Python In The Serverless Era

Benny Bauer
Software Architect, Autodesk

benny.bauer@autodesk.com

@benikbauer

Agenda

- Hello
- Cloud Evolution
- Serverless Architecture
- Ecosystem

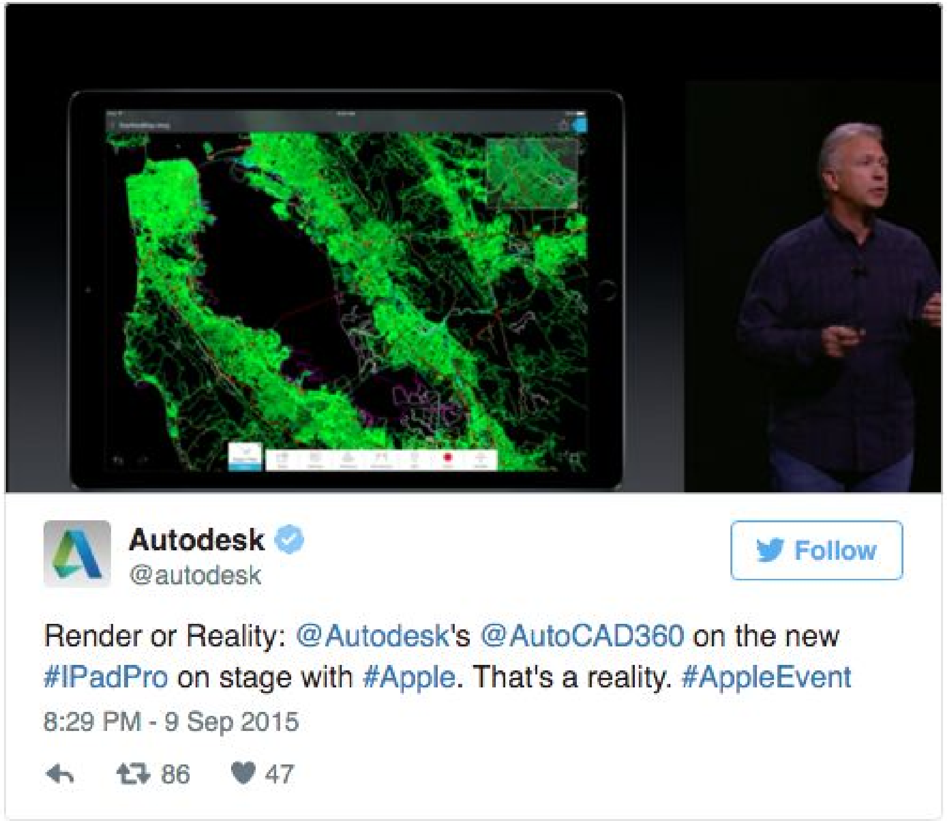
Hello

Hello

TLV

AUTODESK®

AutoCAD 360

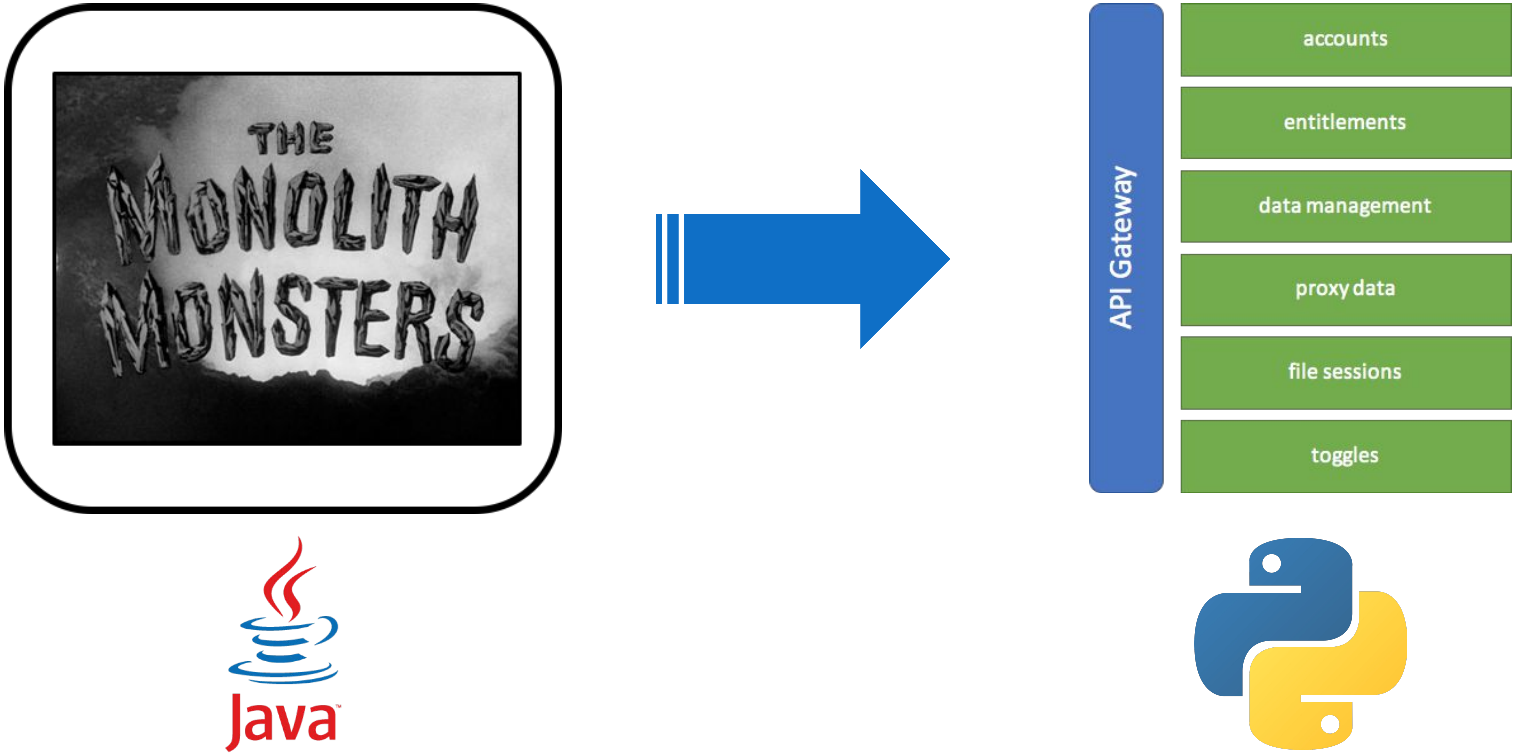


Autodesk
@autodesk

Render or Reality: @Autodesk's @AutoCAD360 on the new #IPadPro on stage with #Apple. That's a reality. #AppleEvent

8:29 PM - 9 Sep 2015

AutoCAD 360 Backend



API Gateway

accounts

entitlements

data management

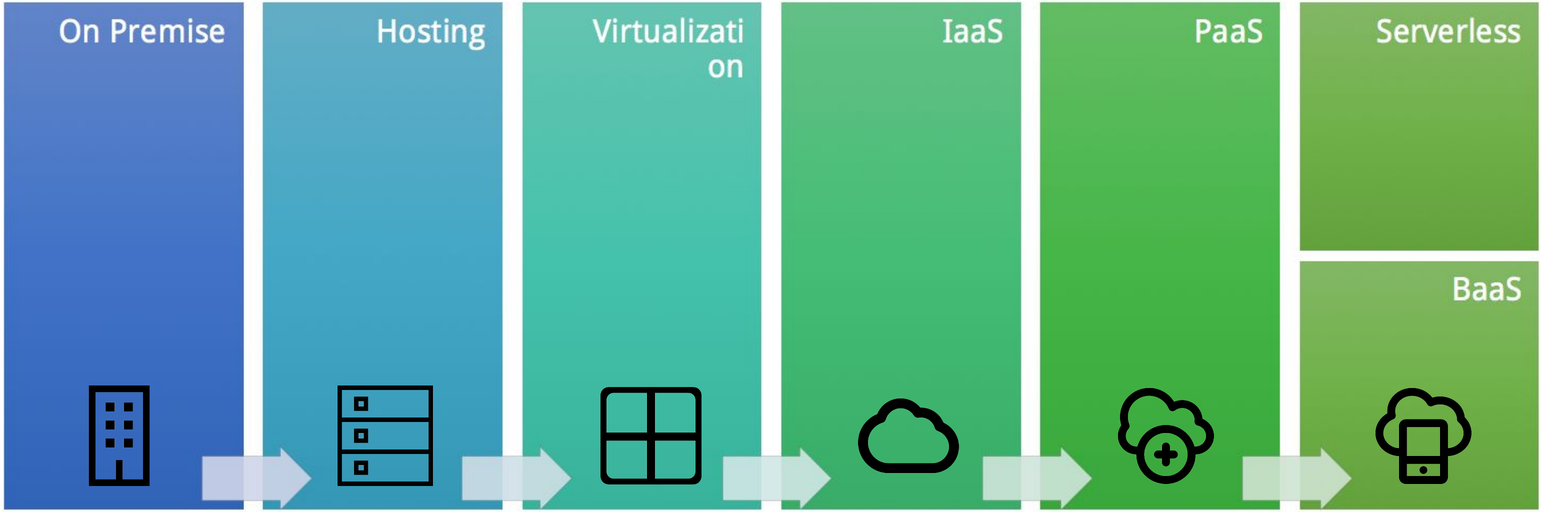
proxy data

file sessions

toggles

Cloud Evolution

It’s Evolution Baby!

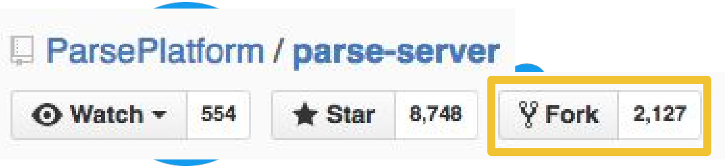


Icons made by freepik from www.flaticon.com

Backend as a Service

- Hosting
- Data access
- Authentication
- Notifications
- Monitoring
- Analytics

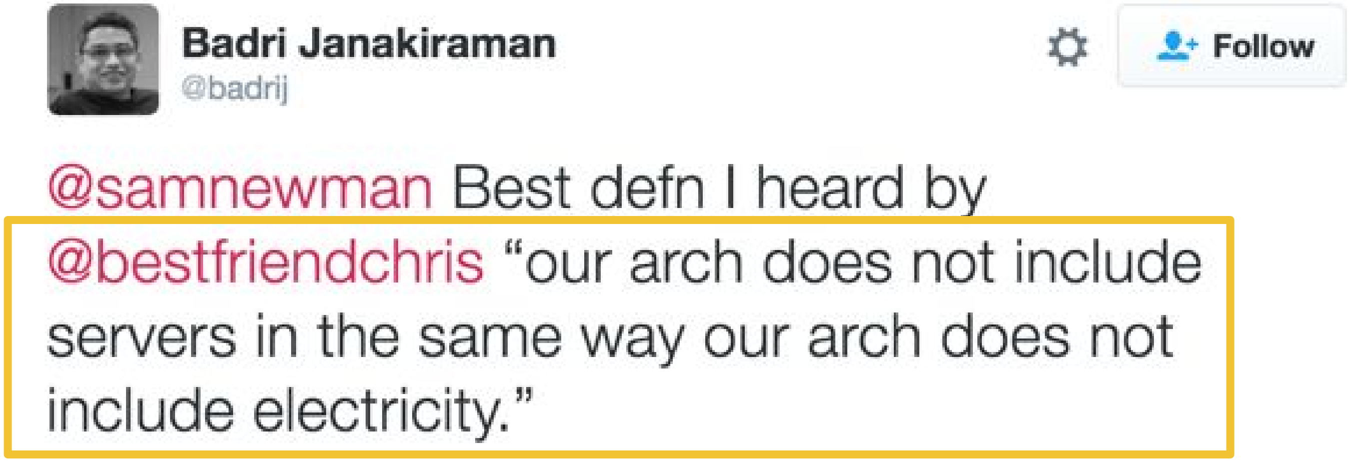
Firebase



ParsePlatform / parse-server

Serverless Architecture

Introducing Serverless



Badri Janakiraman
@badrij

@samnewman Best defn I heard by @bestfriendchris "our arch does not include servers in the same way our arch does not include electricity."

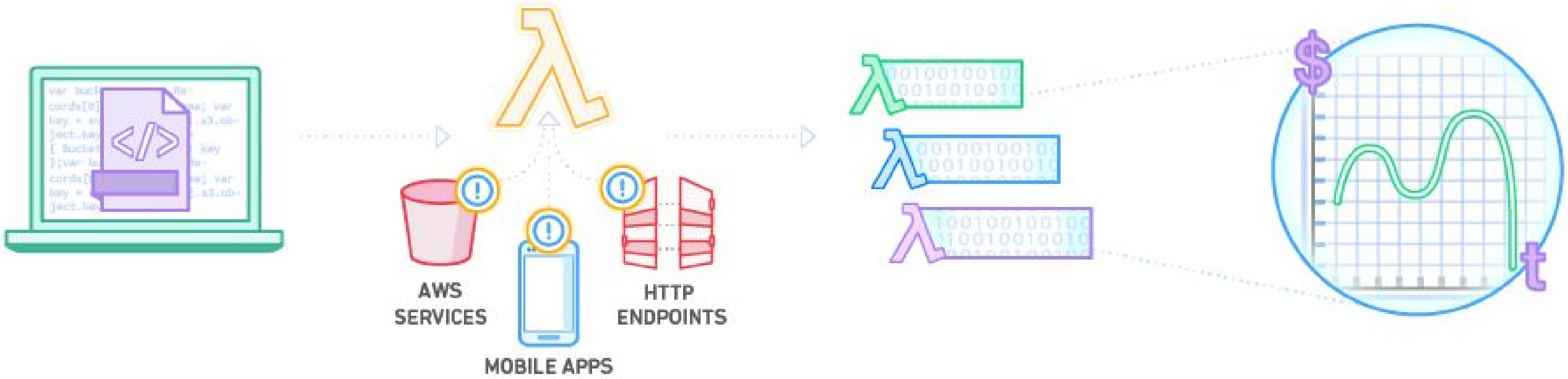
Serverless Architecture

- **Fully managed** compute
  - Provisioning, patching
  - Scalability
  - Monitoring
  - Logging
  - No ops
- **Just deploy** your code
- Pay **only** for actual usage == Full utilisation!

CLOUD FUNCTIONS
Create small, single-purpose functions
that respond to events in the cloud

Parse

AWS Lambda - How it works



Upload your code to AWS Lambda

Set up your code to trigger from other AWS services, HTTP endpoints, or in-app activity

Lambda runs your code only when triggered, using only the compute resources needed

Pay just for the compute time you use

origin: https://aws.amazon.com/lambda/

Use cases

- **REST API**
  - Stateless services
  - Suitable for Slack apps (though not bots)
- **Events**
  - File processing (S3 event) & Data ingestion (Kinesis event)
  - Incidents handling (CloudWatch event)
  - IoT
- **Scheduled tasks**
  - Monitoring, sanity tests, load testing
  - Periodical jobs

AWS Lambda Characteristics

- **Stateless**
- **Autoscaled** according to demand (events or requests)

AWS Lambda Limitations

- Need to **keep it warm**
- Convenience can lead to **vendor lock-in**
- Languages: **Python 2.7**, Node.js, Java 8
- **Execution time** is limited to 5 min
- **Concurrent execution** is limited to 100*
- Various **payload** and **disk size** limits
- Not in all AWS **regions**
- No SSH (which is good!)

The Required Mindset

Code should be:

- Small
- Short-lived
- Stateless

Ecosystem

AWS Lambda & API Gateway are nice, but…

- **Configuration** headache
  - IAM roles
  - API Gateway
- **Deployment** headache
  - Packaging
  - Uploading
  - Rollback

Serverless Framework

Serverless
7,979 ★

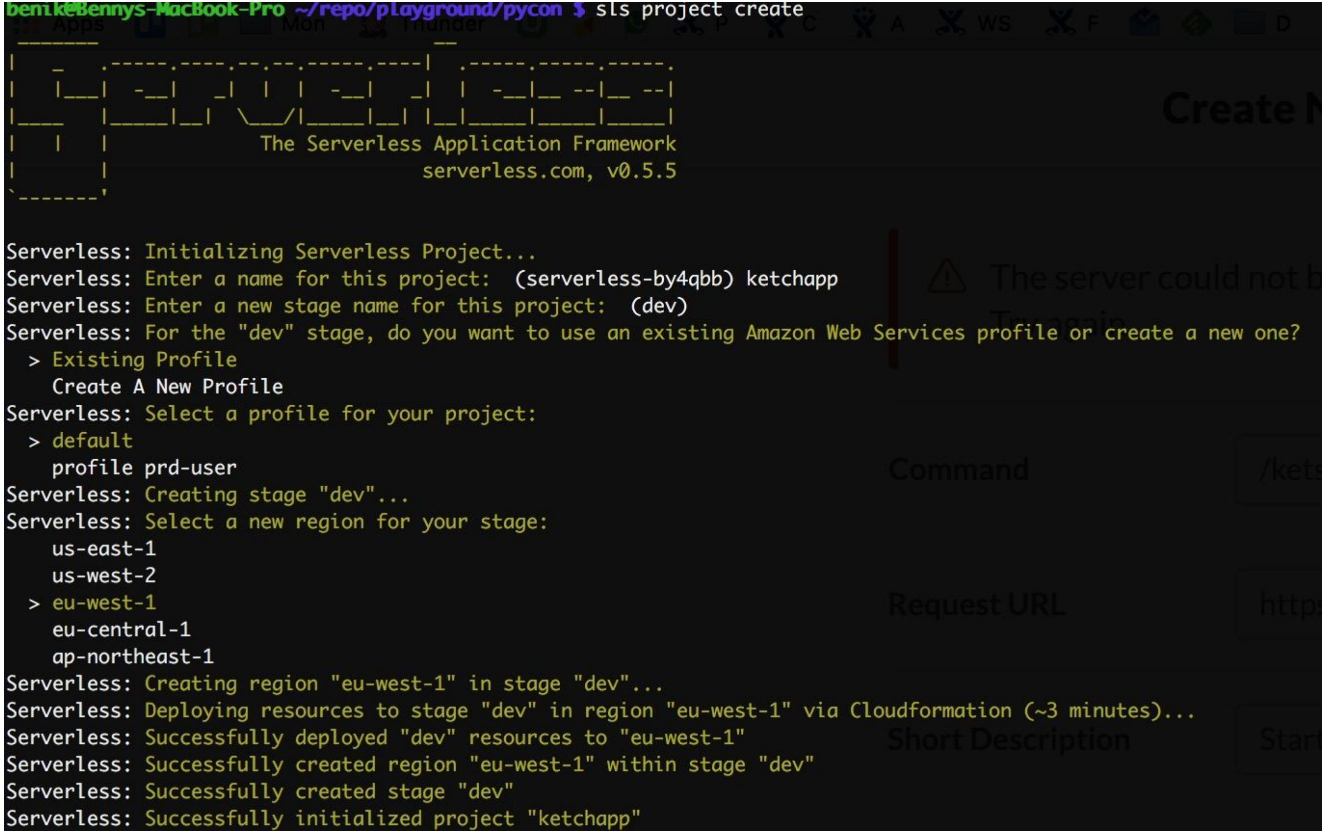
Application framework for building applications exclusively on AWS Lambda and API Gateway

origin: https://github.com/serverless/serverless

Serverless Framework

- **Wraps** AWS Lambda, API Gateway and IAM
- Manages **deployments**
- **CLI**
- **Modular**, functions can be easily shared
- **Extensible** via plugins
- Thriving **community**
- Node.js - posterboy, **Python** - foster child

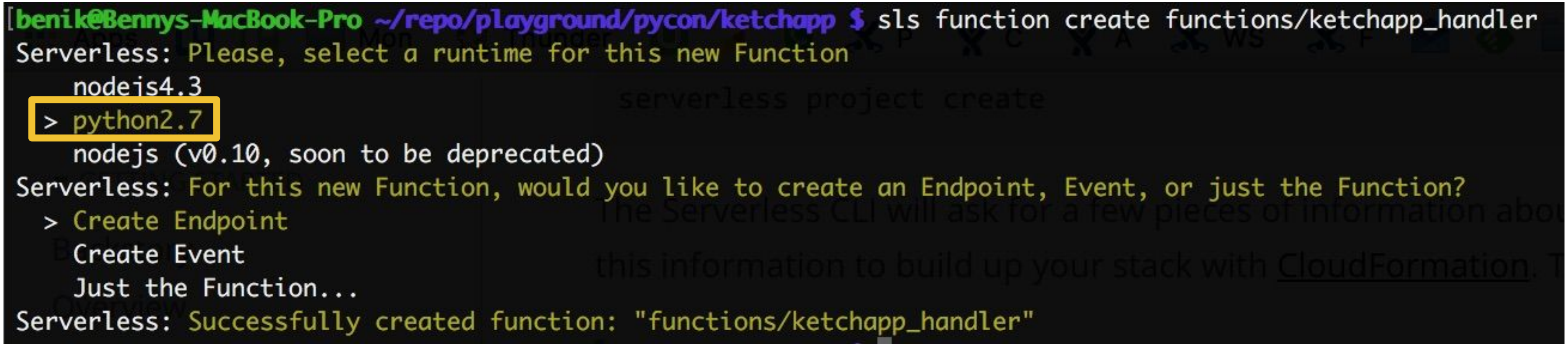
$ sls project create



```
benik@Bennys-MacBook-Pro ~/repo/playground/pycon $ sls project create
 _______                             __
|   _   .-----.----.--.--.-----.----|  .-----.-----.-----.
|   |___|  -__|   _|  |  |  -__|   _|  |  -__|__ --|__ --|
|____   |_____|__|  \___/|_____|__| |__|_____|_____|_____|
|   |   |             The Serverless Application Framework
|       |                           serverless.com, v0.5.5
`-------'

Serverless: Initializing Serverless Project...
Serverless: Enter a name for this project:  (serverless-by4qbb) ketchapp
Serverless: Enter a new stage name for this project:  (dev)
Serverless: For the "dev" stage, do you want to use an existing Amazon Web Services profile or create a new one?
  > Existing Profile
    Create A New Profile
Serverless: Select a profile for your project:
  > default
    profile prd-user
Serverless: Creating stage "dev"...
Serverless: Select a new region for your stage:
    us-east-1
    us-west-2
  > eu-west-1
    eu-central-1
    ap-northeast-1
Serverless: Creating region "eu-west-1" in stage "dev"...
Serverless: Deploying resources to stage "dev" in region "eu-west-1" via Cloudformation (~3 minutes)...
Serverless: Successfully deployed "dev" resources to "eu-west-1"
Serverless: Successfully created region "eu-west-1" within stage "dev"
Serverless: Successfully created stage "dev"
Serverless: Successfully initialized project "ketchapp"
```

$ sls function create

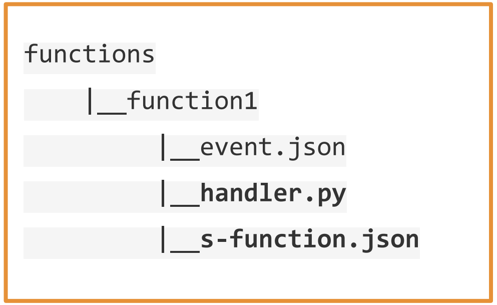


```
[benik@Bennys-MacBook-Pro ~/repo/playground/pycon/ketchapp $ sls function create functions/ketchapp_handler
Serverless: Please, select a runtime for this new Function
    nodejs4.3
  > python2.7
    nodejs (v0.10, soon to be deprecated)
Serverless: For this new Function, would you like to create an Endpoint, Event, or just the Function?
  > Create Endpoint
    Create Event
    Just the Function...
Serverless: Successfully created function: "functions/ketchapp_handler"
```

# implement

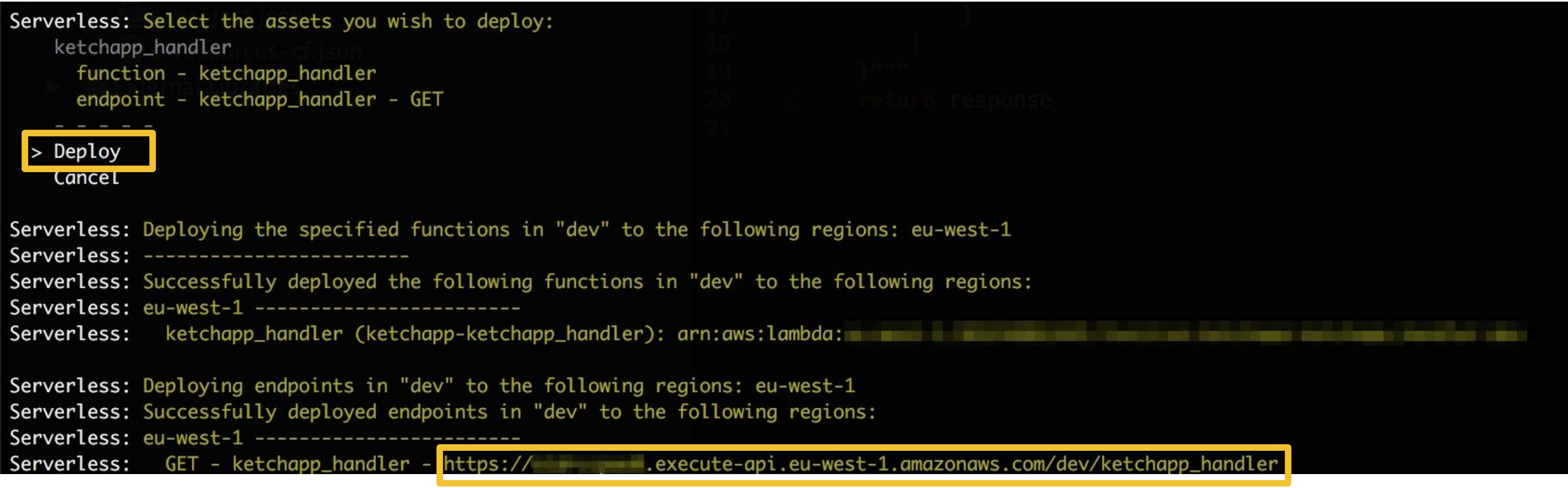
- handler.py
  - Entry point to your implementation

- s-function.json
  - Endpoints configuration
  - Events configuration
  - Env vars definition
  - Lambda configuration (runtime, timeout, size, etc.)



```
functions
    |__function1
         |__event.json
         |__handler.py
         |__s-function.json
```

$ sls dash deploy



```
Serverless: Select the assets you wish to deploy:
    ketchapp_handler
      function - ketchapp_handler
      endpoint - ketchapp_handler - GET
    - - - - - -
  > Deploy
    Cancel

Serverless: Deploying the specified functions in "dev" to the following regions: eu-west-1
Serverless: ------------------------
Serverless: Successfully deployed the following functions in "dev" to the following regions:
Serverless: eu-west-1 ------------------------
Serverless:   ketchapp_handler (ketchapp-ketchapp_handler): arn:aws:lambda:[redacted]

Serverless: Deploying endpoints in "dev" to the following regions: eu-west-1
Serverless: Successfully deployed endpoints in "dev" to the following regions:
Serverless: eu-west-1 ------------------------
Serverless:   GET - ketchapp_handler - https://[redacted].execute-api.eu-west-1.amazonaws.com/dev/ketchapp_handler
```

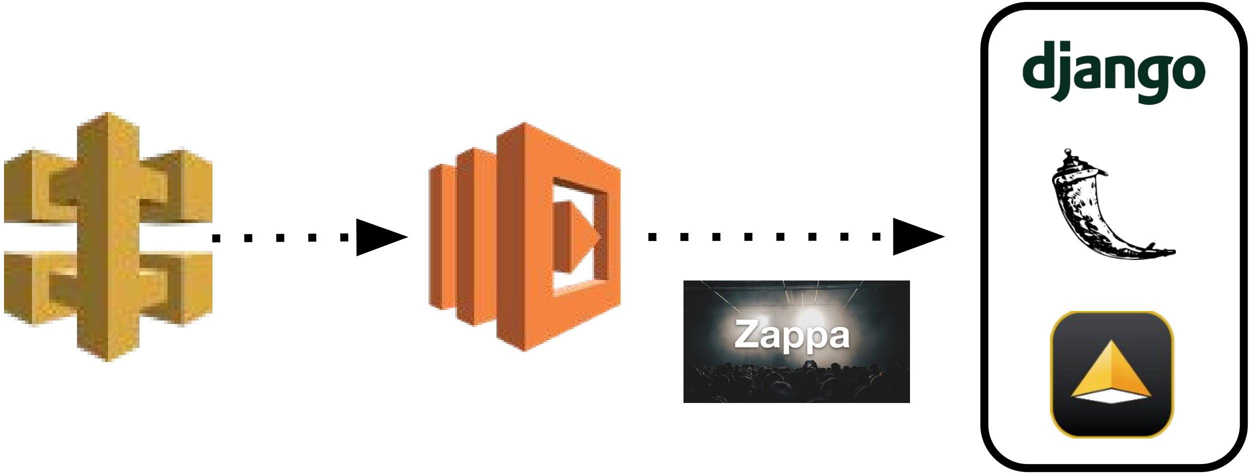
Zappa

Rich Jones
Miserlou

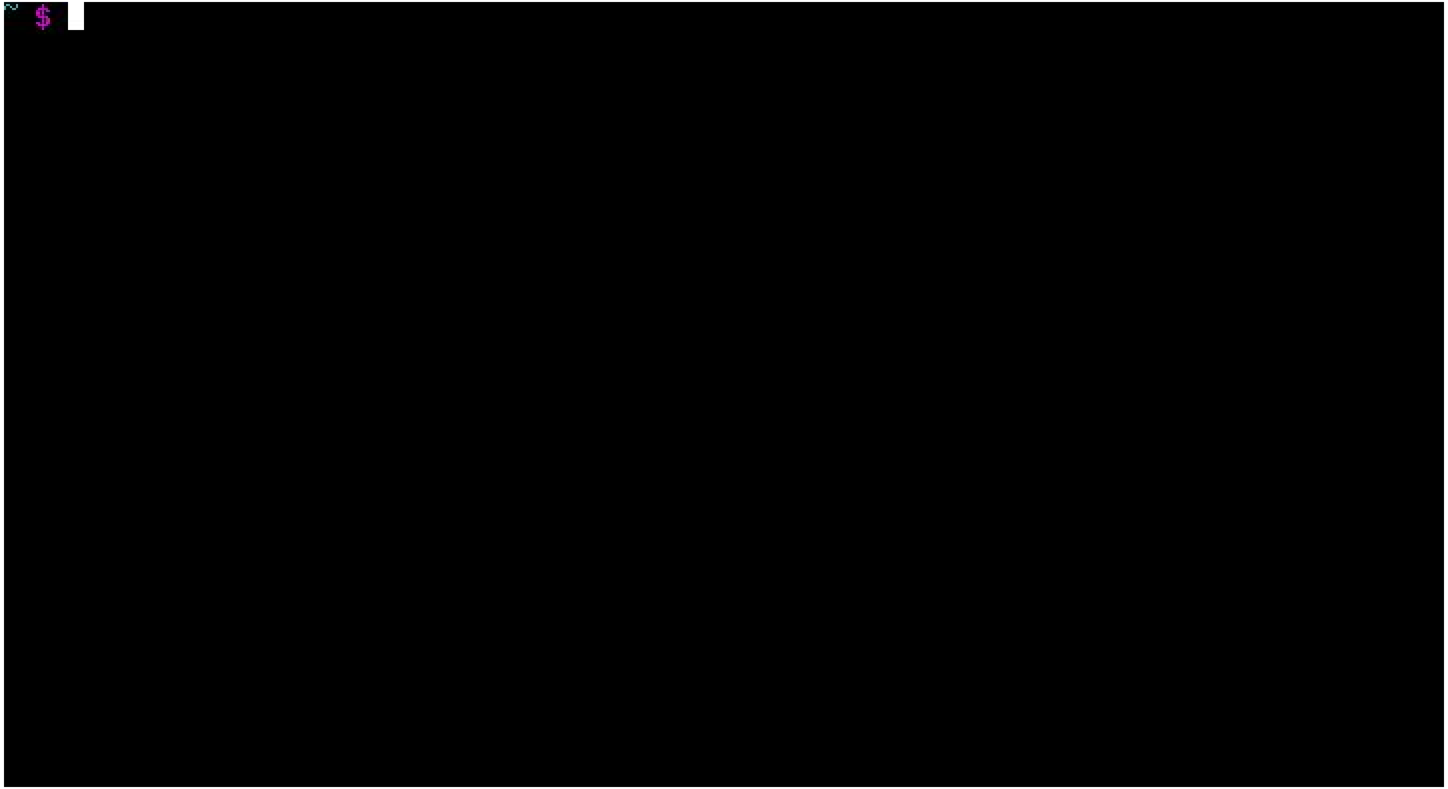
Python WSGI applications deployment on AWS Lambda + API Gateway.

origin: https://github.com/Miserlou/Zappa

Zappa Architecture

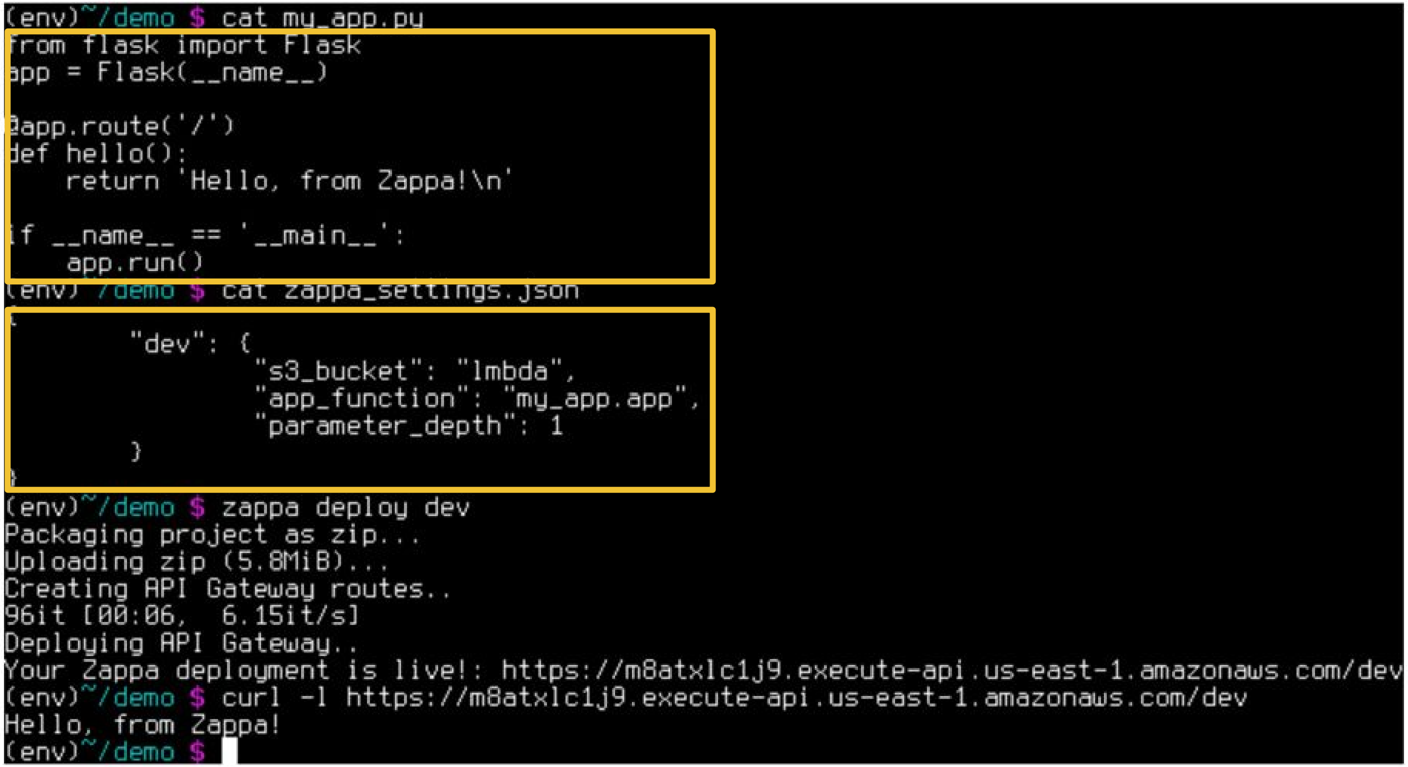


Zappa example



```
~ $
```

Zappa example



```
(env)~/demo $ cat my_app.py
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello():
    return 'Hello, from Zappa!\n'

if __name__ == '__main__':
    app.run()
(env)~/demo $ cat zappa_settings.json
{
        "dev": {
                "s3_bucket": "lmbda",
                "app_function": "my_app.app",
                "parameter_depth": 1
        }
}
(env)~/demo $ zappa deploy dev
Packaging project as zip...
Uploading zip (5.8MiB)...
Creating API Gateway routes..
96it [00:06,  6.15it/s]
Deploying API Gateway..
Your Zappa deployment is live!: https://m8atxlc1j9.execute-api.us-east-1.amazonaws.com/dev
(env)~/demo $ curl -l https://m8atxlc1j9.execute-api.us-east-1.amazonaws.com/dev
Hello, from Zappa!
(env)~/demo $
```

Zappa - How it works

```
$ zappa deploy <env>
```

1. Zips code and dependencies
2. Create AWS Lambda and deploys the zip
3. Creates endpoint on API Gateway and ties to AWS Lambda

Takeaways

Serverless architecture is the next generation of cloud evolution

Takeaways

The Serverless ecosystem is on the rise,
many interesting **opportunities** for
the **Python community** to contribute!

References

- awesome-serverless awesome
- https://github.com/Miserlou/Zappa
- Serverless Framework talk (aka JAWS) on AWS re:invent https://youtu.be/D_U6luQ6I90

Questions?

Thank You!