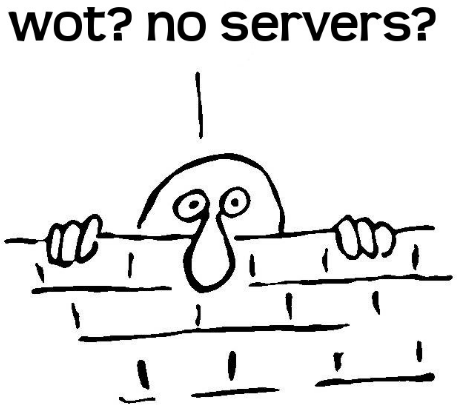
Serverless Architecture



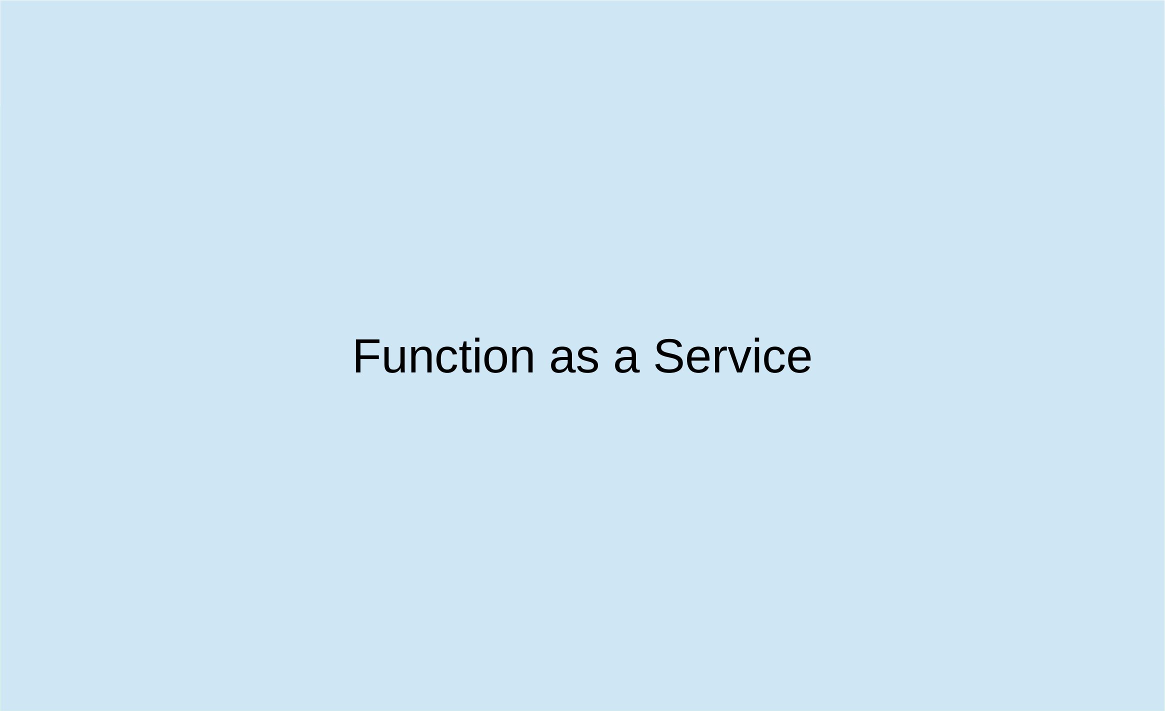
Alex Curtis
CTO, Icarusparts.com Ltd

@statenlogic

MK.js - 4th April 2017

Serverless Architecture

```
function f(s) { return s.toUpperCase(); }
```



Function as a Service

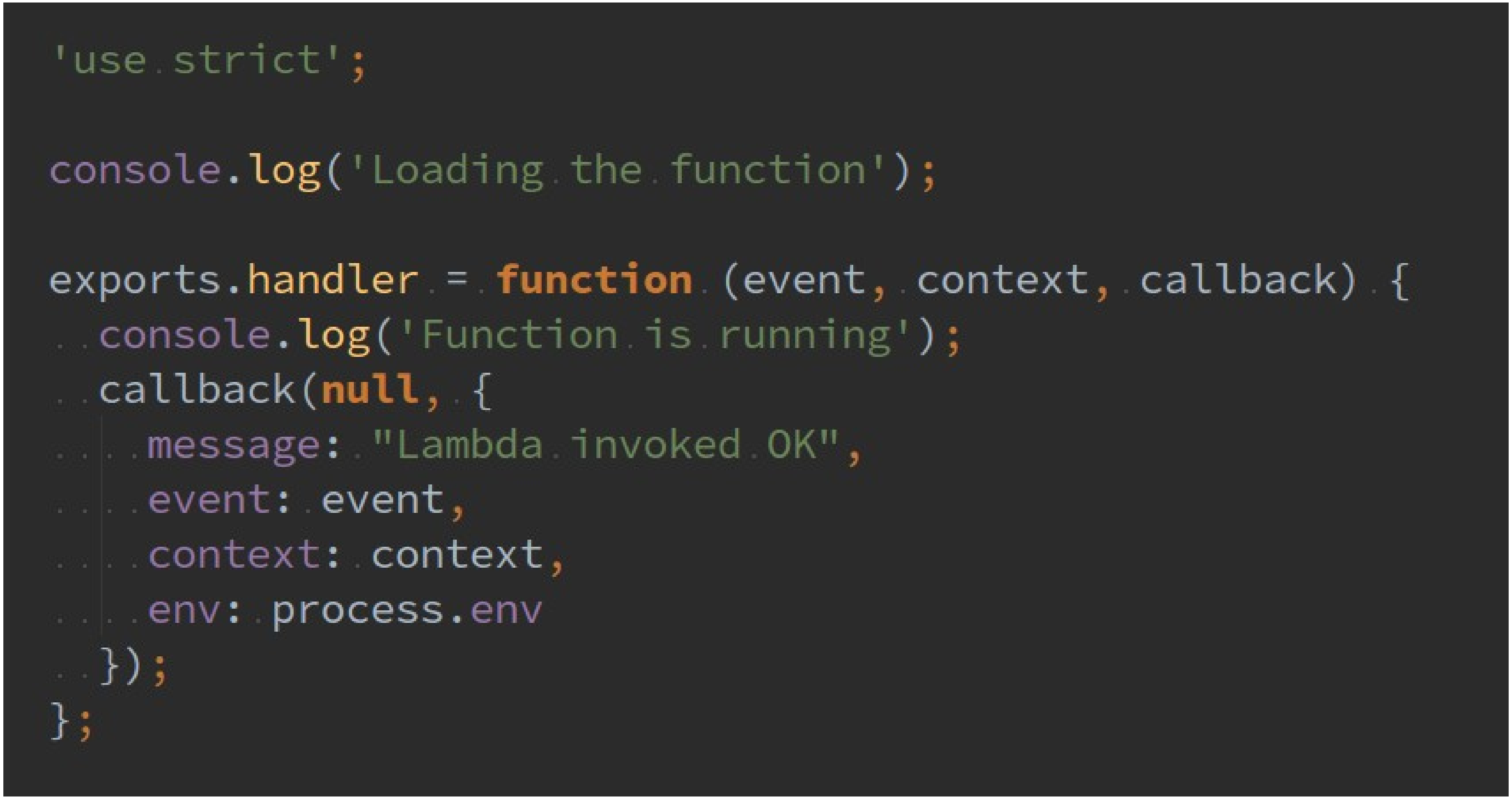
AWS Lambda

Amazon Web Services enables serverless architecture with *AWS Lambda*:

- Functions invoked in response to events
- Written in Javascript, Python, or JVM languages
- Really easy to deploy
- Run in the cloud - no persistent file system
- Constrained in size and running time
- Pay - as - you - use

A Simple Example

simple.js



```
'use strict';

console.log('Loading the function');

exports.handler = function (event, context, callback) {
  console.log('Function is running');
  callback(null, {
    message: "Lambda invoked OK",
    event: event,
    context: context,
    env: process.env
  });
};
```

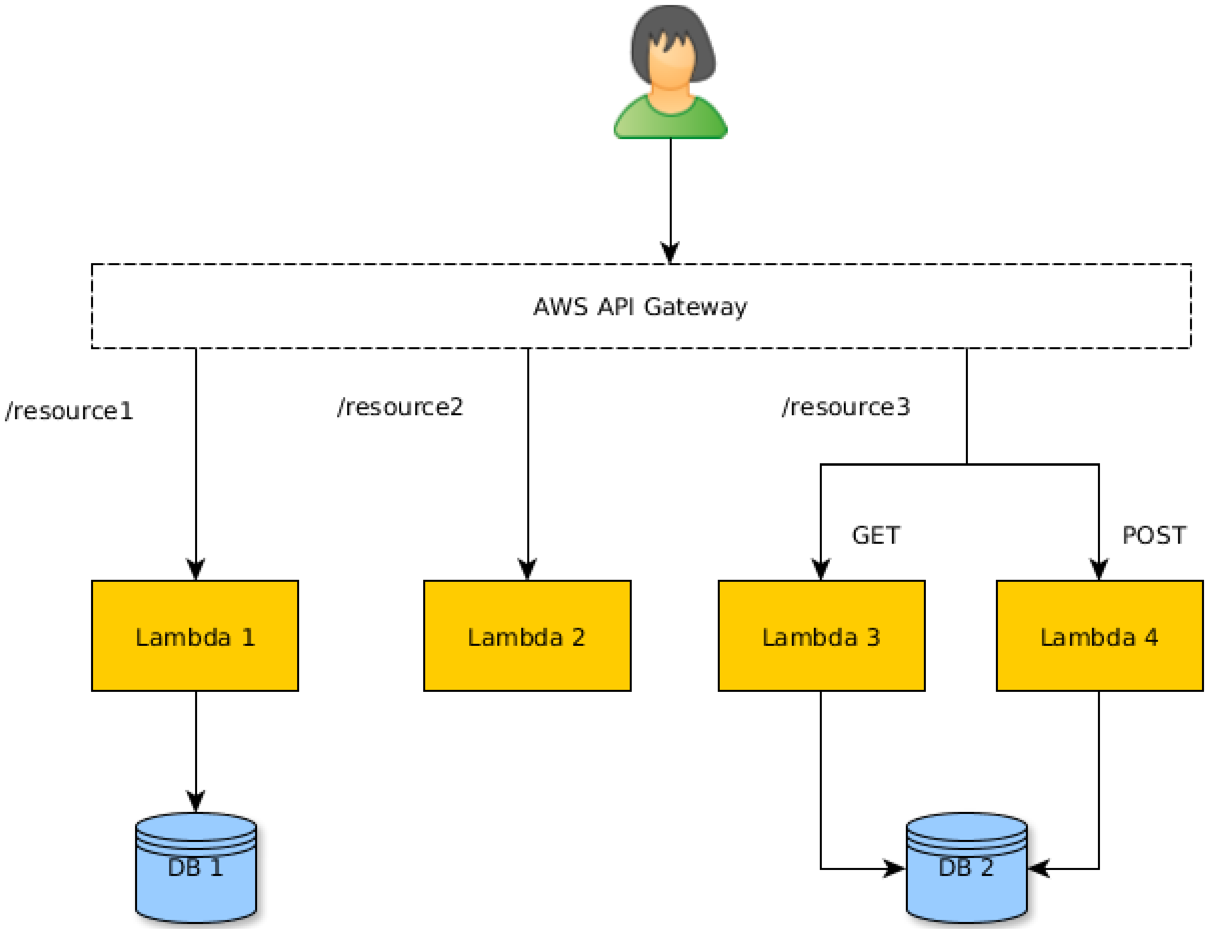
Deployment

```
$ zip -r simple-package.zip simple.js

$ aws lambda create-function \
   --function-name SimpleLambda \
   --role arn:aws:iam::973567109463:role/lamb1 \
   --runtime nodejs6.10 \
   --handler simple.handler \
   --zip-file fileb://simple-package.zip
```

API Gateway

A REST-ful HTTP interface to AWS Lambdas

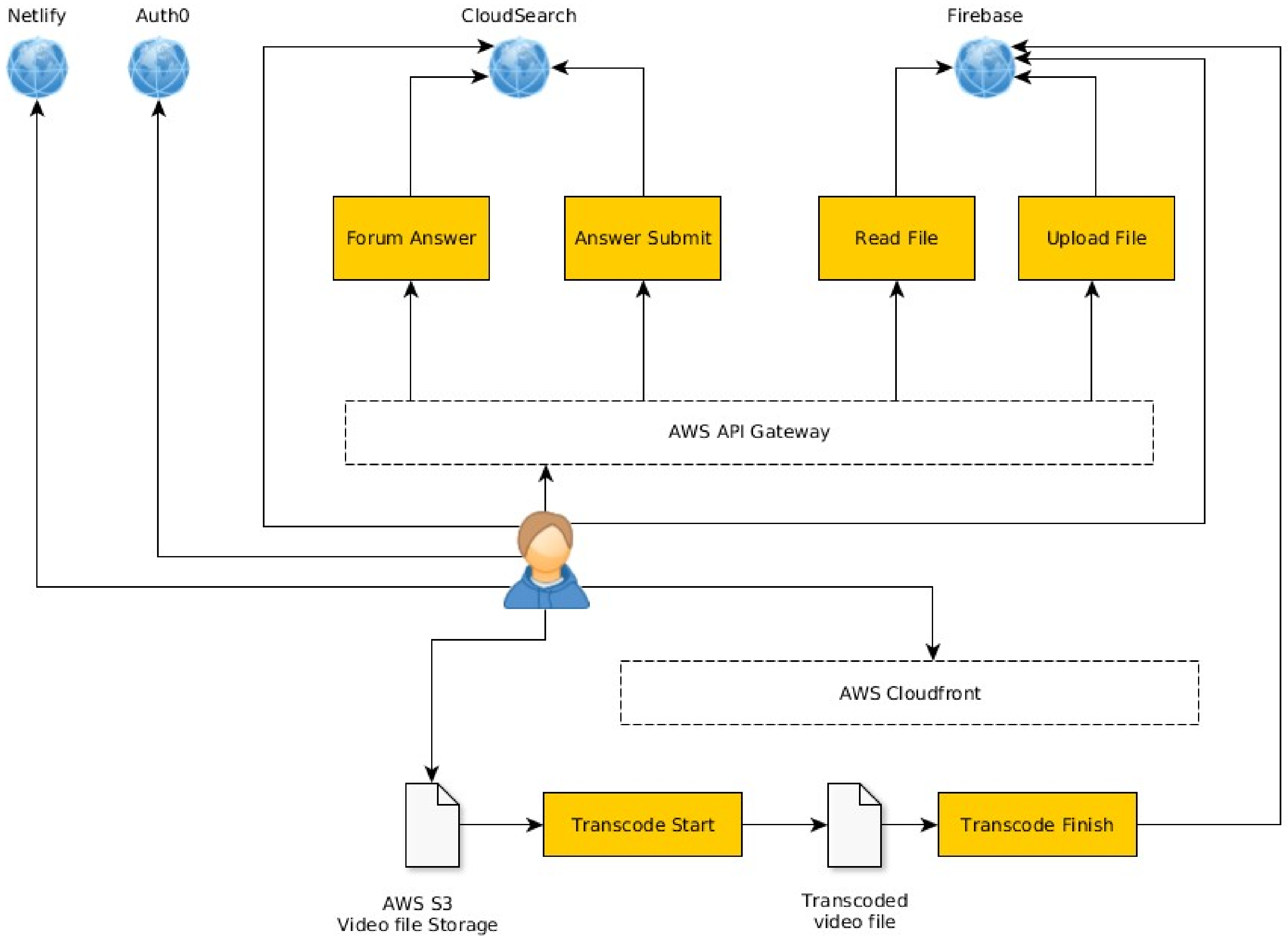


API Gateway

Fiddly to set up as you need to configure the following AWS resources

- `AWS::IAM::Role`
- `AWS::ApiGateway::Account`
- `AWS::ApiGateway::RestApi`
- `AWS::ApiGateway::Deployment`
- `AWS::ApiGateway::Stage`
- `AWS::ApiGateway::BasePathMapping`
- `AWS::ApiGateway::Resource`
- `AWS::Lambda::Permission`
- `AWS::ApiGateway::Method`

100% Serverless Architecture



Pros

Good things about Serverless and AWS Lambda:

- Fewer layers to manage - focus just on code
- Quick to get started
- Quick way to build out back-end
- Really easy to scale
- Encourages modular architecture
- Mostly cheap

Cons

Things to consider when deciding on Serverless architecture with AWS Lambda:

- Pure Serverless relies heavily on 3rd parties
- Lambda not seamlessly integrated with rest of AWS
- Not good for daemon type services
- Limited running time
- Can get complex quite easily
- Harder to do integration testing

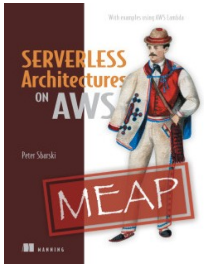
Conclusion

Will it take off?

- Serverless is the natural next-step into the cloud
- Easy to imagine a future where Serverless is the norm
- Current state of the art still has holes
  - Right now Lamdas are best suited in practice to simple event handling tasks

Resources

- AWS Lambda documentation
- Serverless Framework
  - https://serverless.com/
- GitHub source
  - https://github.com/alexandercurtis/serverless-demo
- Book
  - https://www.manning.com/books/serverless-architectures-on-aws



Me: @statenlogic , alexandercurtis

Backup

Thoughtworks

"*We recommend that Lambda functions contain only a moderate amount of code. Ensuring the quality of a solution based on a tangle of many large Lambda functions is difficult, and such a solution may not be cost-effective. For more complex needs, deployments based on containers or VMs are still preferable.*"

– Thoughtworks Tech Radar Q1 2017

Manual Invocation

```
$ aws lambda invoke \
        --invocation-type RequestResponse \
        --function-name SimpleLambda \
        --region eu-west-1 \
        --log-type Tail \
        --payload '{"key1":"value1", "key2":"value2", "key3":"value3"}' \
        out.json
```

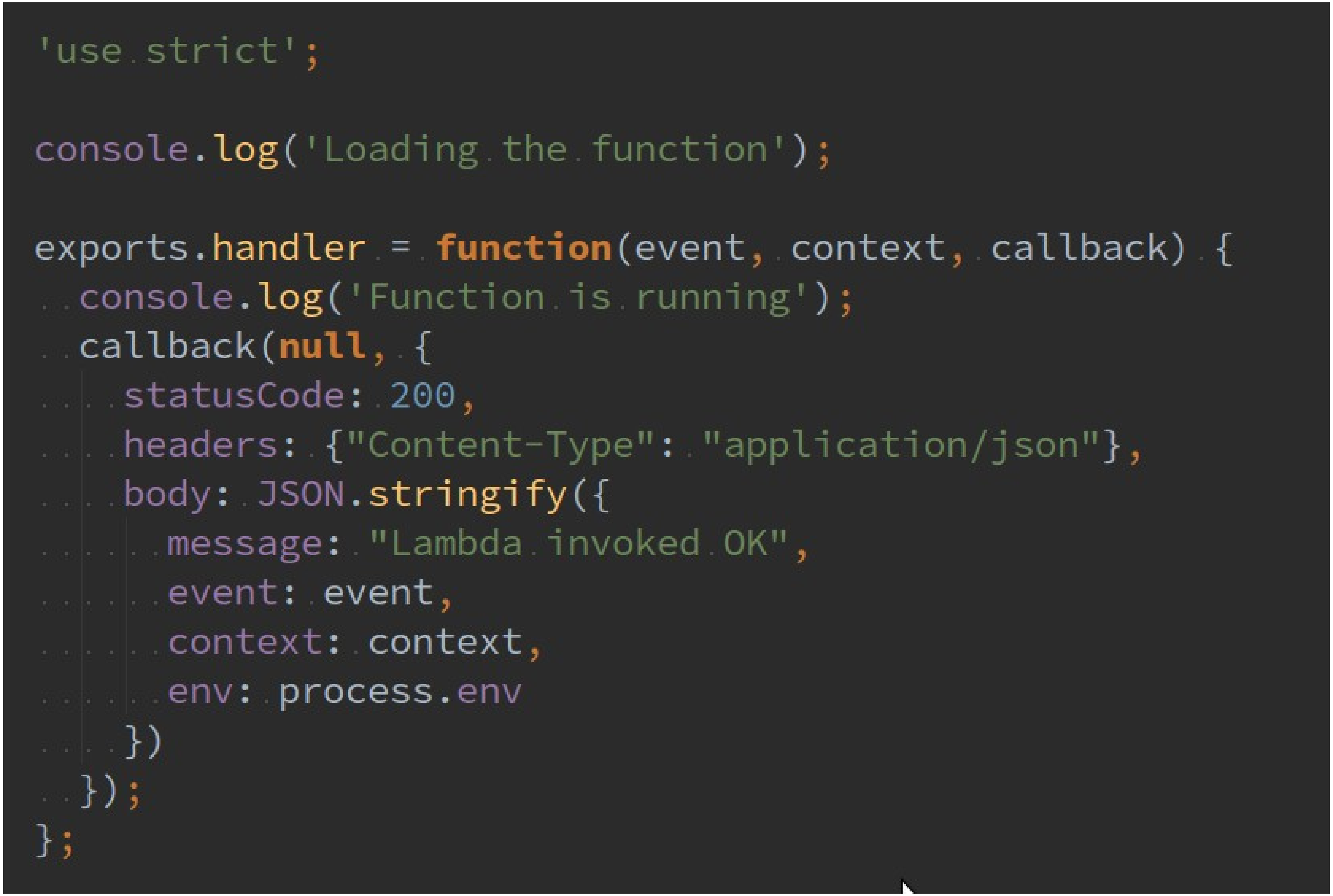
```
{
   "LogResult":
"U1RBUlQgUmVxdWVzdElkOiAwZmZiYzRjNC0xNzlhLTExZTctYjY1ZC0wZGM1YzA2Mj
UwODYgVmVyc2lvbjogJExBVEVTVAoyMDE3LTA0LTAyVDExOjQ2OjUxLjkxNloJMGZm
YmM0YzQtMTc5YS0xMWU3LWI2NWQtMGRjNWMwNjI1MDg2CUZ1bmN0aW9uIGlzIH
J1bm5pbmcKRU5EIFJlcXVlc3RJZDogMGZmYmM0YzQtMTc5YS0xMWU3LWI2NWQtM
GRjNWMwNjI1MDg2ClJFUE9SVCBSZXF1ZXN0SWQ6IDBmZmJjNGM0LTE3OWEtMT
FlNy1iNjVkLTBkYzVjMDYyNTA4NglEdXJhdGlvbjogMC41MyBtcwlCaWxsZWQgRHVyYX
Rpb246IDEwMCBtcyAJTWVtb3J5IFNpemU6IDEyOCBNQglNYXggTWVtb3J5IFVzZWQ
6IDE1IE1CCQo=",
   "StatusCode": 200
}
```

Response Payload

```
{
  "message": "Lambda invoked OK",
  "event": {
    "key1": "value1",
    "key2": "value2",
    "key3": "value3"
  },

  "context": {
    "callbackWaitsForEmptyEventLoop": true,
    "logGroupName": "/aws/lambda/SimpleLambda",
    "logStreamName": "2017/04/02/
[$LATEST]0390950c3e534ea0b4096d26b0dbbe56",
    "functionName": "SimpleLambda",
    "memoryLimitInMB": "128",
    "functionVersion": "$LATEST",
    "invokeid": "8515287a-179d-11e7-8ac0-7d0b08a2cedf",
    "awsRequestId": "8515287a-179d-11e7-8ac0-7d0b08a2cedf",
    "invokedFunctionArn": "arn:aws:lambda:eu-west-
1:973567109463:function:SimpleLambda"
  },
}
```

Lambda Proxy Example



```
'use strict';

console.log('Loading the function');

exports.handler = function(event, context, callback) {
  console.log('Function is running');
  callback(null, {
    statusCode: 200,
    headers: {"Content-Type": "application/json"},
    body: JSON.stringify({
      message: "Lambda invoked OK",
      event: event,
      context: context,
      env: process.env
    })
  });
};
```