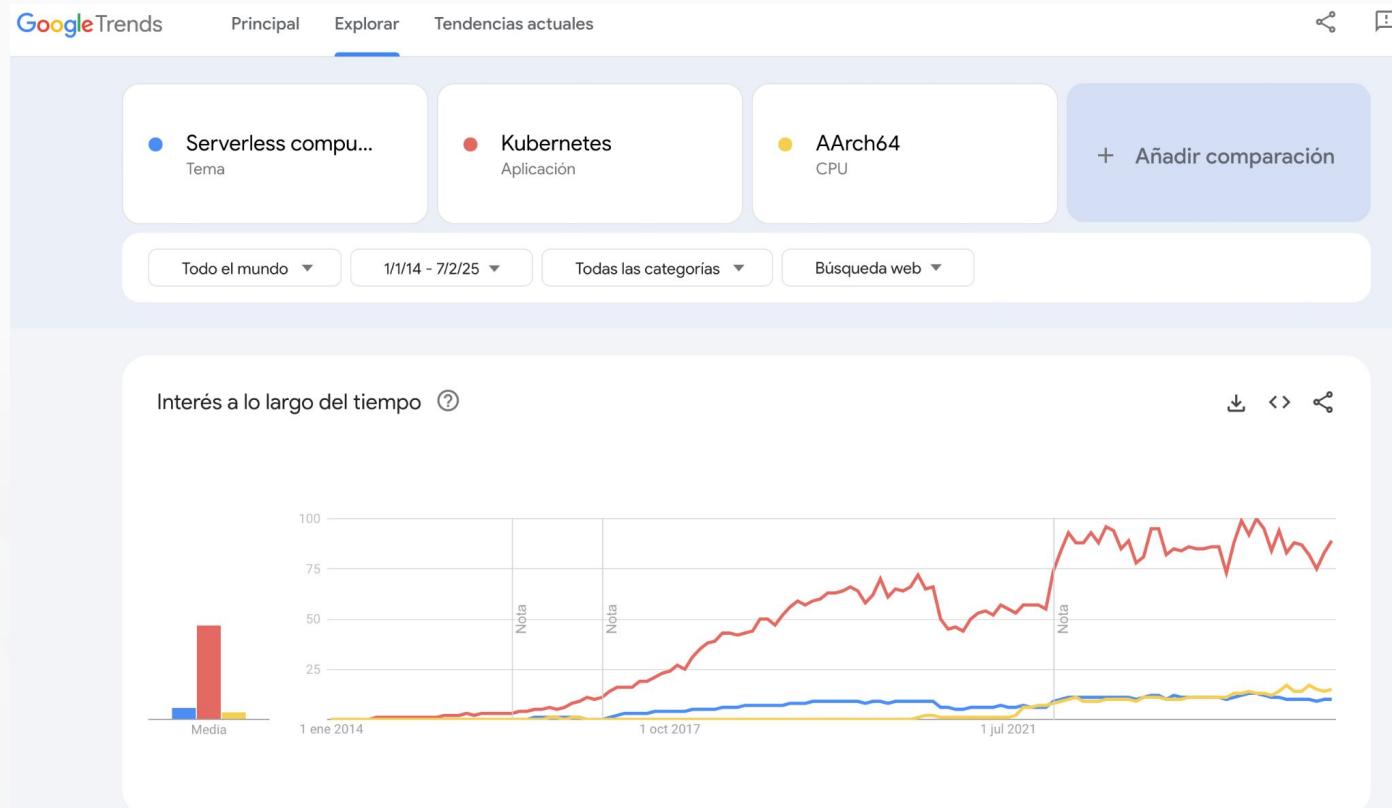
Over 2.4 billion Amazon Elastic Container Service tasks are launched each week.

### 65%

Over 65% of all new AWS container customers use Amazon Elastic Container Service.

<https://aws.amazon.com/es/serverless/10th-anniversary/>

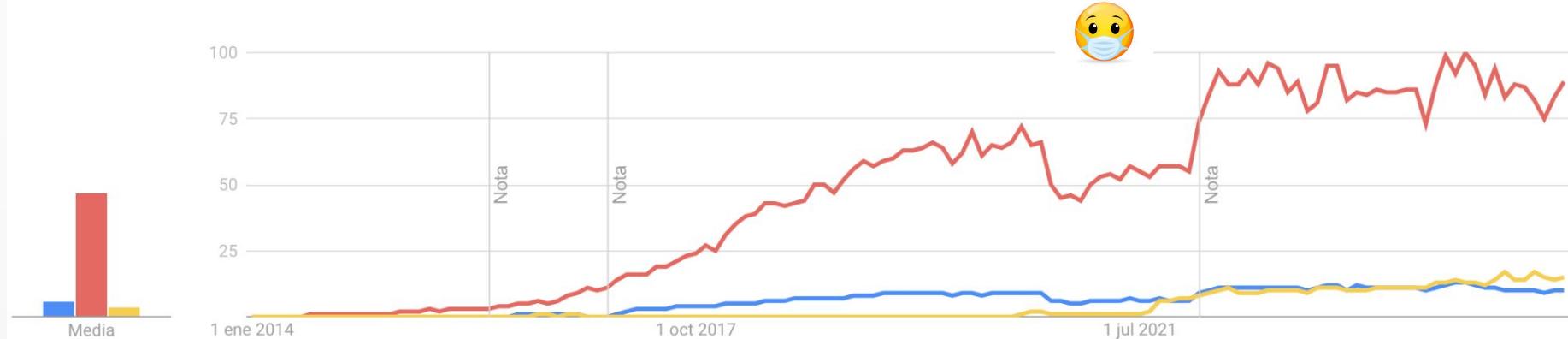
# AWS Lambda



[https://trends.google.es/trends/explore?date=2014-01-01%202025-02-07&q=%2Fg%2F11c0q\\_754d,%2Fg%2F11b7lxp79d,%2Fg%2F11fw9kg3g0&hl=es](https://trends.google.es/trends/explore?date=2014-01-01%202025-02-07&q=%2Fg%2F11c0q_754d,%2Fg%2F11b7lxp79d,%2Fg%2F11fw9kg3g0&hl=es)

# AWS Lambda

Interés a lo largo del tiempo ?



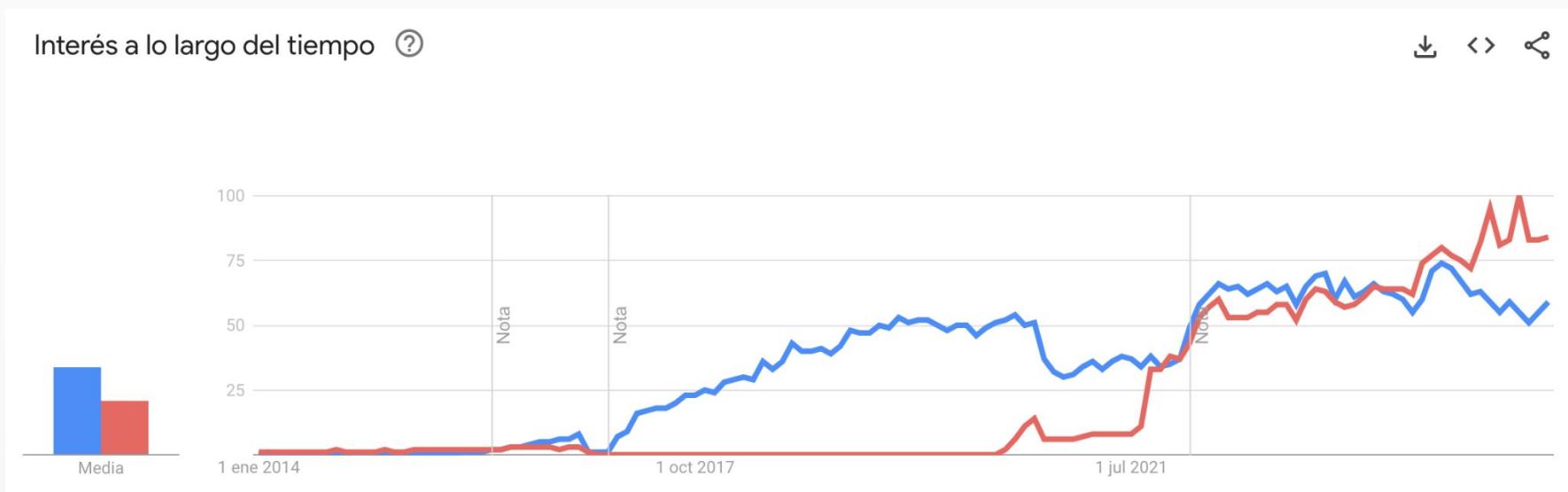
● Serverless compu...  
Tema

● Kubernetes  
Aplicación

⋮

● AArch64  
CPU

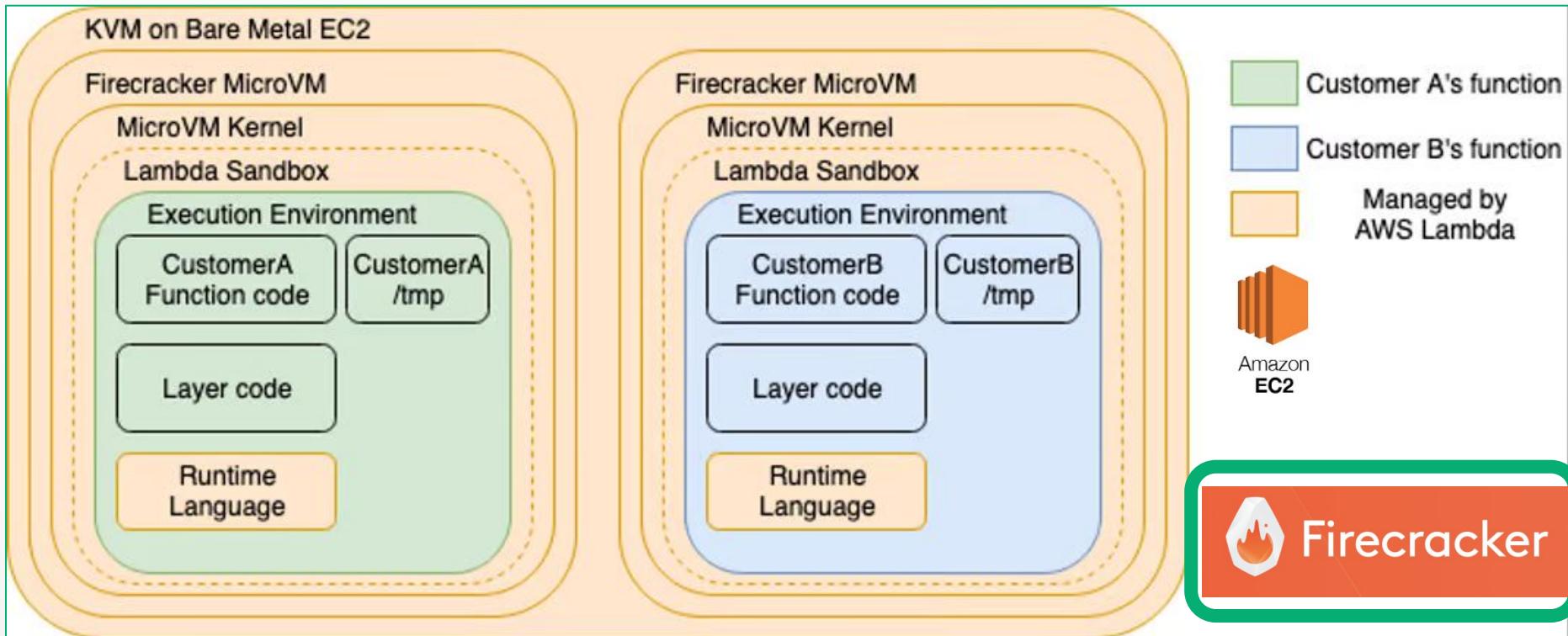
# AWS Lambda



# AWS Lambda



# AWS Lambda



## Memoria

Obtenga información sobre cómo y cuándo aumentar la memoria de una función.

## Almacenamiento efímero

Obtenga información sobre cómo y cuándo aumentar la capacidad de almacenamiento temporal de la función.

## Timeout (Tiempo de espera)

Obtenga información sobre cómo y cuándo aumentar el valor del tiempo de espera de la función.

## Variables de entorno

Puede hacer que el código de su función sea portátil y mantener los secretos fuera de su código almacenándolos en la configuración de la función mediante las variables de entorno.

## Redes salientes

Puede utilizar la función de Lambda con los recursos de AWS de una nube de Amazon VPC. La conexión de la función a una VPC le permite obtener acceso a los recursos de una red privada, como bases de datos relacionales y cachés.

## Redes entrantes

Puede utilizar un punto de conexión de VPC de tipo interfaz para invocar las funciones de Lambda sin pasar por la red pública de Internet.

## Sistema de archivos

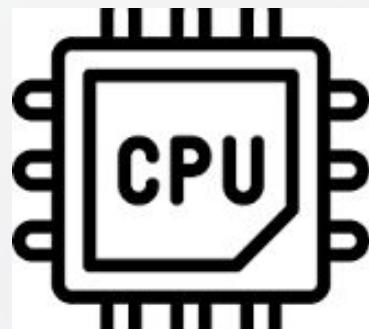
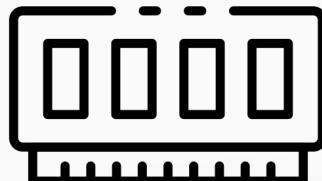
Puede usar la función de Lambda para montar un sistema de Amazon EFS en un directorio local. Un sistema de archivos permite al código de su función acceder a los recursos compartidos de forma segura y en alta simultaneidad, y también le permite modificarlos.

## Alias

Puede configurar sus clientes para invocar una versión específica de la función de Lambda mediante un alias en lugar de actualizar el cliente.

## Versiones

Al publicar una versión de su función, puede almacenar el código y la configuración como recursos independientes que no pueden modificarse.

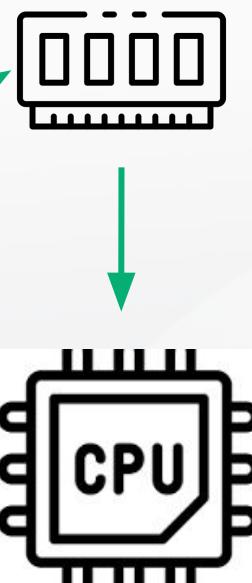


## Determinación de la configuración de memoria adecuada de una función de Lambda

La memoria es la palanca principal para controlar el rendimiento de una función. El valor predeterminado, 128 MB, es el más bajo posible. Solo se recomienda utilizar 128 MB para las funciones de Lambda sencillas, como las que transforman y enrutan eventos a otros servicios de AWS. Una mayor asignación de memoria puede mejorar el rendimiento de las funciones que utilizan bibliotecas importadas, [capas de Lambda](#), Amazon Simple Storage Service (Amazon S3) o Amazon Elastic File System (Amazon EFS). Agregar más memoria aumenta proporcionalmente la cantidad de CPU, lo que aumenta la potencia computacional general disponible. Si una función está vinculada a la CPU, la red o la memoria, aumentar la configuración de memoria puede mejorar drásticamente su rendimiento.

Para encontrar la configuración de memoria adecuada, supervise las funciones con Amazon CloudWatch y establezca alarmas que avisen en caso de que el consumo de memoria se acerque a los máximos configurados. Esto puede ayudar a identificar las funciones vinculadas a la memoria. En el caso de las funciones vinculadas a la CPU y a las operaciones de E/S, la supervisión de la duración también puede aportar información. En estos casos, aumentar la memoria puede ayudar a resolver los cuellos de botella informáticos o de red.

También puede considerar el uso de la herramienta de código abierto [AWS Lambda Power Tuning](#). Esta herramienta utiliza AWS Step Functions para ejecutar varias versiones simultáneas de una función de Lambda en diferentes asignaciones de memoria y medir el rendimiento. La función de entrada se ejecuta en su cuenta de AWS y realiza llamadas HTTP en tiempo real e interacciones con el SDK para medir el rendimiento probable en un escenario de producción activo. También puede implementar un proceso de CI/CD para usar esta herramienta y medir automáticamente el rendimiento de las nuevas funciones que implemente.





## Multithreading? Maybe!

- <1.8GB is still single core
  - CPU bound workloads won't see gains – processes share same resources
- >1.8GB is multi core
  - CPU bound workloads will gain, but need to multi thread
- I/O bound workloads WILL likely see gains
  - e.g. parallel calculations to return

AWS  
re:Invent

© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.



AWS  
re:Invent

aws  
aws

<https://www.youtube.com/watch?v=sSSMTSn2xmA&t=2256s>

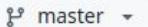
Hoy hablamos de...

## Contenidos de la charla

- AWS Lambda
- Lambda Power Tuning
- Laboratorio
- Resultados

 **aws-lambda-power-tuning** Public

 Watch 83  Fork 386  Starred 5.6k

 master  6 Branches  0 Tags  Go to file  Add file  Code

Commit	Message	Date
 53bf280	Merge pull request #262 from patmeiler/documentation-r...	3 months ago
	Upgrade GHA coveralls v2	10 months ago
	bumped to version 4.3.6	3 months ago
	update state machine screenshot (new ASL)	5 months ago
	Add tests for extractDuration	9 months ago
	Bump fast-xml-parser, @aws-sdk/client-lambda and @aw...	5 months ago
	bumped to version 4.3.6	3 months ago

**About**

AWS Lambda Power Tuning is an open-source tool that can help you visualize and fine-tune the memory/power configuration of Lambda functions. It runs in your own AWS account - powered by AWS Step Functions - and it supports three optimization strategies: cost, speed, and balanced.

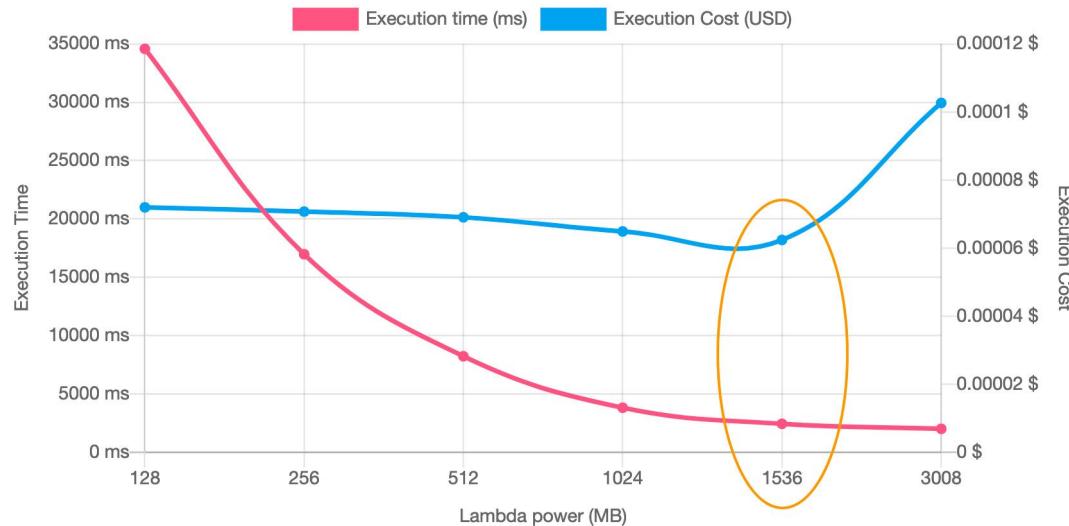
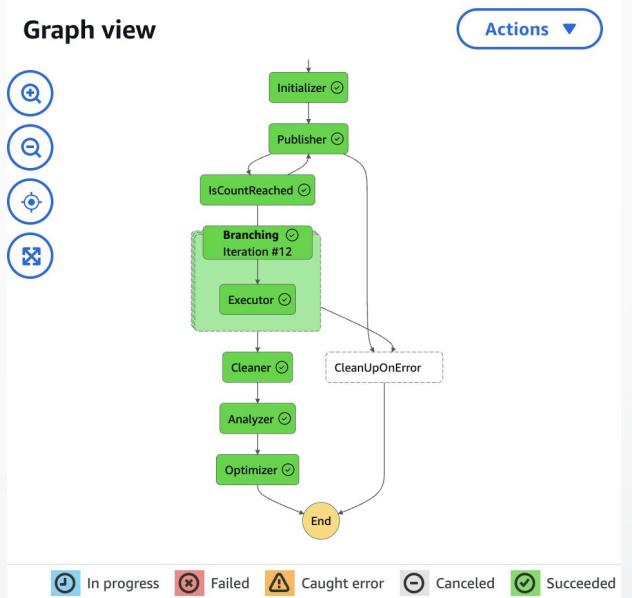
     
    

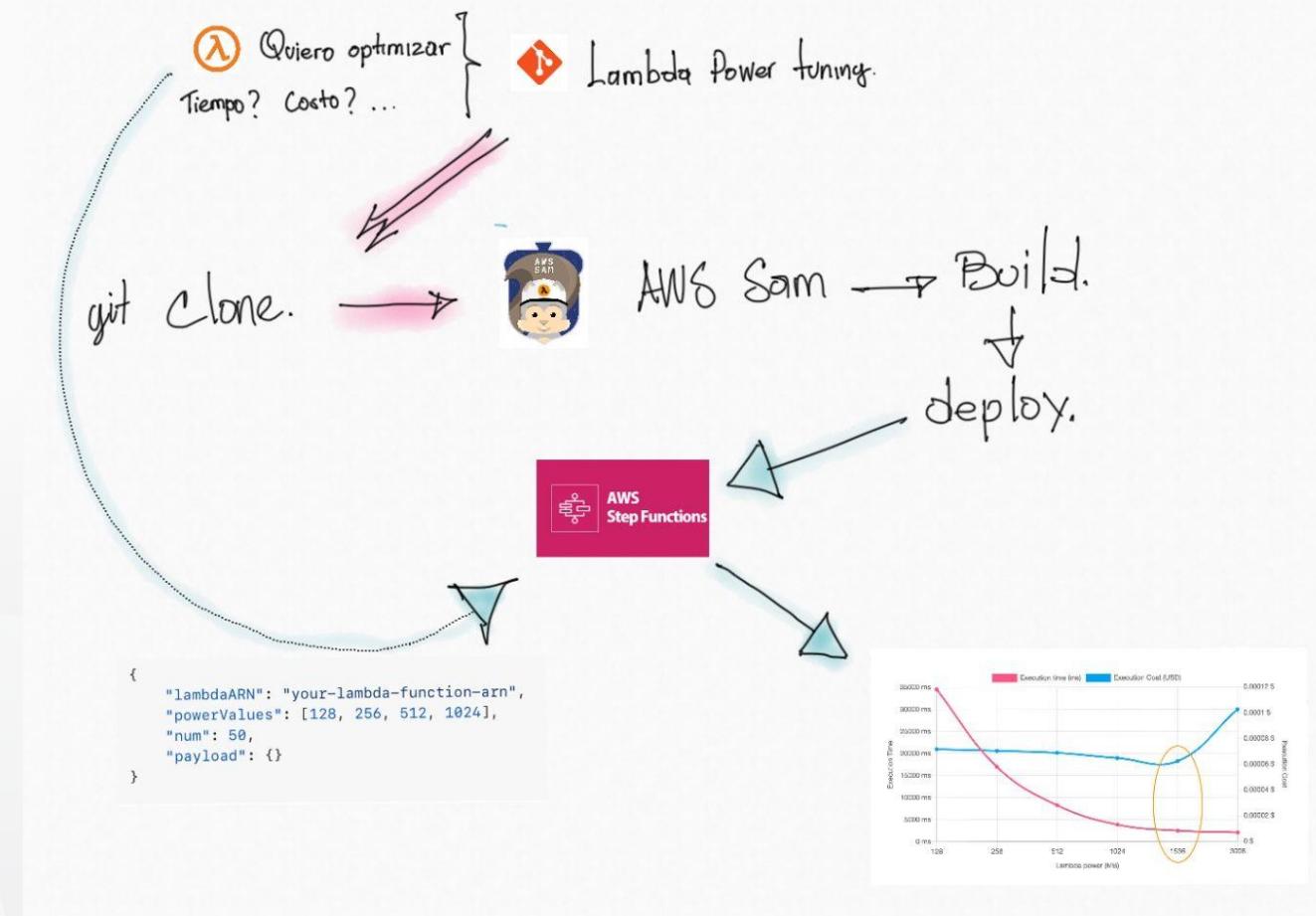

<https://github.com/alexcasalboni/aws-lambda-power-tuning?tab=readme-ov-file>

# Lambda Power Tuning

<https://github.com/alexcasalboni/aws-lambda-power-tuning?tab=readme-ov-file>

## Graph view





Express (17)							
Step-FinishDate		From	To	Reason	ReasonAfter	Timestamp	
Initial creation	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Developer assigned	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Code Development	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
User Acceptance	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Peer review	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (1)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (2)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (3)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (4)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (5)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (6)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (7)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (8)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (9)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (10)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (11)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (12)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (13)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (14)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (15)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (16)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	
Deployment (17)	2023-04-18 10:00:00	Initial creation	None	None	None	2023-04-18 10:00:00	

Hoy hablamos de...

## Contenidos de la charla

- AWS Lambda
- Lambda Power Tuning
- **Laboratorio**
- Resultados

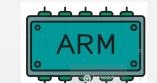
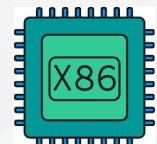
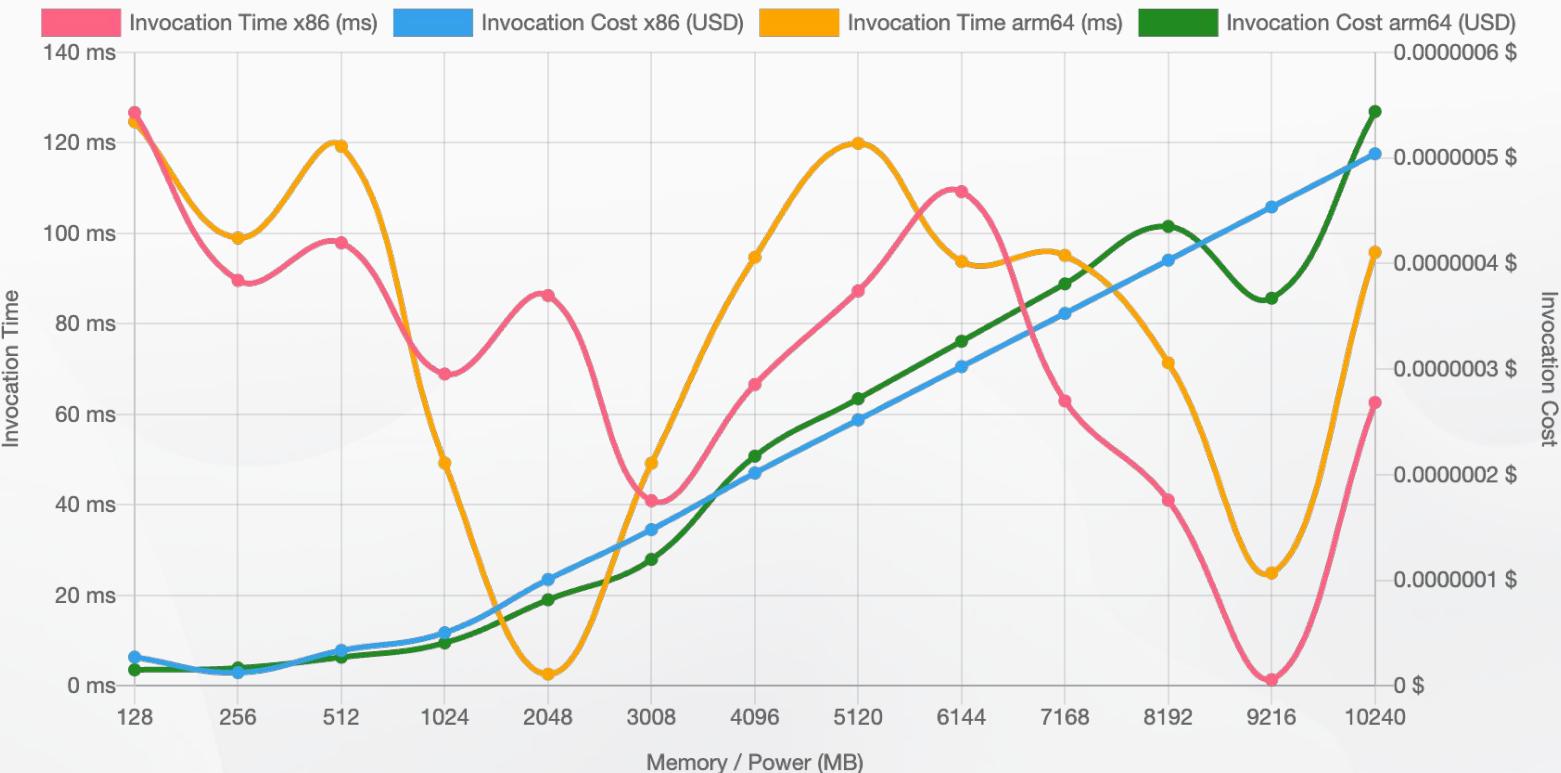
## hello world x86 vs arm64 Node22



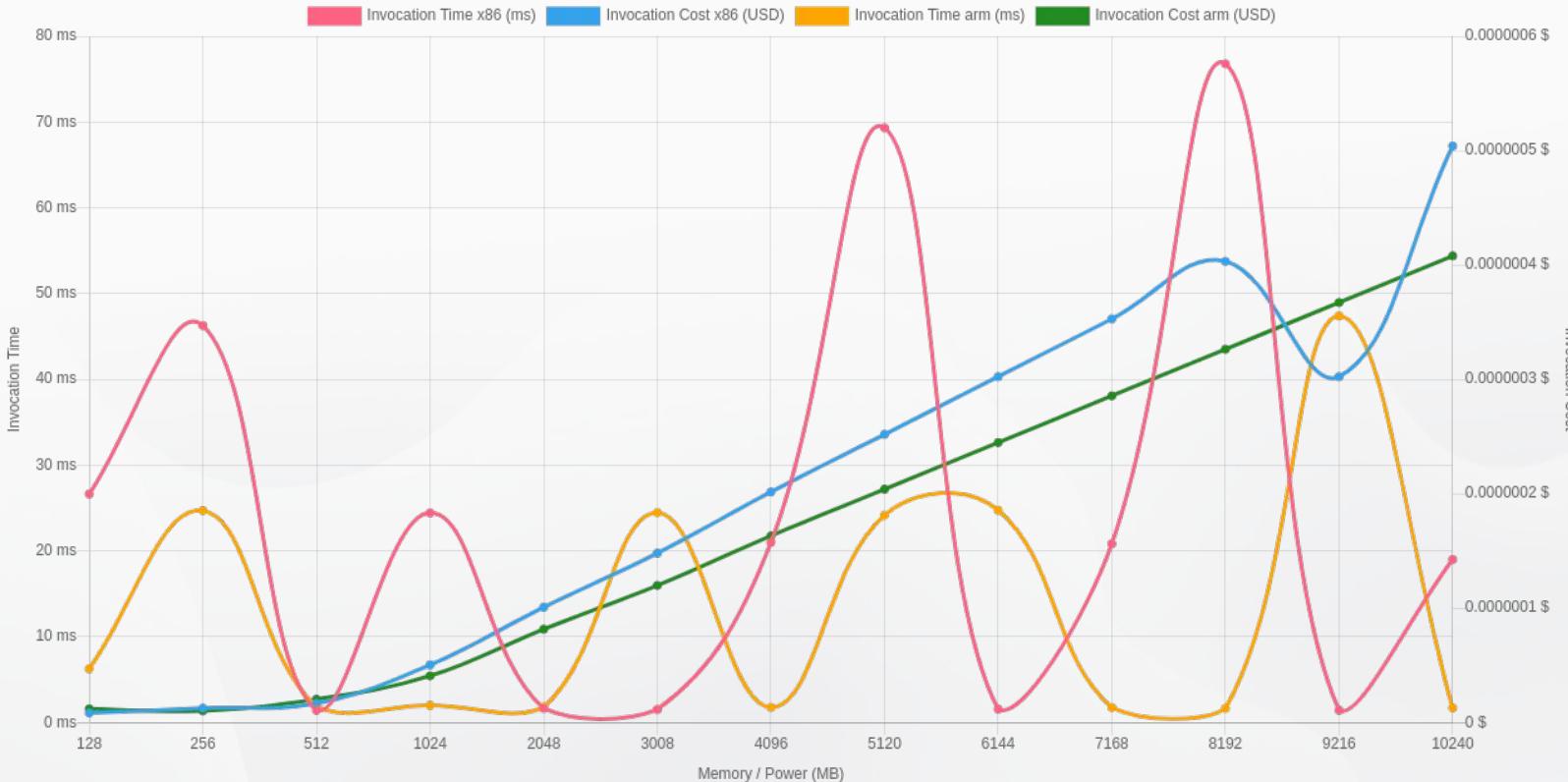
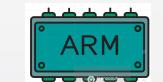
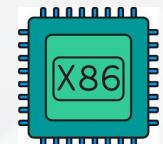
```
export const handler = async (event) => {
  // TODO implement
  const response = {
    statusCode: 200,
    body: JSON.stringify('Hello from Lambda!'),
  };
  return response;
};
```



# hello world x86 vs arm64 Node22



# hello world x86 vs arm64 Node22-10-provisioned



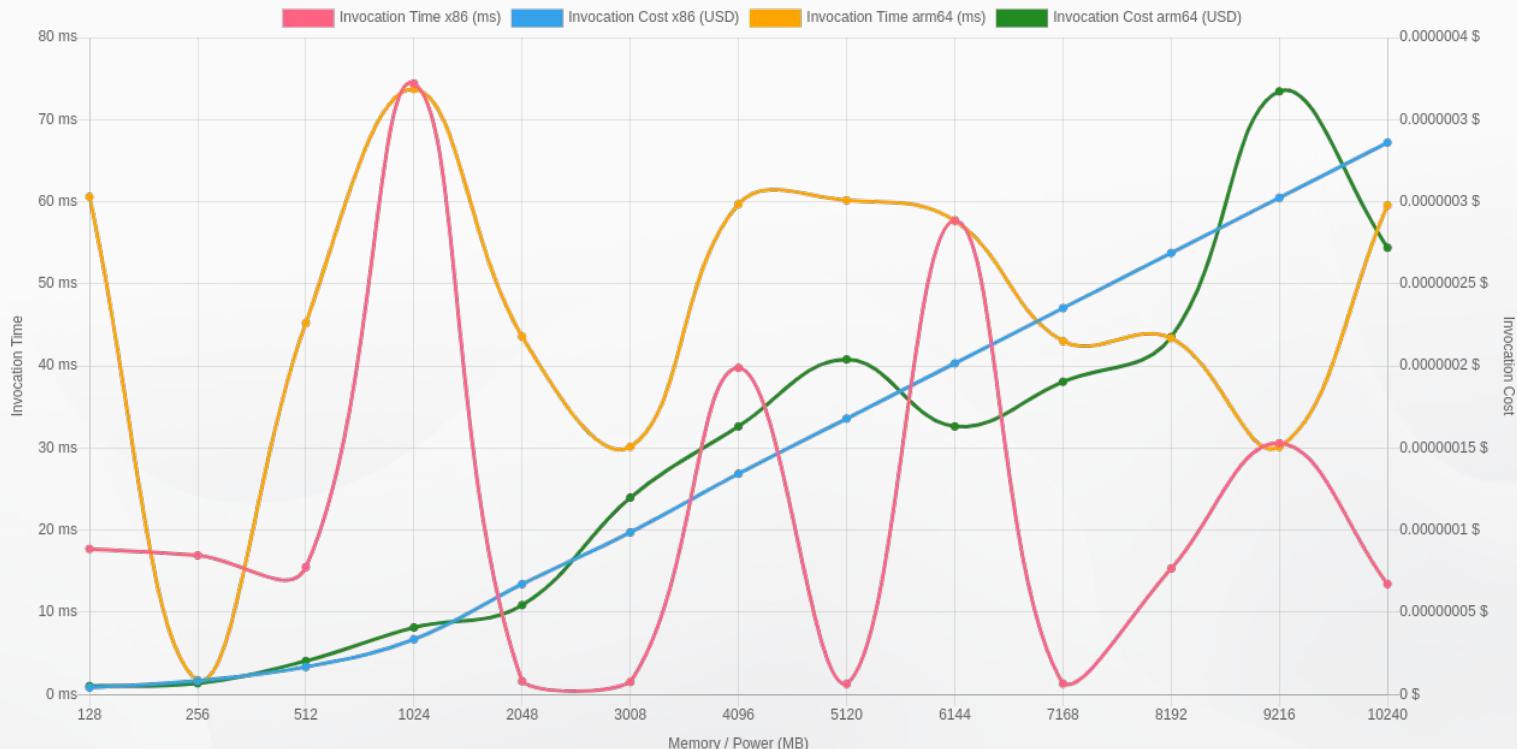
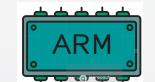
# hello world x86 vs arm64 Python



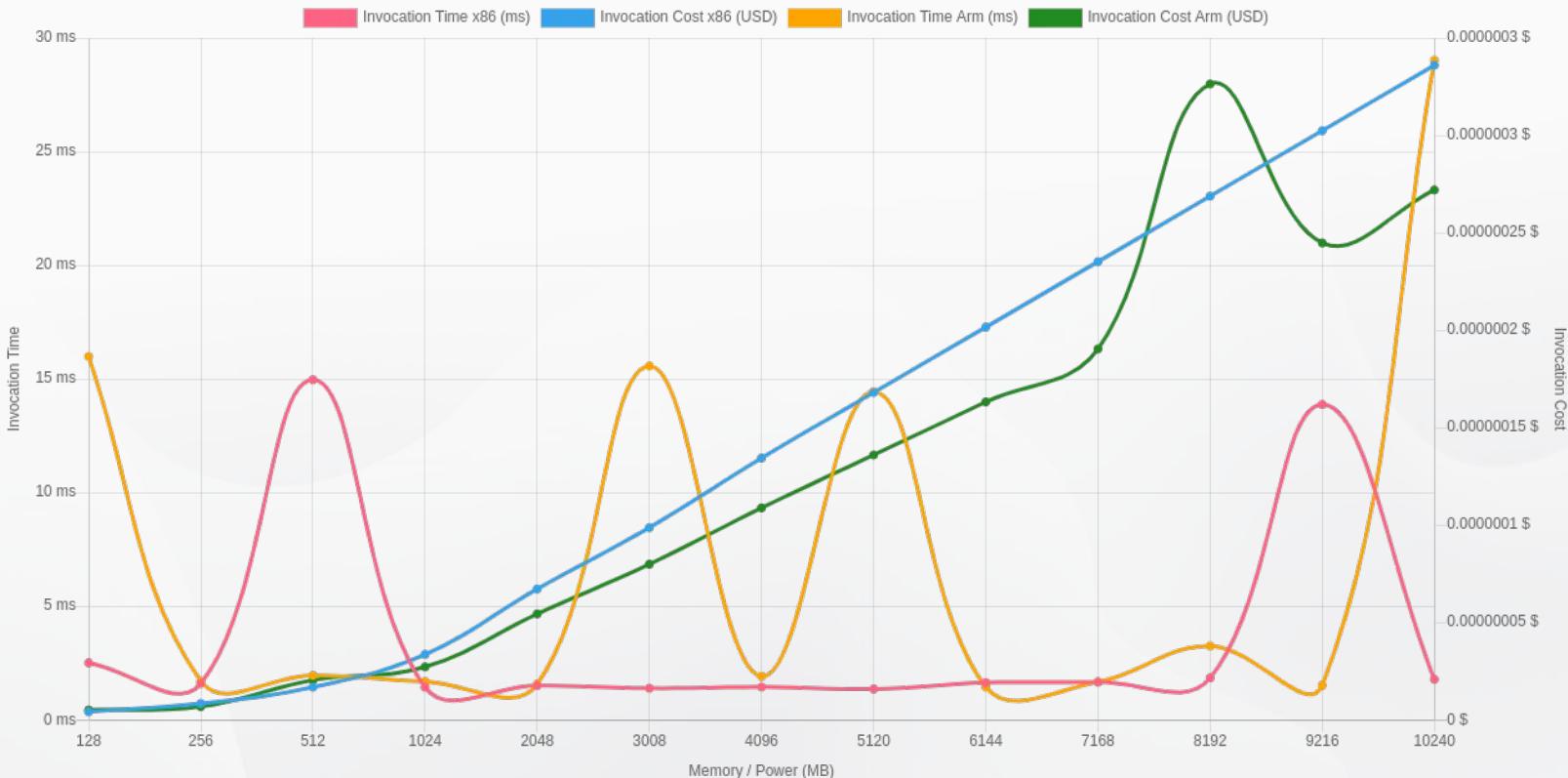
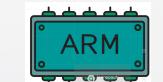
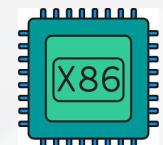
```
lambda_function.py X
lambda_function.py
1 import json
2
3 def lambda_handler(event, context):
4     # TODO implement
5     return {
6         'statusCode': 200,
7         'body': json.dumps('Hello from Lambda!')
8     }
```

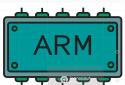


# hello world x86 vs arm64 Python



# hello world x86 vs arm64 Python - 10-provisioned





AWS Container Images search results for 'lambda' showing the AWS Lambda Python base image.

**Filters:** Clear all, Operating Systems (Linux, Windows), Architectures (ARM, ARM 64, x86, x86-64).

**Repositories:** Showing 1 - 7 results (of 7)

- lambda/python** (132M+ downloads) - AWS Lambda base images for Python. OS/Arch: Linux, ARM 64, x86-64.



Repositories > arm-images-lambda

ⓘ Image scan overview, status, and full vulnerabilities has moved to the Image detail page. To access,

## Images (2)

Search artifacts

	Image tag	Artifact type	Pushed at
<input type="checkbox"/>	latest	Image	07 de febrero de 2025, 12:50:01 (UTC+01)

## Image

ⓘ No code preview available

Your function code is deployed as a container image. The IDE cannot display your code.

### Image URI

536395113578.dkr.ecr.eu-central-1.amazonaws.com/arm-images-lambda@sha256:0590763a8e624f6aa5d1d070b5860257a6d224d27d982c706b082f0cc4b50609

Architecture Info  
arm64



Choose one of the following options to create your function.

Author from scratch  
Start with a simple Hello World example.

Use a blueprint  
Build a Lambda application from sample code and configuration presets for common use cases.

Container image  
Select a container image to deploy for your function.

### Basic Information

Function name  
Enter a name that describes the purpose of your function.

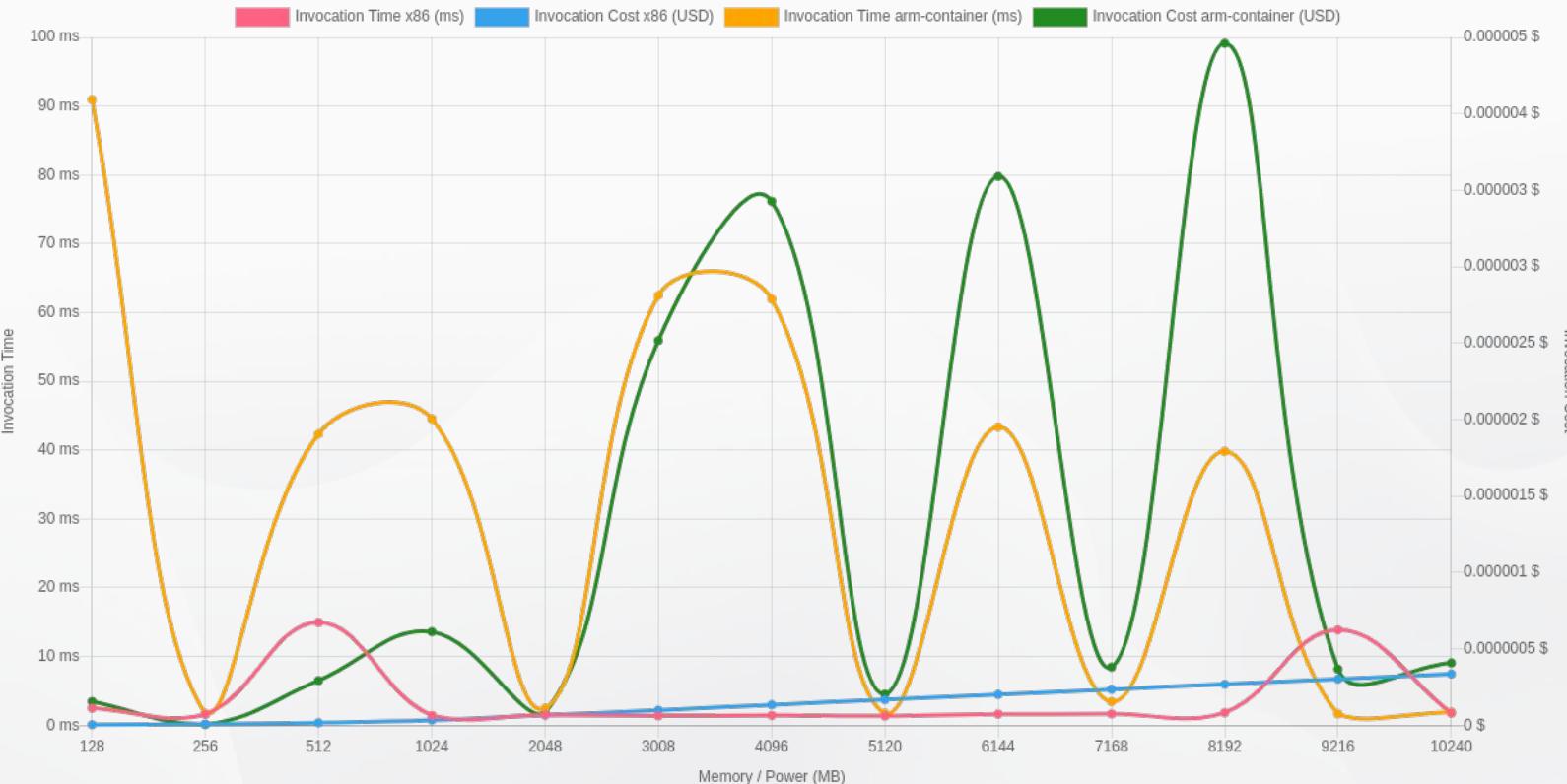
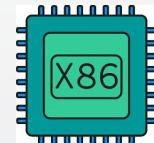
Container image URI  
Info  
The location of the container image to use for your function.

Container image overrides

Architecture Info  
Choose the instruction set architecture you want for your function code.  
 x86\_64  
 arm64



# hello world x86 vs arm64 Python - 10-provisioned - Container

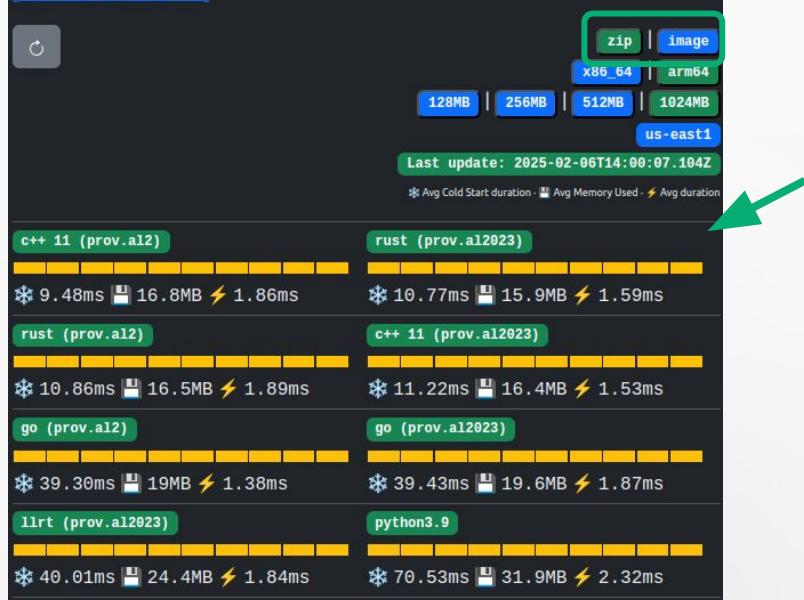


# Lambda Cold Starts benchmark by [maxday](#).

Visualize 10 Cold Starts for each runtime, updated daily.

[[How to deploy a Rust Lambda function?](#)]

[[How does it work?](#)]

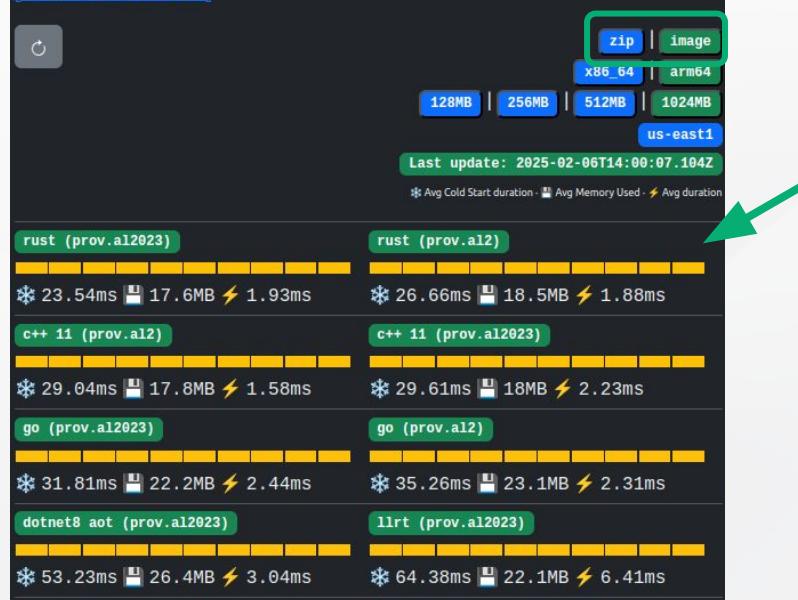


# Lambda Cold Starts benchmark by [maxday](#).

Visualize 10 Cold Starts for each runtime, updated daily.

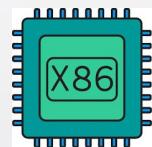
[[How to deploy a Rust Lambda function?](#)]

[[How does it work?](#)]



# Laboratorio

Algoritmo	Memoria	Input	Strategy
Criba Eratóstenes	All range	n=1..X	Cost
Fibonacci	All range	n=1..X	Cost
mimic	All range	body json	Cost



Executions | Monitoring | Logging | Definition | AI

### Executions (0/63)

Filter executions by property or value

<input type="checkbox"/> Name	Status
<a href="#">hello-world-python-13-arm-container10...</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python-13-arm-container10...</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-container-10R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-contaliner</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-amr64-10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-x86-10-R</a>	<span style="color: green;">✓ Succeeded</span>

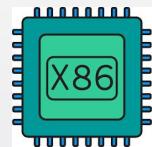
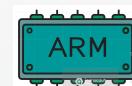
Hoy hablamos de...

## Contenidos de la charla

- AWS Lambda
- Lambda Power Tuning
- Laboratorio
- **Resultados**

# Laboratorio

Algoritmo	Memoria	Input	Strategy
Fibonacci	All range	n=1..100	Cost
Criba Eratóstenes	All range	n=1..X	Cost
mimic	All range	body json	Cost



Executions | Monitoring | Logging | Definition | AI

### Executions (0/63)

Filter executions by property or value

<input type="checkbox"/> Name	Status
<a href="#">hello-world-python-13-arm-container10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python-13-arm-container10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-container-10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-contalner</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-amr64-10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-x86-10-R</a>	<span style="color: green;">✓ Succeeded</span>

# fibonacci

```
import json

def fibonacci(n):
    if n < 2:
        return n
    else:
        # fn = fn-1 + fn-2
        return fibonacci(n-1) + fibonacci(n-2)

def lambda_handler(event, context):
    return {
        'statusCode': 200,
        'body': fibonacci(event['value'])
    }
```

## Execution: fibonacci\_x86\_64\_100

Details Execution input and output Definition

### Execution status

Succeeded

### Execution type

Standard

### Execution ARN

arn:aws:states:eu-central-1:536395113578:execution:powerTuningStateMachine-af7807b0-c433-11ef-9b68-06def10dc923:fibonacci\_x86\_64\_100

### IAM role ARN

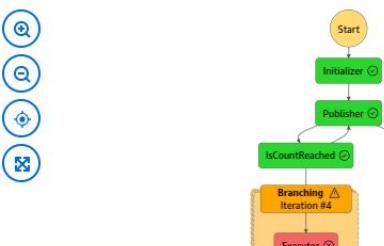
arn:aws:iam::536395113578:role/arcos-test-lambda-power-tuning-statemachineRole-QW8Kjjb44WgL

### State transitions Learn more

37

Graph view Table view

### Graph view



# fibonacci\_20\_all\_memory\_range x86 vs arm64

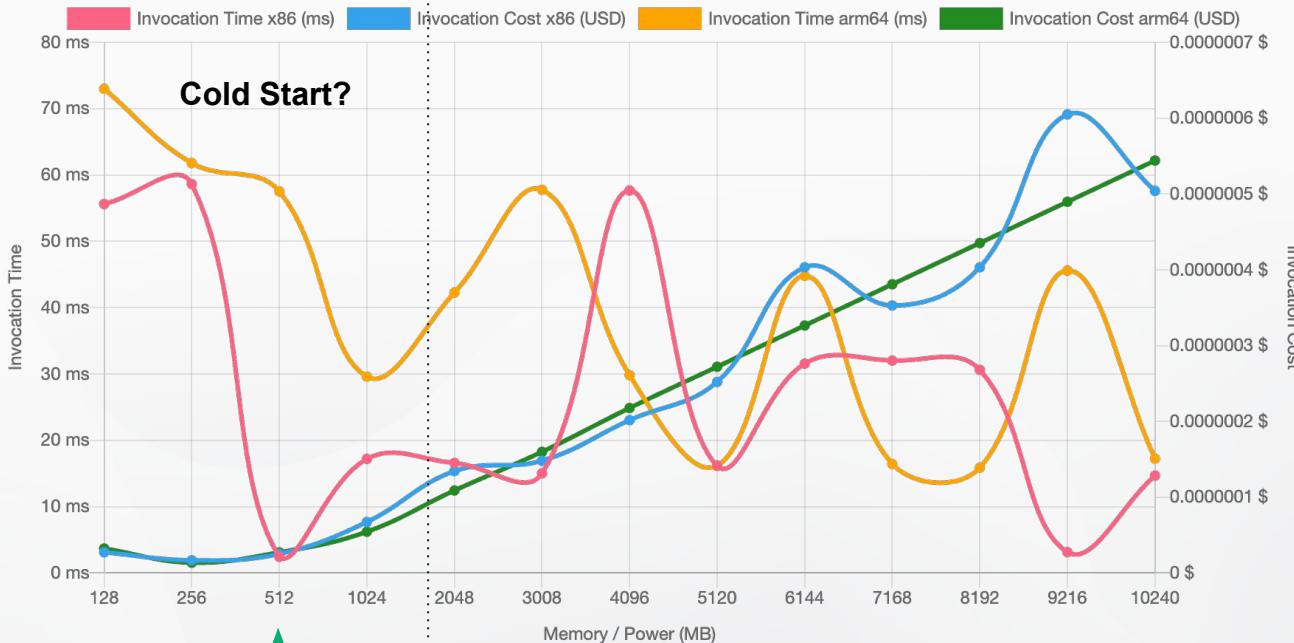


Tabla: <https://newsletter.simpleaws.dev/p/aws-lambda-underlying-architecture>

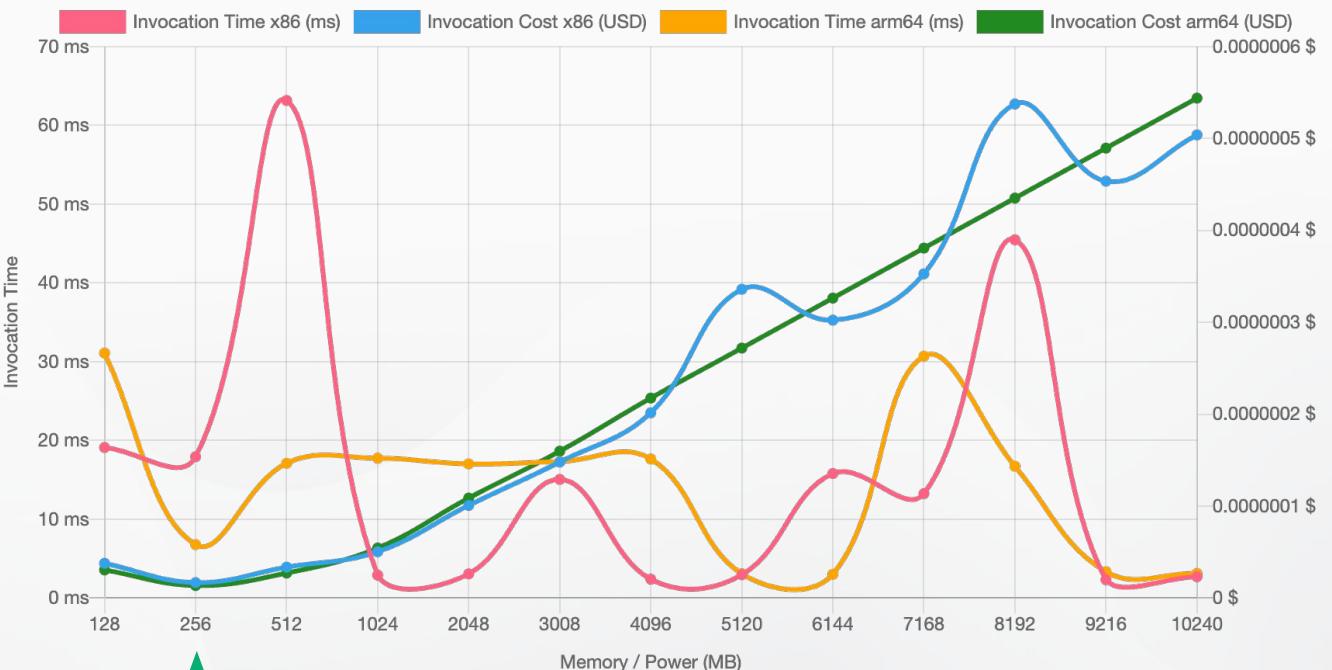


Memory	vCPUs	CPU Ceiling
832 MB	2	0.50
1769 MB	2	1.00
3008 MB	2	1.67
3009 MB	3	1.70
5307 MB	3	2.39
5308 MB	4	2.67
7076 MB	4	2.84
7077 MB	5	3.86
8845 MB	5	4.23
8846 MB	6	4.48
10240 MB	6	4.72

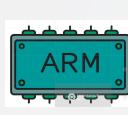
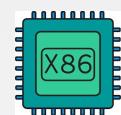
Multithreading? Maybe!

- <1.8GB is still single core
  - CPU bound workloads won't see gains – processes share same resources
- >1.8GB is multi core
  - CPU bound workloads will gain, but need to multi thread
  - I/O bound workloads WILL likely see gains
    - e.g. parallel calculations to return

# fibonacci\_20\_all\_memory\_range x86 vs arm64\_concurrency\_10

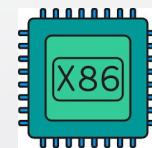
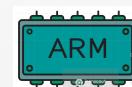


Memory	vCPUs	CPU Ceiling
832 MB	2	0.50
1769 MB	2	1.00
3008 MB	2	1.67
3009 MB	3	1.70
5307 MB	3	2.39
5308 MB	4	2.67
7076 MB	4	2.84
7077 MB	5	3.86
8845 MB	5	4.23
8846 MB	6	4.48
10240 MB	6	4.72



# Laboratorio

Algoritmo	Memoria	Input	Strategy
Criba Eratóstenes	All range	n=1..X	Cost
Fibonacci	All range	n=1..100	Cost
mimic	All range	body json	Cost



Executions | Monitoring | Logging | Definition | AI

### Executions (0/63)

Filter executions by property or value

<input type="checkbox"/> Name	Status
<a href="#">hello-world-python-13-arm-container10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python-13-arm-container10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-container-10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-contaliner</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-amr64-10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-x86-10-R</a>	<span style="color: green;">✓ Succeeded</span>

# criba eratóstenes

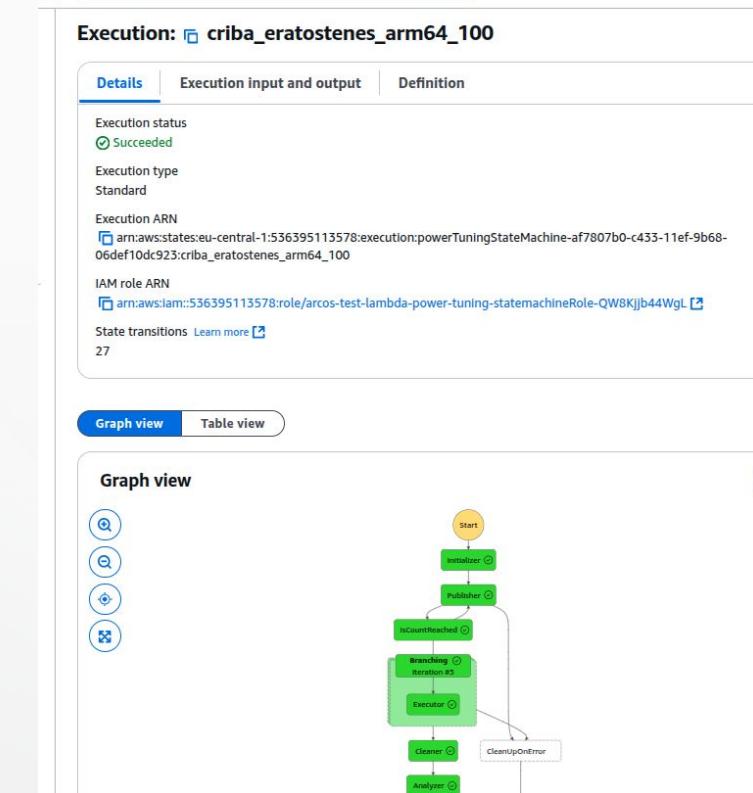
```
import json
import time

def criba_eratostenes(limite):
    primos = [True] * (limite + 1)
    primos[0] = primos[1] = False

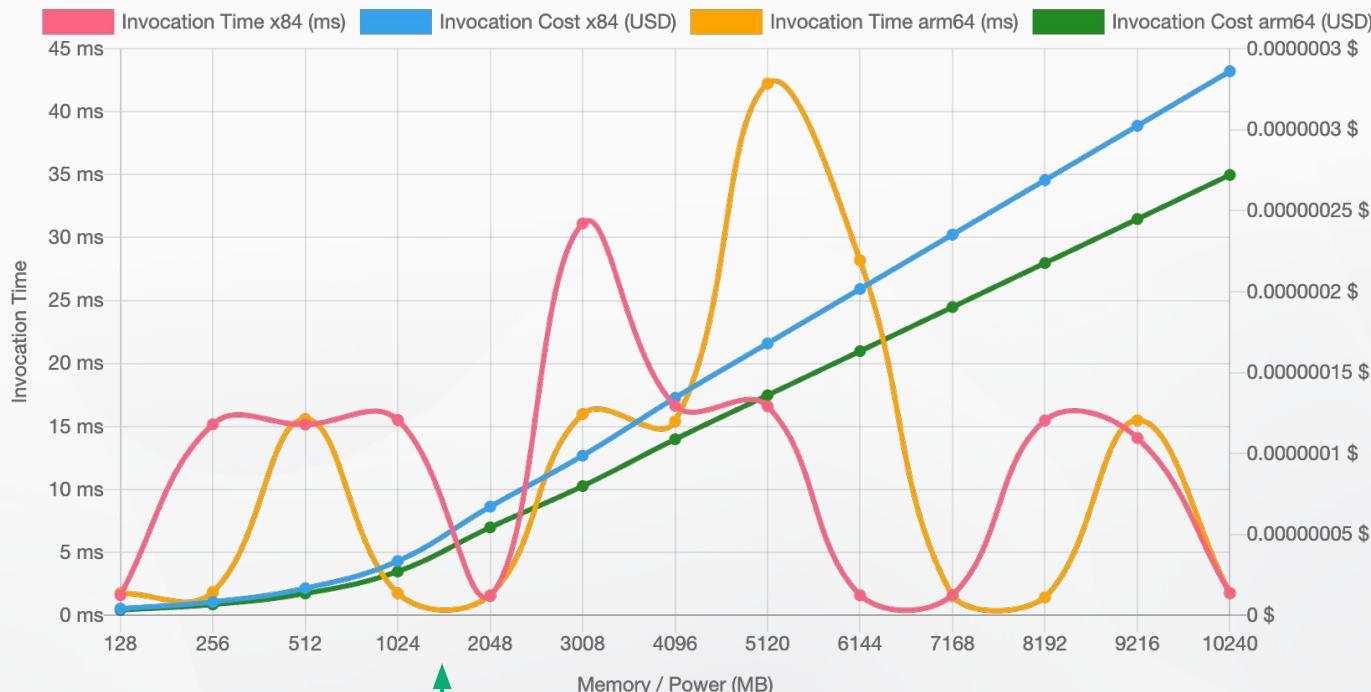
    for i in range(2, int(limite ** 0.5) + 1):
        if primos[i]:
            for j in range(i * i, limite + 1, i):
                primos[j] = False

    return [x for x in range(limite + 1) if primos[x]]

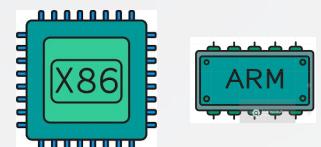
def lambda_handler(event, context):
    return {
        'statusCode': 200,
        'body': criba_eratostenes(event['value'])
    }
```



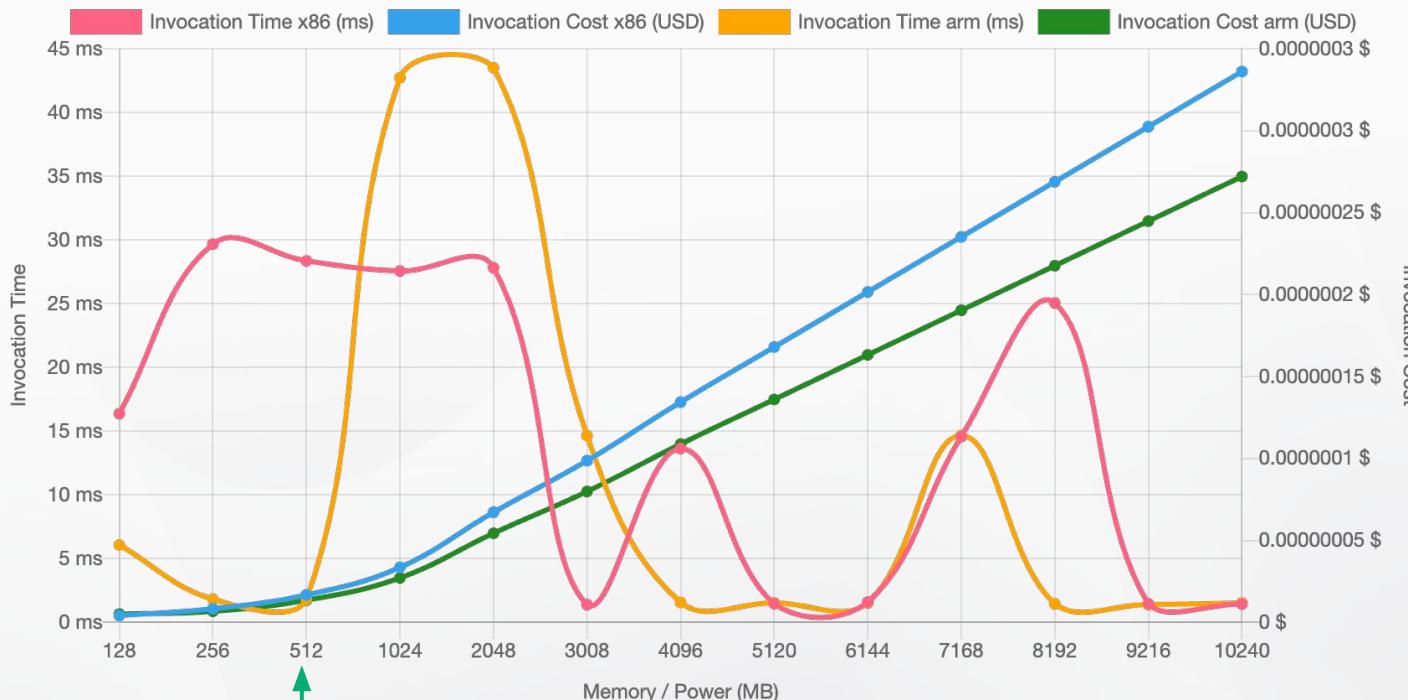
# criba\_eratóstenes\_50\_all\_memory\_range x86 vs arm64\_10\_R



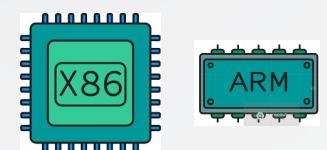
Memory	vCPUs	CPU Ceiling
832 MB	2	0.50
1769 MB	2	1.00
3008 MB	2	1.67
3009 MB	3	1.70
5307 MB	3	2.39
5308 MB	4	2.67
7076 MB	4	2.84
7077 MB	5	3.86
8845 MB	5	4.23
8846 MB	6	4.48
10240 MB	6	4.72



# criba\_ératóstenes\_50\_all\_memory\_range x86 vs arm64\_10\_R

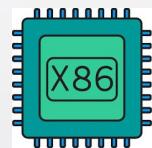


Memory	vCPUs	CPU Ceiling
832 MB	2	0.50
1769 MB	2	1.00
3008 MB	2	1.67
3009 MB	3	1.70
5307 MB	3	2.39
5308 MB	4	2.67
7076 MB	4	2.84
7077 MB	5	3.86
8845 MB	5	4.23
8846 MB	6	4.48
10240 MB	6	4.72



# Laboratorio

Algoritmo	Memoria	Input	Strategy
Criba Eratóstenes	All range	n=1..X	Cost
Fibonacci	All range	n=1..X	Cost
mimic	All range	body json	Cost



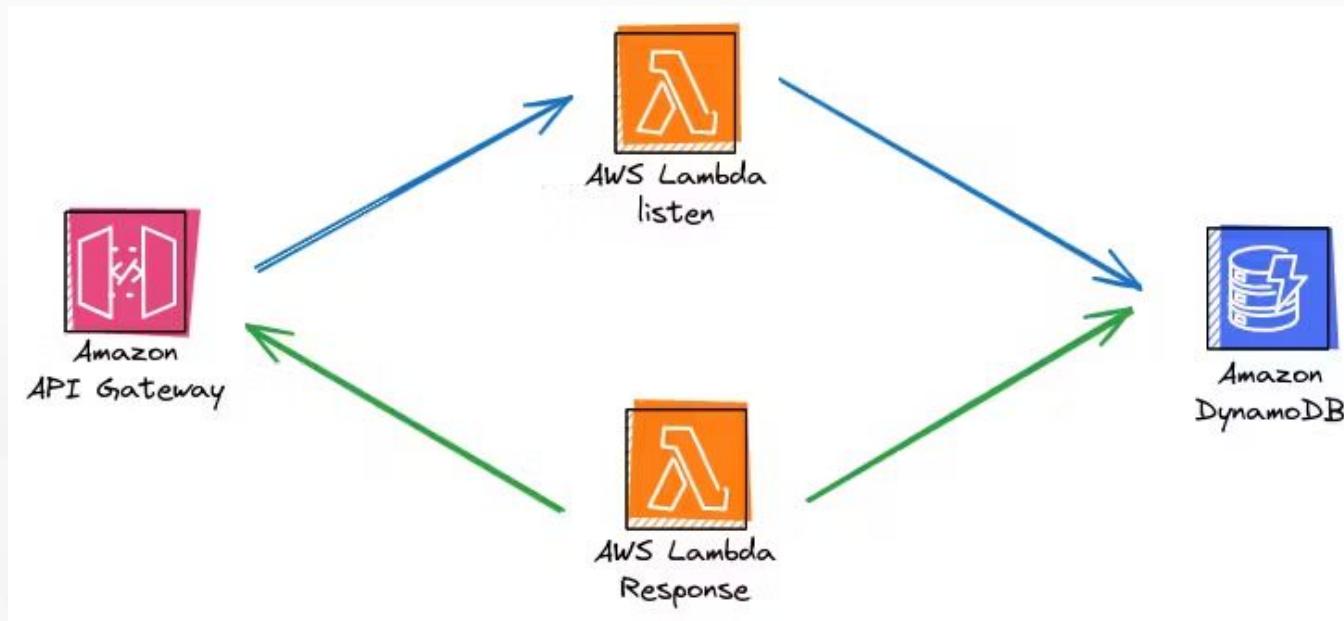
Executions | Monitoring | Logging | Definition | AI

### Executions (0/63)

Filter executions by property or value

<input type="checkbox"/> Name	Status
<a href="#">hello-world-python-13-arm-container10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python-13-arm-container10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-container-10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python13-arm-contaliner</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-amr64-10-R</a>	<span style="color: green;">✓ Succeeded</span>
<a href="#">hello-world-python3-13-x86-10-R</a>	<span style="color: green;">✓ Succeeded</span>

# AWS Lambda



# mimic

```
'use strict';

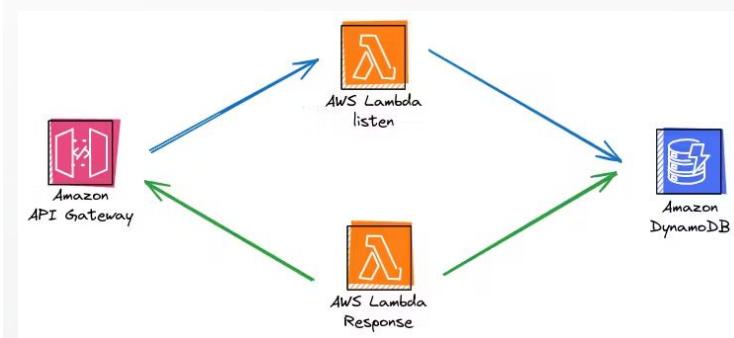
const dynamoService = require('./services/dbServices.js');
const { SCHEMAS_NAMES, schemaValidate } = require('./validations/index.js');

exports.lambdaHandler = async (event, context) => {
  try {
    const { path, queryStringParameters, headers, body } = event;

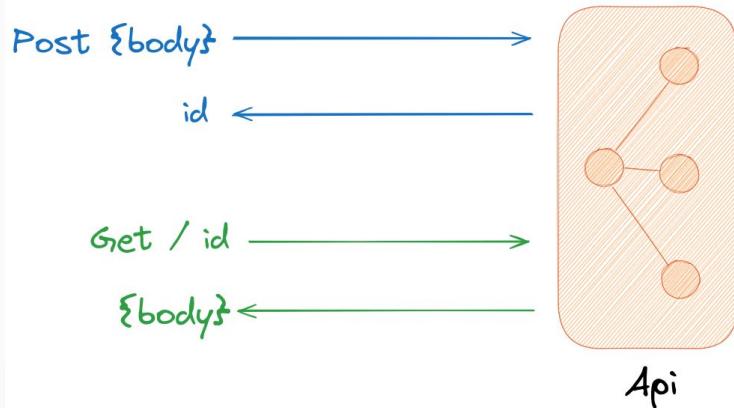
    const objectBody = JSON.parse(body);
    await schemaValidate(SCHEMAS_NAMES.IN, objectBody);

    const id = await dynamoService.createBodyResponse(objectBody);

    return {
      'statusCode': 200,
      'body': id
    };
  } catch (err) {
    console.log('End request with error!');
    console.log(err);
    return { 'statusCode': 500 };
  };
};
```



# AWS Lambda



POST https://9pnc5fizu8.execute-api.eu-central-1.amazonaws.com/ →

Status: 200 OK Size: 36 Bytes Time: 1.44 s

Headers 3 Auth Body 1 Tests Pre Run

Response Headers 11 Cookies Results Docs { }

1 11549f53-c041-4372-aef6-c90962a8ffac

JSON Content Format

```

1 {
2   "name": "Terraform",
3   "version": "version 1.4",
4   "ref": 103,
5   "desc": "Sample of Iac Tool"
6 }
```

Attributes	
Attribute name	Value
id - Partition key	11549f53-c041-4372-aef6-c90962a8ffac
body	<input type="button" value="Insert a field ▾"/>
desc	Sample of Iac Tool
name	Terraform
ref	103
version	version 1.4

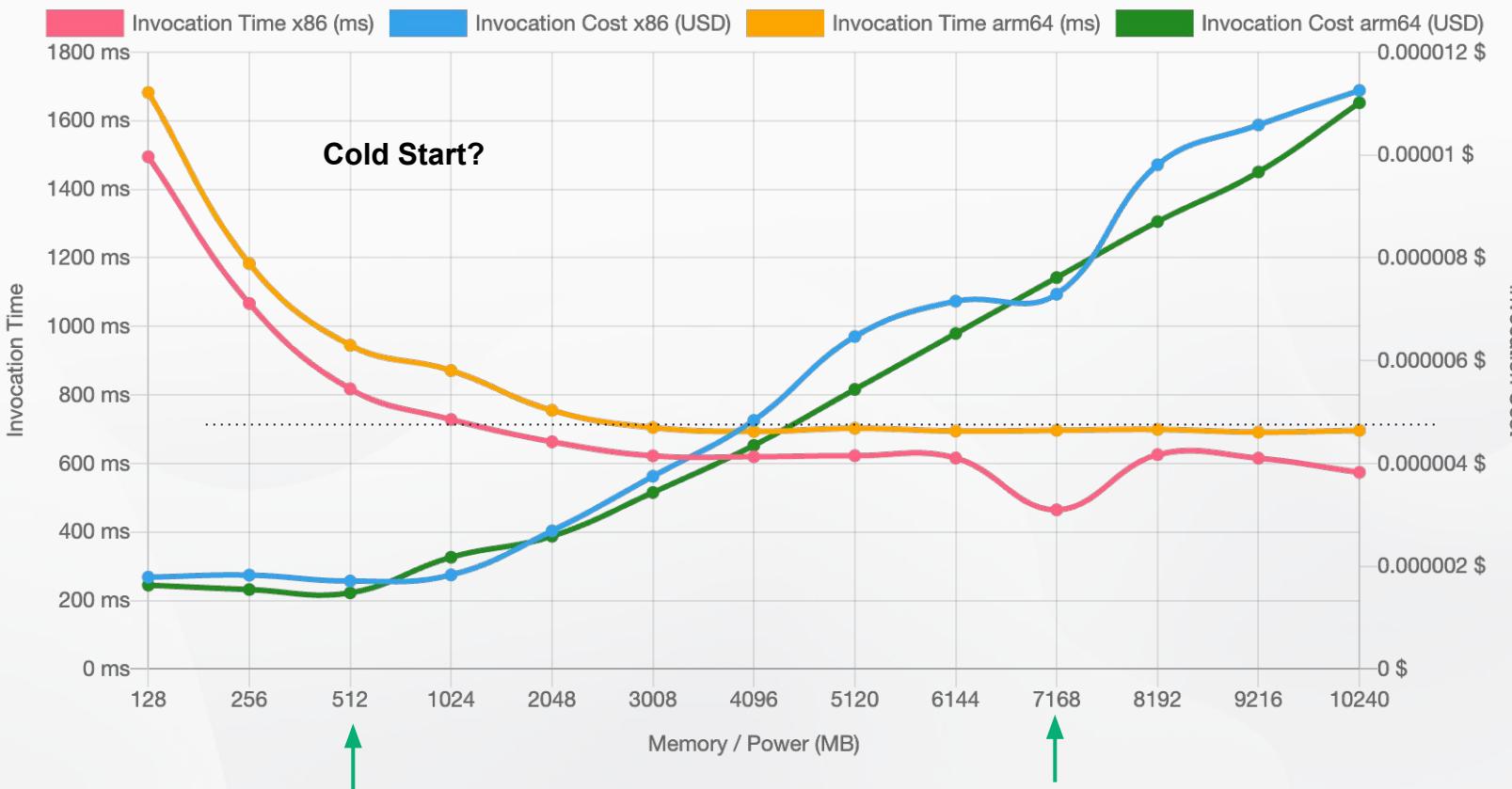
Add new attribute ▾

Status: 200 OK Size: 135 Bytes Time: 1.40 s

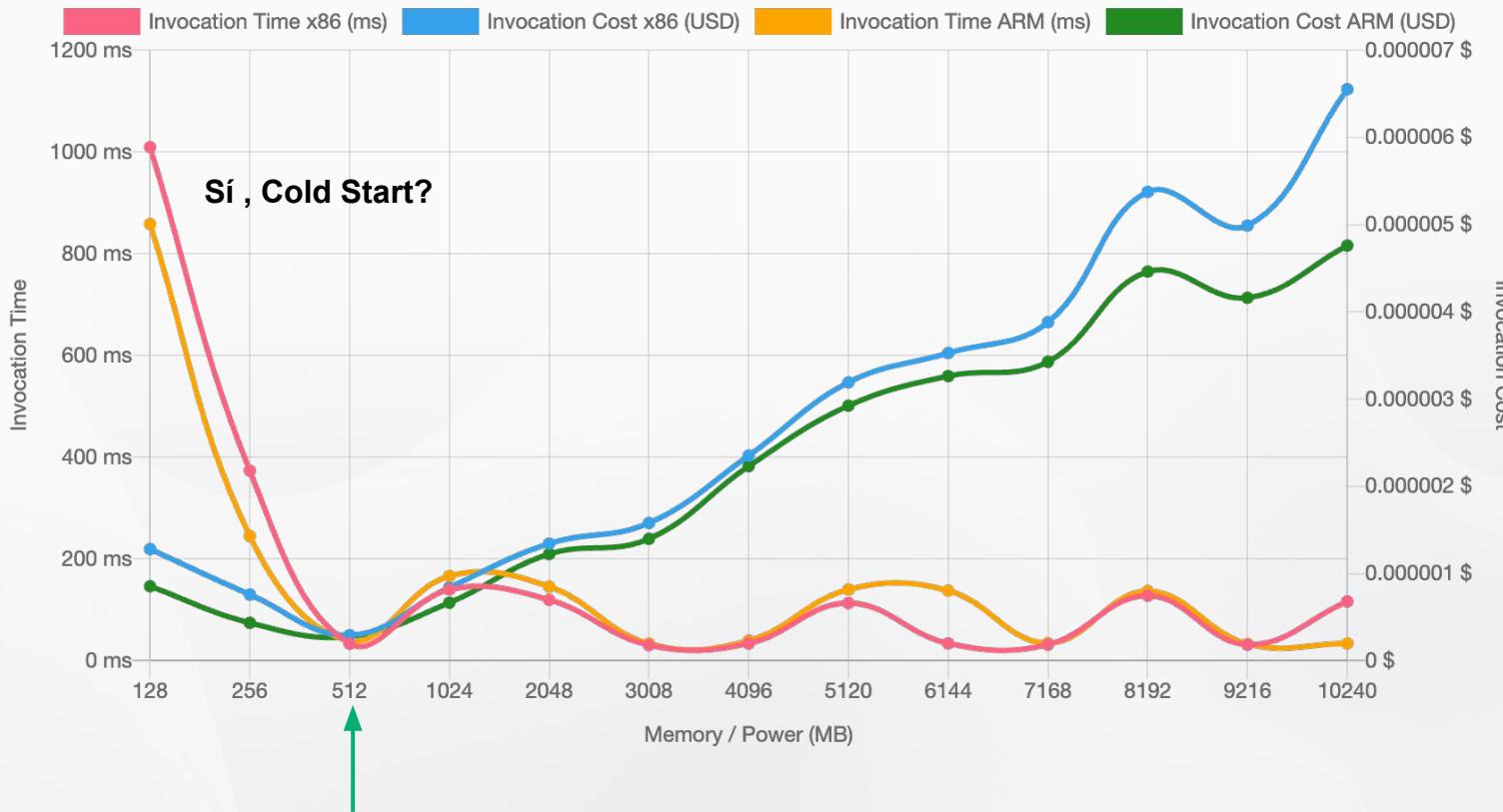
Response Headers 11 Cookies Results Docs { }

1 {
2 "id": "11549f53-c041-4372-aef6-c90962a8ffac",
3 "body": {
4 "name": "Terraform",
5 "ref": 103,
6 "version": "version 1.4",
7 "desc": "Sample of Iac Tool"
8 }
9 }

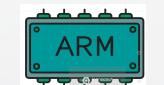
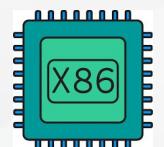
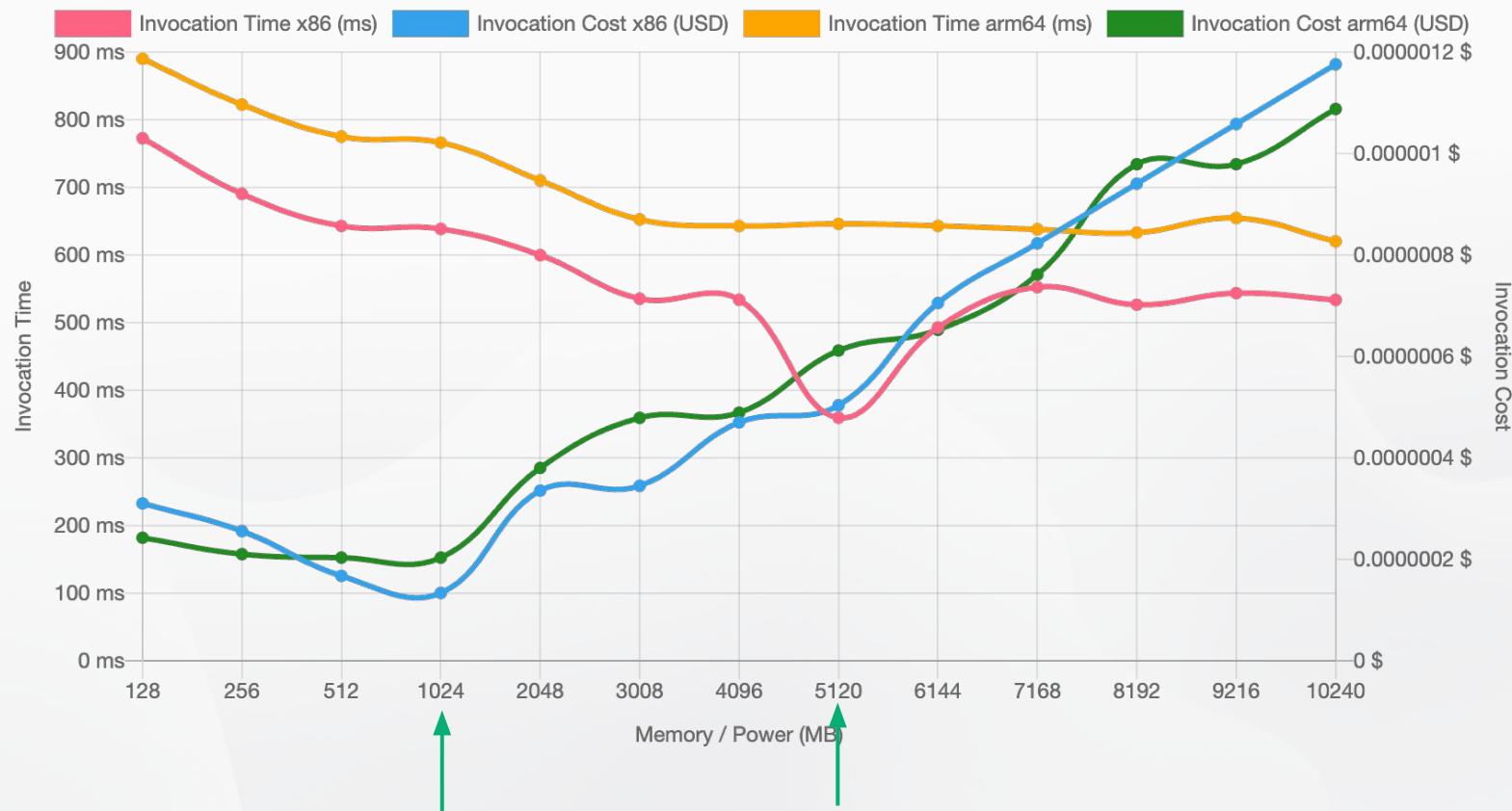
## mimic\_all\_memory\_range x86 vs arm64 (small body)



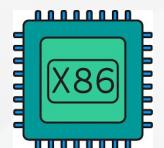
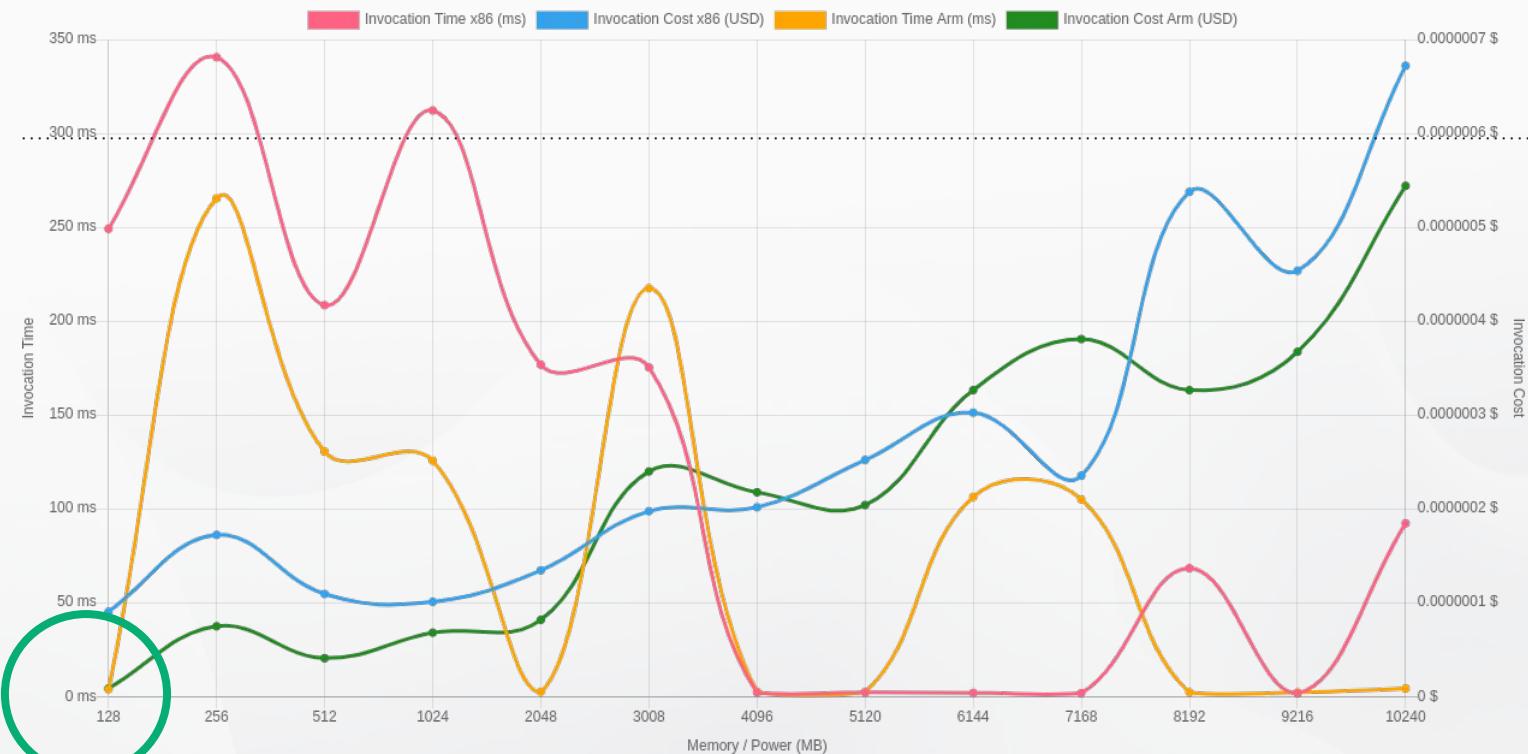
## mimic\_all\_memory\_range x86 vs arm64 (small body) - 10 provisioned



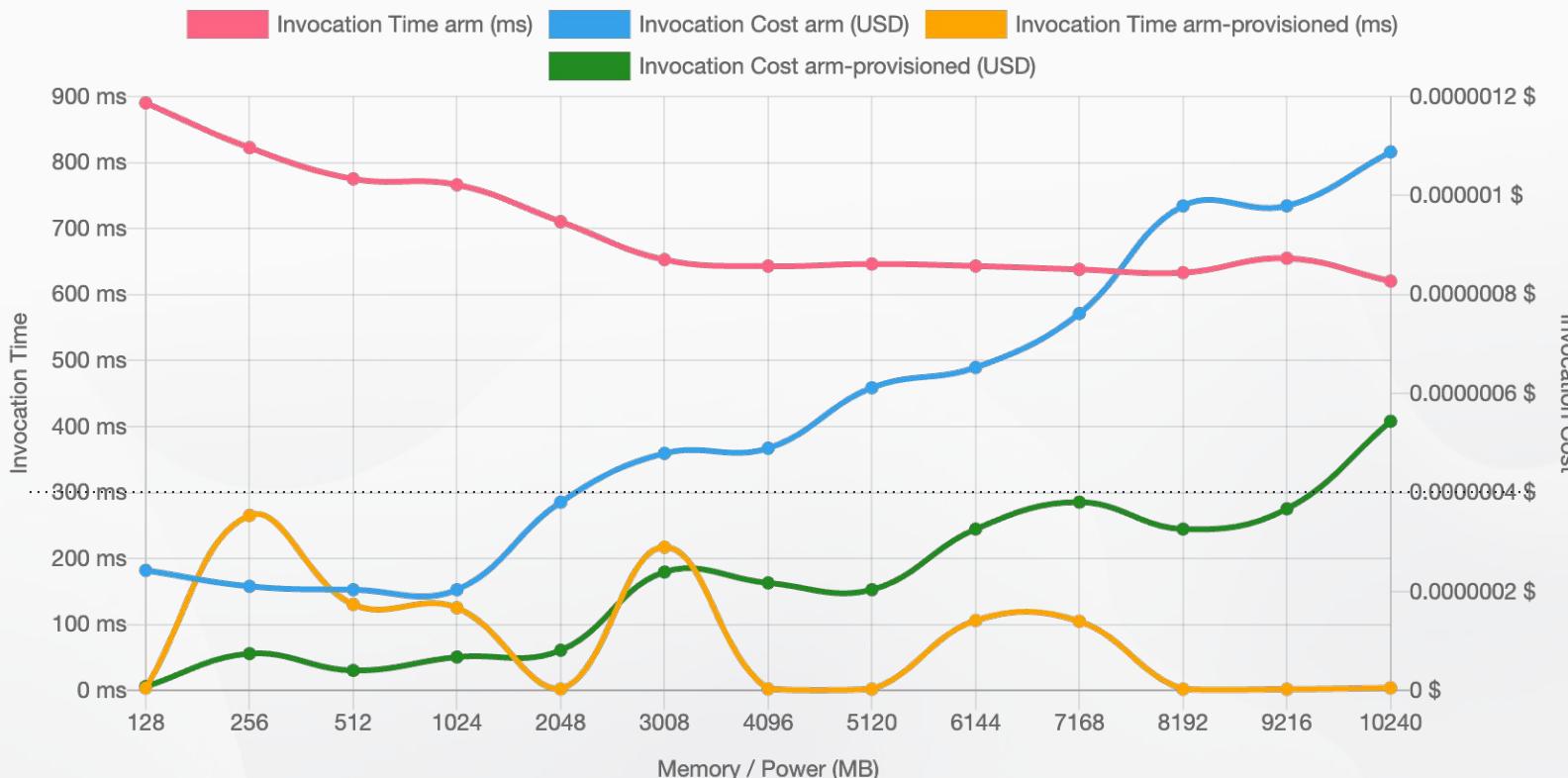
# mimic\_all\_memory\_range x86 vs arm64 (medium body)



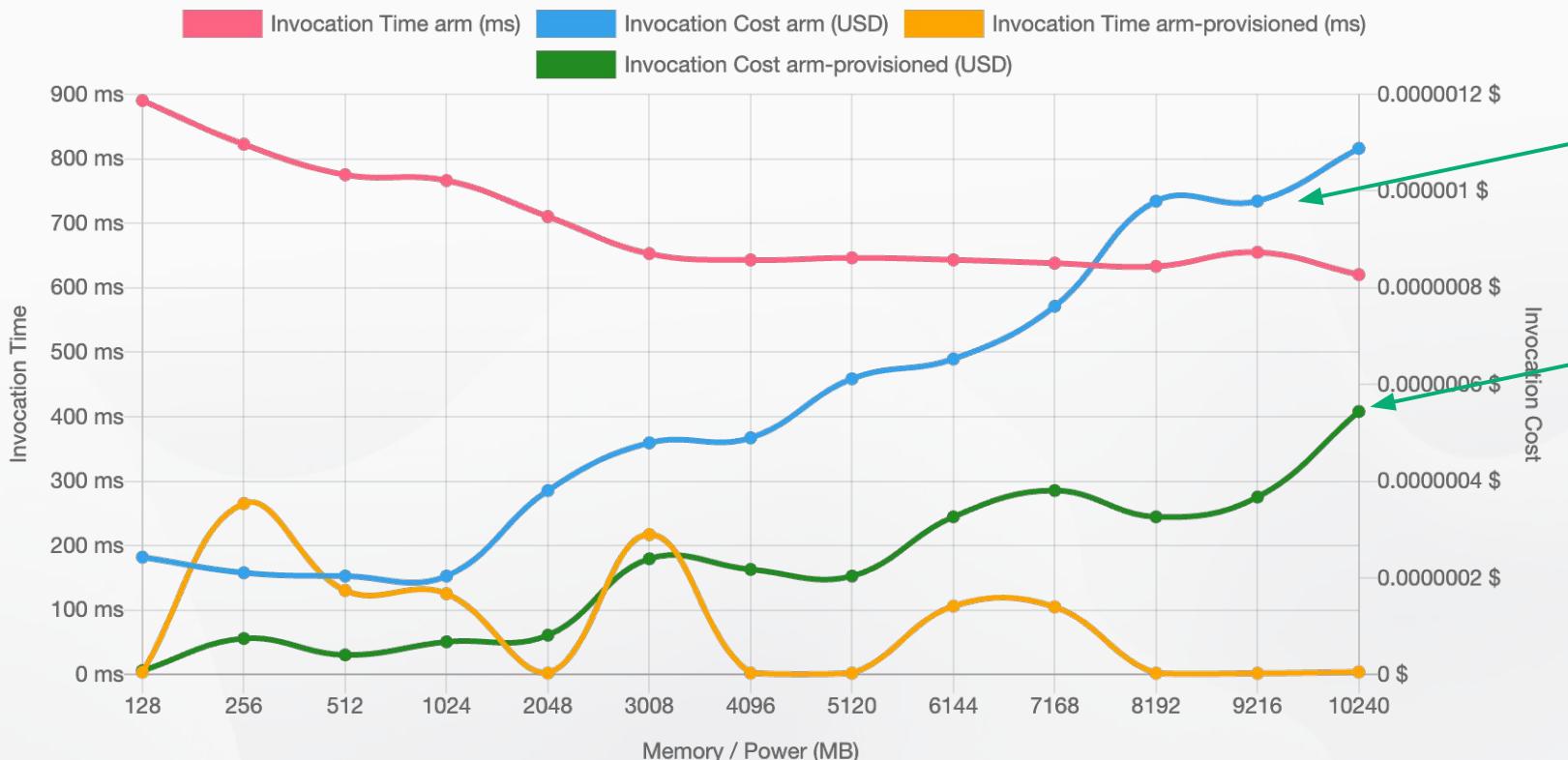
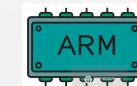
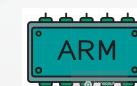
# mimic\_all\_memory\_range x86 vs arm64 (medium body)-10-R



# mimic\_all\_memory\_range arm64 vs arm64 (medium body)-10-R



# mimic\_all\_memory\_range arm64 vs arm64 (medium body)-10-R



# Métricas

**Logs Insights** [Info](#)

Select log groups, and then run a query or [choose a sample query](#).

Logs Insights QL    OpenSearch PPL - new    OpenSearch SQL - new

Select log groups by    Selection criteria

Log group name    Select up to 50 log groups

/aws/lambda/lambda\_listen\_pulumi-44d69e7 [X](#)    Clear all

```

1 filter @type = "REPORT"
2 | stats max(@memorySize / 1000 / 1000) as provisionedMemoryMB,
3 min(@maxMemoryUsed / 1000 / 1000) as smallestMemoryRequestMB,
4 avg(@maxMemoryUsed / 1000 / 1000) as avgMemoryUsedMB,
```

[Query generator](#)

[Run query](#)    Cancel    Save    History

Logs Insights QL query can run for maximum of 60 minutes.

Completed. Query executed for 1 log groups. [①](#)

#	provisionedMemoryMB	smallestMemoryRequestMB	avgMemoryUsedMB	maxMemoryUsedMB	overProvisionedMB
▼ 1	10240	94	98.1827	104	10136
	Field	Value			
	avgMemoryUsedMB	98.1827			
	maxMemoryUsedMB	104			
	overProvisionedMB	10136			
	provisionedMemoryMB	10240			
	smallestMemoryRequestMB	94			

**mimic-pulumi-table-24f06ce** [①](#)

[Overview](#)    [Indexes](#)    [Monitor](#)

[① Protect your DynamoDB table from accident](#)  
When you turn on point-in-time recovery (PIR) to 35 days. Additional charges apply. [Learn more](#)

**General information** [Info](#)

Partition key  
id (String)

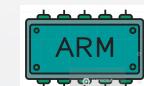
Alarms  
[② No active alarms](#)

► Additional info

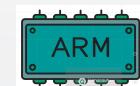
**Items summary**  
DynamoDB updates the following information automatically

Item count  
782

# mimic\_all\_memory\_range x86 vs arm64 (medium body)-10-R



# mimic\_all\_memory\_range x86 vs arm64 (medium body)-10-R

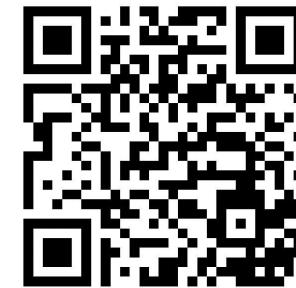


# Métricas

Highest AWS Region spend		Label	Min	Max	Sum	Average		
Highest AWS Region spend	<b>USD 7.48</b>	⌚ Invocations	1	634	9.05k	133		
Region name	EU (Frankfurt)	⌚ ConcurrentExecutions	1	53.9k	312k	4.59k		
Highest AWS Region spend	<b>USD 5.29</b>	⌚ Errors	0	198	952	14		
Region name	EU (Frankfurt)	⌚ Duration	3.8ms	55,889s	339,872s	4,998s		
AWS Free Tier usage limit	▼	Current usage	▼	Forecasted usage	▼	MTD actual usage %	▼	MTD forecasted usage %
400000.0 seconds are always free per month as part of AWS Free Usage Tier (Global-Lambda-GB-Second)	369,933 seconds	1,042,540 seconds	<div style="width: 92.48%; background-color: #007bff;"></div>	92.48%	<div style="width: 260.63%; background-color: #dc3545;"></div>	260.63%		
10.0 Alarms are always free per month as part of AWS Free Usage Tier (Global-CW:AlarmMonitorUsage)	7 Alarms	19 Alarms	<div style="width: 65.86%; background-color: #007bff;"></div>	65.86%	<div style="width: 185.61%; background-color: #dc3545;"></div>	185.61%		
18600.0 ReadCapacityUnit-Hrs are always free per month as part of AWS Free Usage Tier (EUC1-ReadCapacityUnit-Hrs)	6,370 ReadCapacityUnit-Hrs	17,952 ReadCapacityUnit-Hrs	<div style="width: 34.25%; background-color: #007bff;"></div>	34.25%	<div style="width: 96.52%; background-color: #dc3545;"></div>	96.52%		

## Conclusiones

- El uso de la herramienta **Lambda Power Tuning** es una buena estrategia para ajustar y entender el funcionamiento del Servicio de AWS Lambda(Ajuste Fino).
- Su implementación está gestionada por herramientas de infraestructura como código que facilitan la creación, gestión y posterior eliminación de la misma.
- Como cualquier análisis es importante definir los parámetros que dejaremos fijos y cuales ajustar en función de nuestras necesidades.
- La eficiencia de los procesadores merece un estudio detallado en función de lo que esperamos y de nuestras cargas de trabajo, aunque en condiciones generales siempre será más económico la tecnología ARM.



**Company | Hacker Dreams**

[www.linkedin.com](https://www.linkedin.com/company/hacker-dreams/)



**DevSecCon - Spain | Meetup**

[www.meetup.com](https://www.meetup.com/DevSecCon-Spain-Meetup/)

## Escríbenos

Un email es la vía más directa.

 [info@ackstorm.com](mailto:info@ackstorm.com)

## Llámanos

Nuestro horario de asistencia técnica es de lunes a viernes laborables de 8h a 20h, excepto en agosto, que es de lunes a viernes laborables de 8h a 15h.

 **(+34) 91 005 41 66**  
 **(+34) 93 393 63 88**

## Síguenos en redes sociales



## Hablemos

Ven a vernos a nuestras oficinas de Barcelona.

 **C/ de Pujades 362 Planta 1, 08019 Barcelona**



## ¡Gracias!

Cuéntanos la idea para tu negocio y te ayudaremos a hacerla posible en la nube.

[www.ackstorm.com](http://www.ackstorm.com)