

Going Serverless

With  aws



@AvinashDalvi_



@LearnWithAvinashDalvi

DevOps Services

About Me

Avinash Shashikant Dalvi

Senior Software Engineer @Eagleview

Full Stack Developer

AWS Community Builder

Supporting co-organiser of Docker Bangalore
and Collabnix meetup



@AvinashDalvi_



@LearnWithAvinashDalvi



www.internetkatta.com



avinashdalvi.com





What will learn today ?

- Exploring the necessity for Serverless
- What is Serverless ?
- Where Serverless can be used ?
- Discovering the Serverless offerings provided by AWS
- What Makes AWS Lambda Different?
- Examples and Limitation
- Walkthrough



@AvinashDalvi_



@LearnWithAvinashDalvi

Necessity for Serverless



- Paying for not using server
- Responsible for uptime and maintenance of the server
- Responsible for applying the security updates to the server
- Scale up and Scale down need to manage



@AvinashDalvi_



@LearnWithAvinashDalvi

Serverless - No server ???



Serverless is an execution approach where cloud provider like AWS, Azure or Google Cloud provide is responsible for executing piece of code by allocating runtime resources. And only charge for amount of resource used to run the code

Serverless implies a “NoOps” approach where developers don’t manage any backend servers or infrastructure



@AvinashDalvi_



@LearnWithAvinashDalvi

Ordering Pizzas



- What toppings will be - RAM, Storage etc
- What base will be - Which programming language
- Small/Medium/Large - code size
- Pizza delivery time whether immediate or schedule - execution time

Modular like pizza slices



@AvinashDalvi_



@LearnWithAvinashDalvi

Where Serverless can be used ?



- Unpredictable workloads or require quick scaling
- REST API
- Cron jobs and scheduled tasks
- Event-driven workflows
- Experimentation and prototyping (Building MVP)
- Data processing and analytics
- Many more...

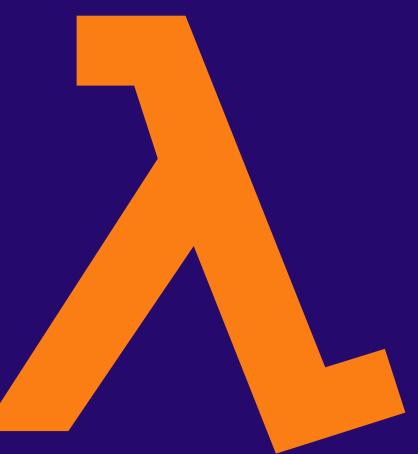


@AvinashDalvi_



@LearnWithAvinashDalvi

Serverless with

Serverless = 

Serverless computing gained prominence around 2014 with the release of AWS Lambda



@AvinashDalvi_



@LearnWithAvinashDalvi

What Makes AWS Lambda Different?

- Pay-as-you-go for great cost savings
- Completely event-driven
- Fully scalable
- Supports multiple languages and frameworks



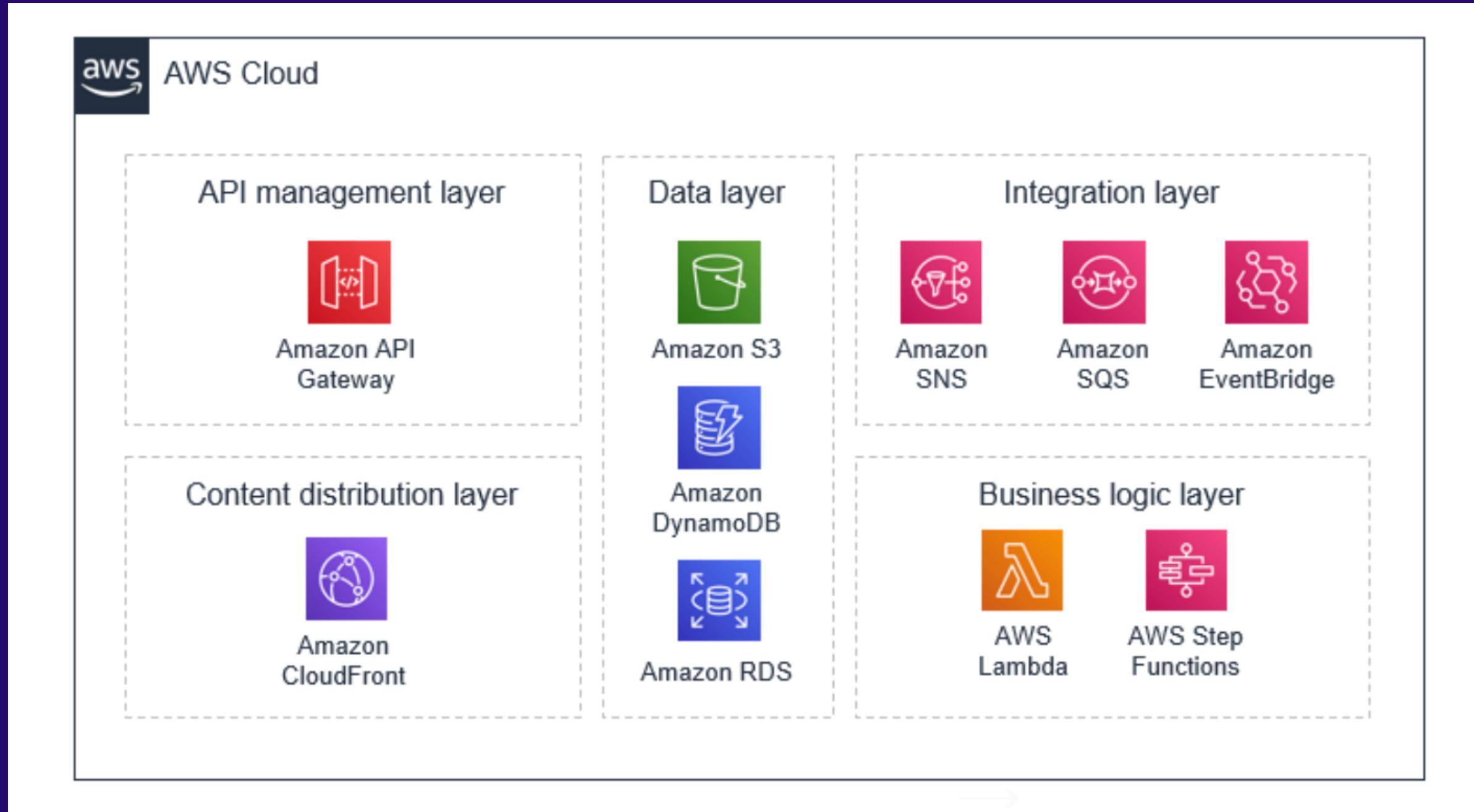
@AvinashDalvi_



@LearnWithAvinashDalvi

Serverless with AWS

Pic taken from AWS Documentation



@AvinashDalvi_



@LearnWithAvinashDalvi

Examples

- Bank statement analyser
- any CRUD APIs
- Schedule backup data
- Image processing
- Mass mailing or marketing mailer
- Any notification service - example renewal of your PUC by reading image
- For Building Serverless Chatbot
- Real-time Data Processing
- Processing uploaded S3 objects
- Log analysis on the fly



@AvinashDalvi_



@LearnWithAvinashDalvi

Limitation

AWS Lambda is not a silver bullet for every use case

The screenshot shows the AWS Lambda function configuration interface. It includes sections for Memory (128 MB), Ephemeral storage (512 MB), SnapStart (None), and Timeout (15 min 3 sec). A note at the bottom states: "The maximum timeout is 15 minutes."

Memory [Info](#)
Your function is allocated CPU proportional to the memory configured.
128 MB
Set memory to between 128 MB and 10240 MB

Ephemeral storage [Info](#)
You can configure up to 10 GB of ephemeral storage (/tmp) for your function. [View pricing](#)
512 MB
Set ephemeral storage (/tmp) to between 512 MB and 10240 MB.

SnapStart [Info](#)
Reduce startup time by having Lambda cache a snapshot of your function after the function has initialized. To evaluate whether your function code is resilient to snapshot operations, review the [SnapStart compatibility considerations](#).
None
Supported runtimes: Java 11, Java 17, Java 21.

Timeout
15 min 3 sec
⚠ The maximum timeout is 15 minutes.

But AWS keeps enhancing every year, which keeps Lambda's favourite option.



@AvinashDalvi_



@LearnWithAvinashDalvi

Lets walkthrough Lambda



@AvinashDalvi_



@LearnWithAvinashDalvi

References

- <https://docs.aws.amazon.com/lambda/latest/dg/gettingstarted-limits.html>
- <https://github.com/jarismar/b3c-invoice-reader-lambda>
- <https://github.com/teohrt/cruddyAPI>



@AvinashDalvi_



@LearnWithAvinashDalvi

- Learn Together & Grow Together

Thank You.



@AvinashDalvi_



@LearnWithAvinashDalvi