

Build Serverless Python Applications using AWS Chalice

Harsh Bardhan Mishra

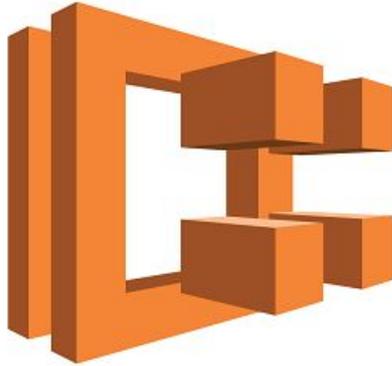
Agenda

1. What is Serverless?
2. Background to AWS Lambda
3. Setting up a Project on Chalice
4. Developing a REST API Service (Hands-on)
5. Deploying the Application on AWS (Hands-on)
6. Testing the API (Hands-on)
7. Q&A

How do you deploy your APIs?



Virtual Machines



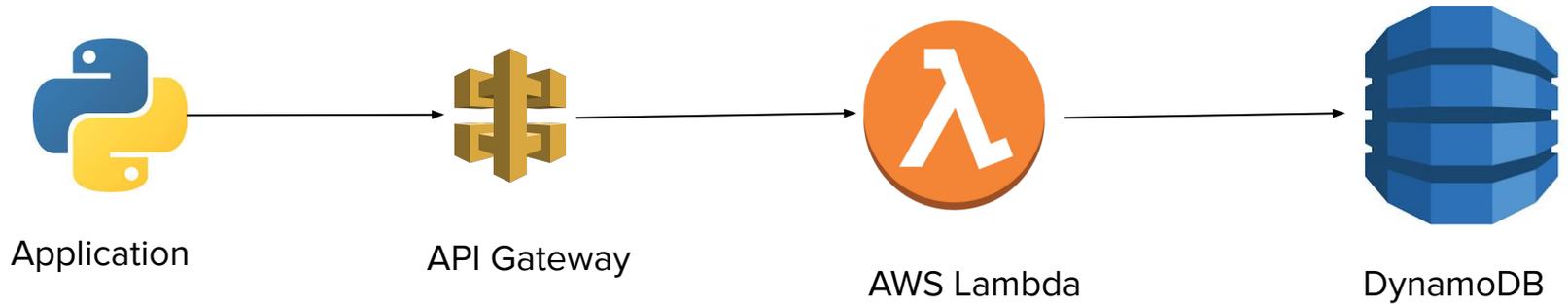
Containers



Serverless

What is Serverless?

Serverless is a method of providing machine resources on an as-used basis.



Why Serverless?

- No server management.
- Flexible scaling.
- High availability.
- No idle capacity.
- Pay as per execution.

AWS Lambda to the rescue!



AWS Lambda

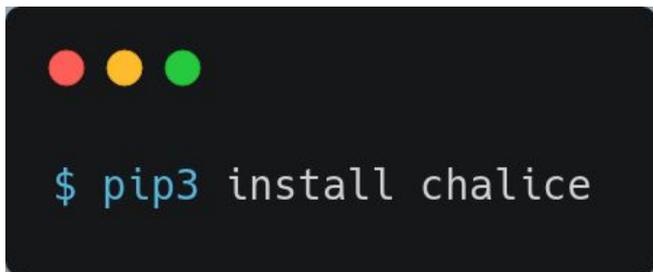
General Concepts

- Write functions as the code.
 - Lambda functions can be written in various languages (Python, Go, JavaScript and more)
 - Deploy your code as a ZIP file or in a container image.
- Manages complexity with continuous scaling.
 - Scale your functions without managing or provisioning any servers.
 - Scaling happens as per the workload and compute power required.
- Automated resource allocation
 - Each trigger is processed individually and runs in parallel.
 - Size of workload and computing resources is automatically chosen and allocated.

Welcome AWS Chalice!



Yet Another Python Micro Web-Framework!



```
$ pip3 install chalice
```

Standard setup using `pip`



```
from chalice import Chalice

app = Chalice(app_name="helloworld")

@app.route("/")
def index():
    return {"hello": "world"}
```

Simple Hello World API!

Why AWS Chalice?

- Serverless Python development and deployment in a single-go.
- Unified CLI interface for setting, testing and deploying.
- Zero external configurations and setup required, apart from pip, Python and AWS credentials.
- Tightly integrates with other AWS services in an extended ecosystem.

Practical Hands-On

- Setting up an AWS Chalice project (*pip3 install chalice*)
- Testing it locally (*chalice local*)
- Deploying the application (*chalice deploy*)
- Extending the API

Thank You!

Any questions?