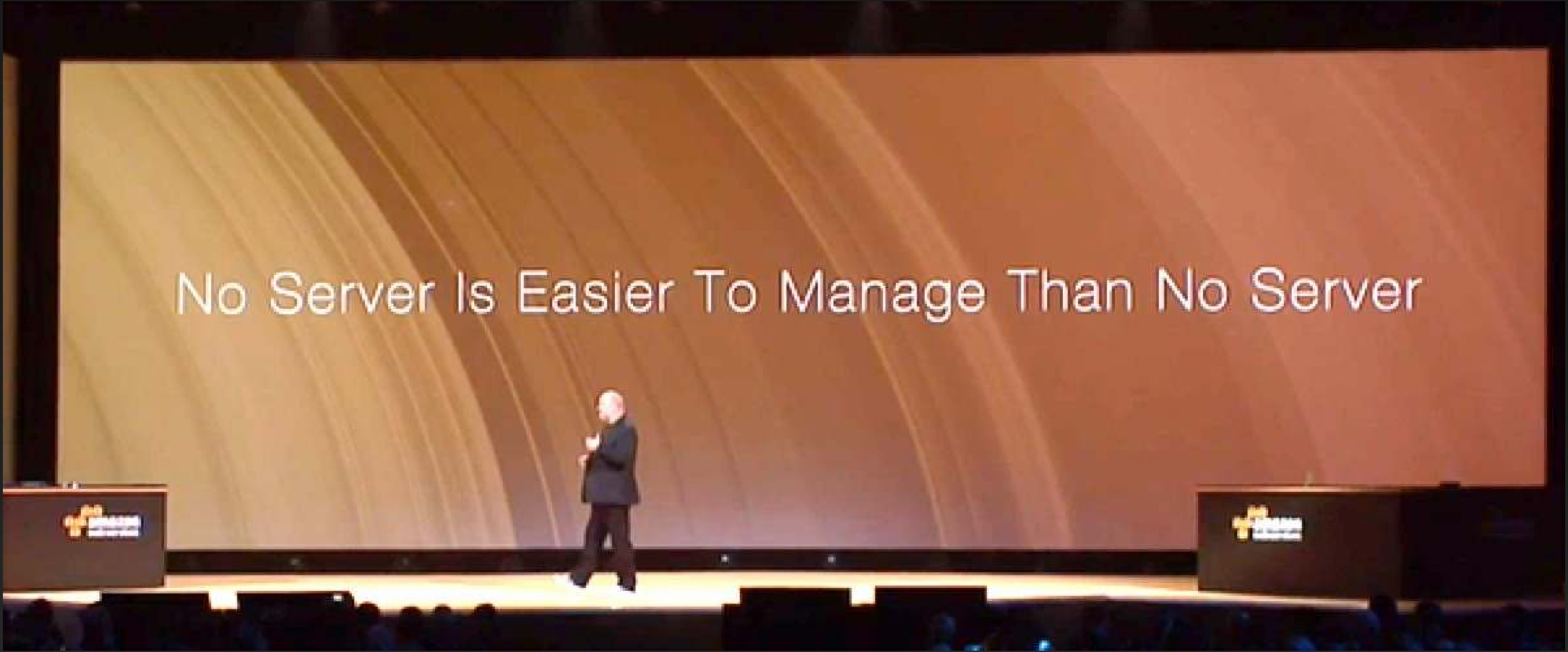
Building serverless APIs

Julien Simon, Principal Technical Evangelist, AWS
julsimon@amazon.fr
@julsimon



Werner Vogels, CTO, Amazon.com
AWS re:Invent 2015

Serverless architecture = Managed services + AWS Lambda

Amazon API Gateway

Amazon Kinesis Streams

Amazon DynamoDB

Amazon S3

Reference architectures: http://www.allthingsdistributed.com/2016/06/aws-lambda-serverless-reference-architectures.html

AWS Lambda

- Announced at re:Invent 2014
- Deploy pure functions in Java, Python, Node.js and C#
- Just code, without the infrastructure drama
- Built-in scalability and high availability
- Integrated with many AWS services
- Pay as you go
    - Combination of execution time (100ms slots) & memory used
    - Starts at $0.000000208 per 100ms
    - Free tier available: first 1 million requests per month are free

What can you do with AWS Lambda?

- Grow ‘connective tissue’ in your AWS infrastructure
  - Example: http://www.slideshare.net/JulienSIMON5/building-a-serverless-pipeline

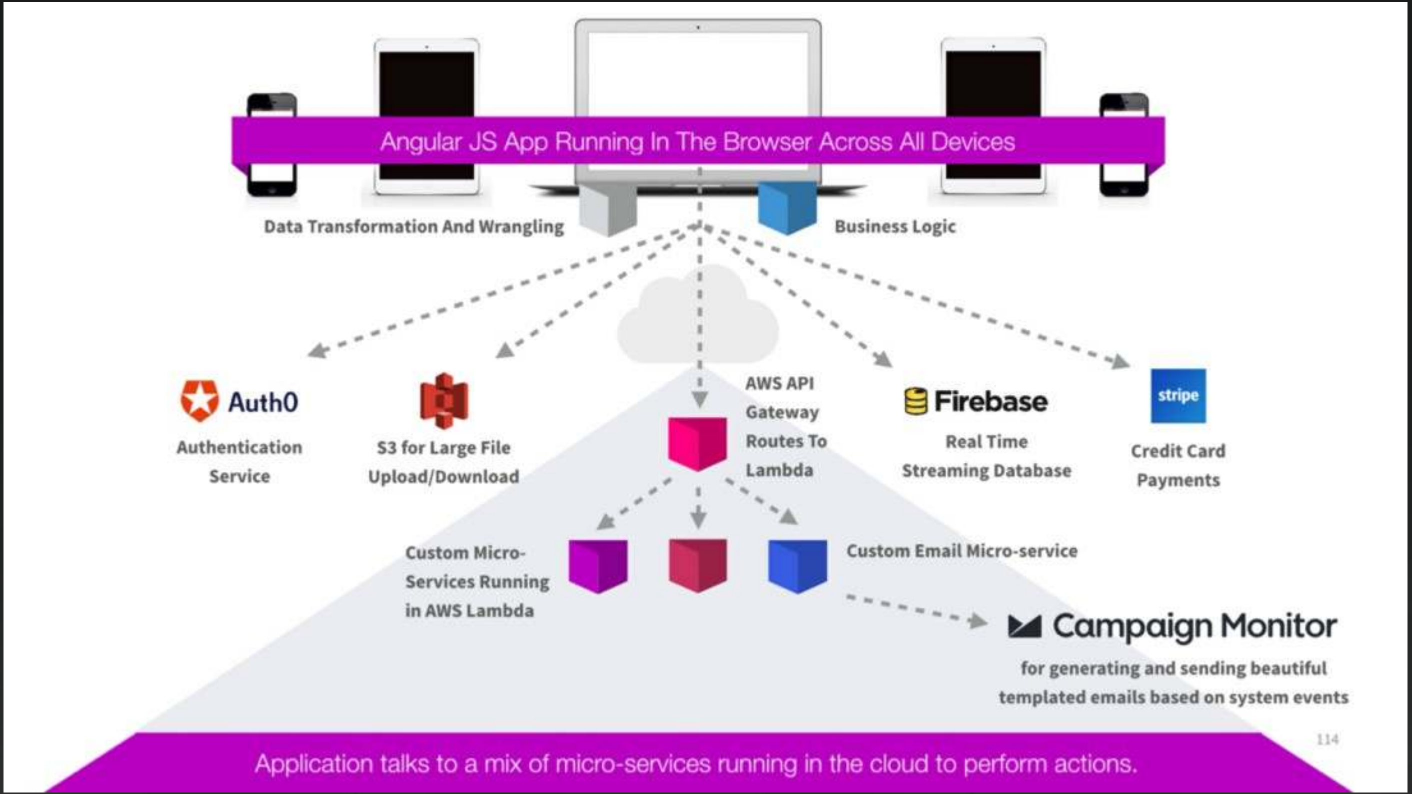
- Build event-driven applications

- Build APIs together with Amazon API Gateway
  - RESTful APIs
  - Resources, methods
  - Stages

http://aws.amazon.com/apigateway

A Cloud Guru: 100% Serverless

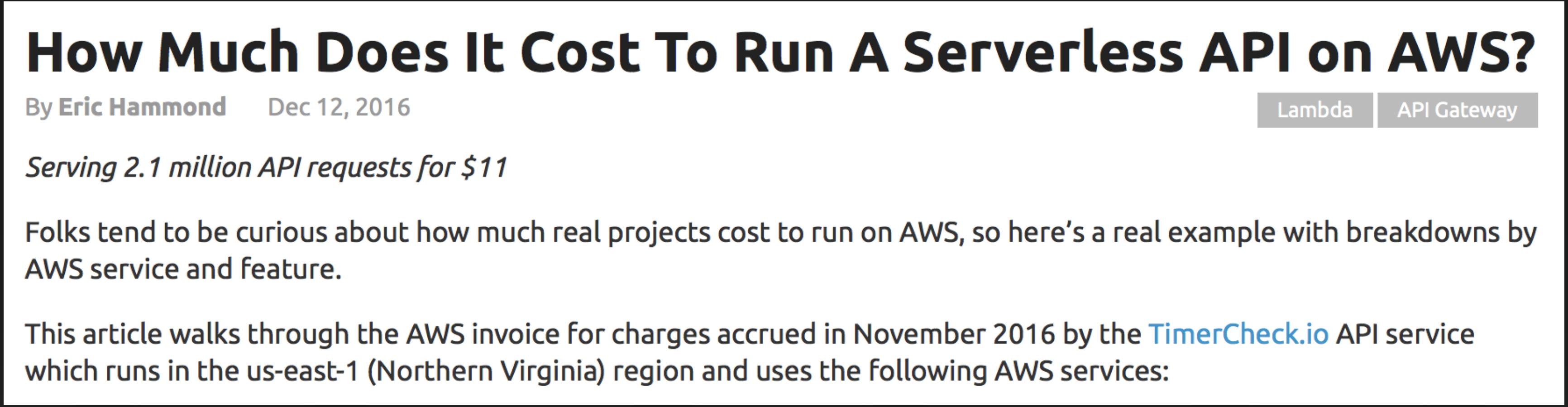
A CLOUD GURU



Angular JS App Running In The Browser Across All Devices
Data Transformation And Wrangling
Business Logic
Auth0
Authentication Service
S3 for Large File Upload/Download
AWS API Gateway Routes To Lambda
Firebase
Real Time Streaming Database
stripe
Credit Card Payments
Custom Micro-Services Running in AWS Lambda
Custom Email Micro-service
Campaign Monitor
for generating and sending beautiful templated emails based on system events
114
Application talks to a mix of micro-services running in the cloud to perform actions.

https://read.acloud.guru/serverless-the-future-of-software-architecture-d4473ffed864

amazon web services



How Much Does It Cost To Run A Serverless API on AWS?

By **Eric Hammond** Dec 12, 2016 Lambda API Gateway

Serving 2.1 million API requests for $11

Folks tend to be curious about how much real projects cost to run on AWS, so here's a real example with breakdowns by AWS service and feature.

This article walks through the AWS invoice for charges accrued in November 2016 by the TimerCheck.io API service which runs in the us-east-1 (Northern Virginia) region and uses the following AWS services:

https://alestic.com/2016/12/aws-invoice-example/

Typical development workflow

1. Write and deploy a Lambda function
2. Create and deploy a REST API with API Gateway
3. Connect the API to the Lambda function
4. Invoke the API
5. Test, debug and repeat ;)

Simplifying Development

Code samples available at https://github.com/juliensimon/aws/tree/master/lambda_frameworks

The Serverless framework

formerly known as JAWS: Just AWS Without Servers

- Announced at re:Invent 2015 by Austen Collins and Ryan Pendergast
- Supports Node.js, as well as Python and Java (with restrictions)
- Auto-deploys and runs Lambda functions, locally or remotely
- Auto-deploys your Lambda event sources: API Gateway, S3, DynamoDB, etc.
- Creates all required infrastructure with CloudFormation
- Simple configuration in YML

http://github.com/serverless/serverless
https://serverless.com
AWS re:Invent 2015 | (DVO209) https://www.youtube.com/watch?v=D_U6luQ6I90 & https://vimeo.com/141132756

Serverless: “Hello World” API

```
$ serverless create
```

Edit handler.js, serverless.yml and event.json

```
$ serverless deploy [--stage stage_name]

$ serverless invoke [--local] --function function_name

$ serverless info

$ http $URL
```

Gordon

- Released in Oct’15 by Jorge Batista
- Supports Python, Javascript, Golang, Java, Scala, Kotlin (including in the same project)
- Auto-deploys and runs Lambda functions, locally or remotely
- Auto-deploys your Lambda event sources: API Gateway, CloudWatch Events, DynamoDB Streams, Kinesis Streams, S3
- Creates all required infrastructure with CloudFormation
- Simple configuration in YML

https://github.com/jorgebastida/gordon
https://news.ycombinator.com/item?id=11821295

Gordon: “Hello World” API

```
$ gordon startproject hellonode

$ gordon startapp hello

Write function

$ gordon build

$ echo '{"name":"Julien"}' | gordon run hello.helloworld

$ gordon apply [--stage stage_name]

$ http post $URL name=Julien
```

AWS Chalice

Think of it as a serverless framework for Flask apps

- Released in Jul'16, still in beta
- Just add your Python code
  - Deploy with a single call and zero config
  - The API is created automatically, the IAM policy is auto-generated
- Run APIs locally on port 8000 (similar to Flask)
- Fast & lightweight framework
  - 100% *boto3* calls (AWS SDK for Python) → fast
  - No integration with CloudFormation → no creation of event sources

https://github.com/awslabs/chalice
https://aws.amazon.com/blogs/developer/preview-the-python-serverless-microframework-for-aws/

AWS Chalice: “Hello World” API

```
$ chalice new-project helloworld
```

Write your function in app.py

```
$ chalice local

$ chalice deploy

$ export URL=`chalice url`

$ http $URL

$ http put $URL/hello/julien

$ chalice logs [ --include-lambda-messages ]
```

AWS Chalice: PUT/GET in S3 bucket

```
$ chalice new-project s3test

Write your function in app.py

$ chalice local

$ http put http://localhost:8000/objects/doc.json value1=5 value2=8

$ http get http://localhost:8000/objects/doc.json

$ chalice deploy [stage_name]

$ export URL=`chalice url`

$ http put $URL/objects/doc.json value1=5 value2=8

$ http get $URL/objects/doc.json
```

Summing things up

Serverless

The most popular serverless framework

Built with and for Node.js. Python and Java: YMMV

Rich features, many event sources

Not a web framework

Gordon

Great challenger!

Node.js, Python, Java, Scala, Golang

Comparable to Serverless feature-wise

Not a web framework

Chalice

AWS project, in beta

Python only

Does only one thing, but does it great

Dead simple, zero config

Flask web framework

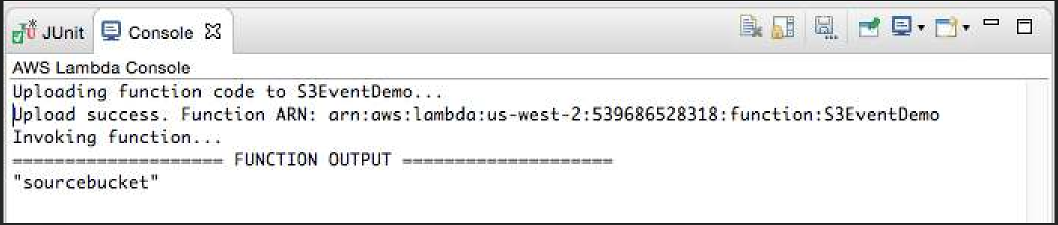
More Lambda frameworks

- Kappa https://github.com/garnaat/kappa
  - Released Dec'14 by Mitch Garnaat, author of boto and the AWS CLI (still maintained?)
  - Python only, multiple event sources
- Apex https://github.com/**apex**/**apex**
  - Released in Dec'15 by TJ Holowaychuk
  - Python, Javascript, Java, Golang
  - Terraform integration to manage infrastructure for event sources
- Zappa https://github.com/Miserlou/Zappa
  - Released in Feb'16 by Rich Jones
  - Python web applications on AWS Lambda + API Gateway
- Docker-lambda https://github.com/lambci/docker-lambda
  - Released in May'16 by Michael Hart
  - Run functions in Docker images that "replicate" the live Lambda environment

2 Java tools for AWS Lambda

Eclipse plug-in

- Code, test and deploy Lambdas from Eclipse
- Run your functions locally and remotely
- Test with local events and Junit4
- Deploy standalone functions, or with the AWS Serverless Application Model (Dec'16)



```
AWS Lambda Console
Uploading function code to S3EventDemo...
Upload success. Function ARN: arn:aws:lambda:us-west-2:539686528318:function:S3EventDemo
Invoking function...
==================== FUNCTION OUTPUT ====================
"sourcebucket"
```

Serverless Java Container

- Run Java RESTful APIs as-is
- Default implementation of the Java servlet `HttpServletRequest` `HttpServletResponse`
- Support for Java frameworks such as Jersey or Spark

https://java.awsblog.com/post/TxWZES6J1RSQ2Z/Testing-Lambda-functions-using-the-AWS-Toolkit-for-Eclipse
https://aws.amazon.com/blogs/developer/aws-toolkit-for-eclipse-serverless-application
https://github.com/awslabs/aws-serverless-java-container

Simplifying Deployment

AWS Serverless Application Model (SAM)

formerly known as Project Flourish

- CloudFormation extension released in Nov'16 to bundle Lambda functions, APIs & events
- 3 new CloudFormation resource types
  - *AWS::Serverless::Function*
  - *AWS::Serverless::Api*
  - *AWS::Serverless::SimpleTable*
- 2 new CloudFormation CLI commands
  - *'aws cloudformation package'*
  - *'aws cloudformation deploy'*
- Integration with CodeBuild and CodePipeline for CI/CD
- Expect SAM to be integrated in most / all frameworks

https://aws.amazon.com/fr/blogs/compute/introducing-simplified-serverless-application-deplyoment-and-management
https://github.com/awslabs/serverless-application-model/

amazon web services

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Description: Get items from a DynamoDB table.
Resources:
 GetFunction:
  Type: AWS::Serverless::Function
  Properties:
   Handler: index.get
   Runtime: nodejs4.3
   Policies: AmazonDynamoDBReadOnlyAccess
   Environment:
    Variables:
     TABLE_NAME: !Ref Table
   Events:
    GetResource:
     Type: Api
     Properties:
      Path: /resource/{resourceId}
      Method: get
 Table:
  Type: AWS::Serverless::SimpleTable
```

Sample SAM template for:

- Lambda function
- HTTP GET API
- DynamoDB table

Going further

New Lambda videos from re:Invent 2016

AWS re:Invent 2016: What's New with AWS Lambda (SVR202)https://www.youtube.com/watch?v=CwxWhyGteNc

AWS re:Invent 2016: Serverless Apps with AWS Step Functions (SVR201)
https://www.youtube.com/watch?v=75MRve4nv8s

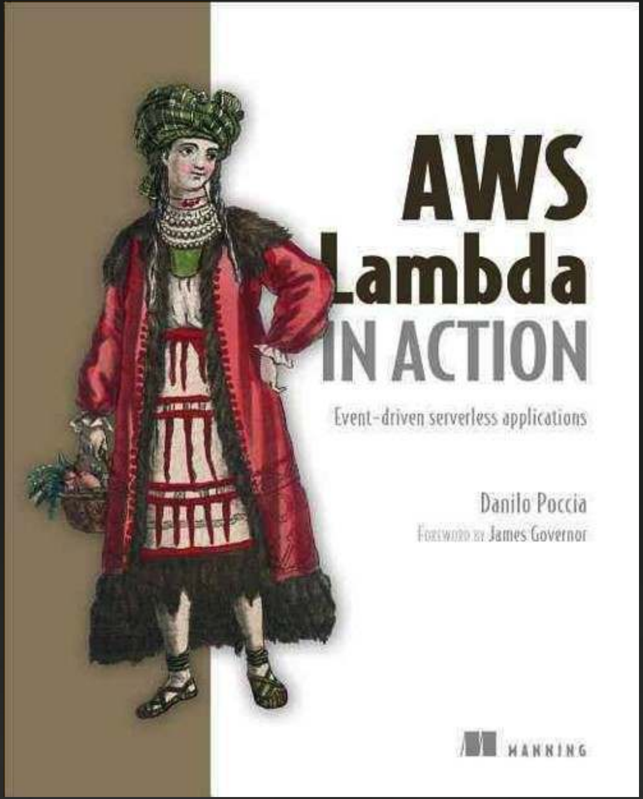
AWS re:Invent 2016: Real-time Data Processing Using AWS Lambda (SVR301)
https://www.youtube.com/watch?v=VFLKOy4GKXQ

AWS re:Invent 2016: Serverless Architectural Patterns and Best Practices (ARC402)
https://www.youtube.com/watch?v=b7UMoc1iUYw

AWS re:Invent 2016: Bringing AWS Lambda to the Edge (CTD206)
https://www.youtube.com/watch?v=j26novaqF6M

AWS re:Invent 2016: Ubiquitous Computing with Greengrass (IOT201)
https://www.youtube.com/watch?v=XQQjX8GTEko

The only Lambda book you need to read


AWS
Lambda
IN ACTION
Event-driven serverless applications
Danilo Poccia
Foreword by James Governor
MANNING

Written by AWS Technical Evangelist Danilo Poccia

Just released!

https://www.amazon.com/Aws-Lambda-Action-Event-driven-Applications/dp/1617293717/

Thank you!

Julien Simon, Principal Technical Evangelist, AWS
julsimon@amazon.fr
@julsimon