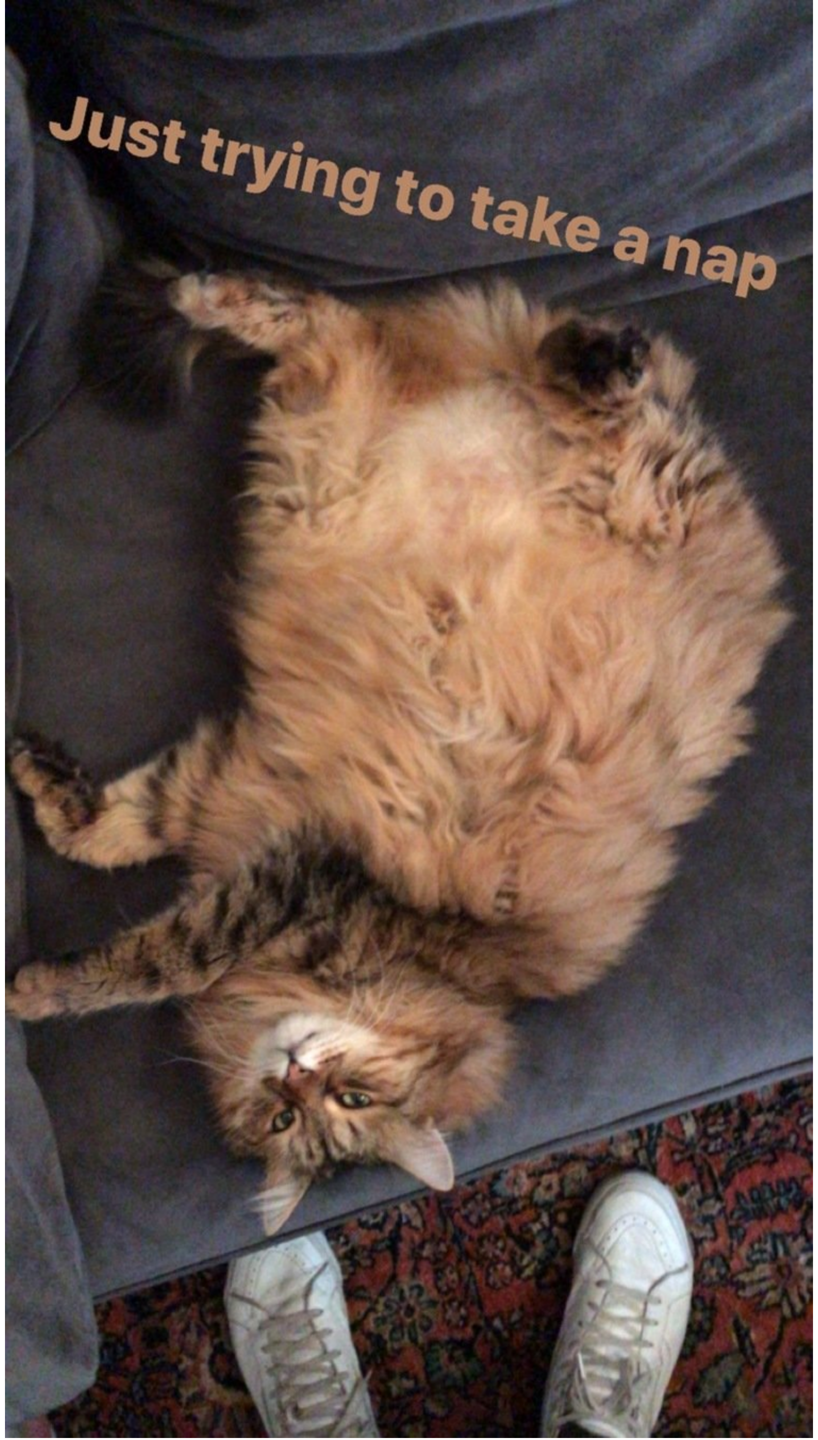
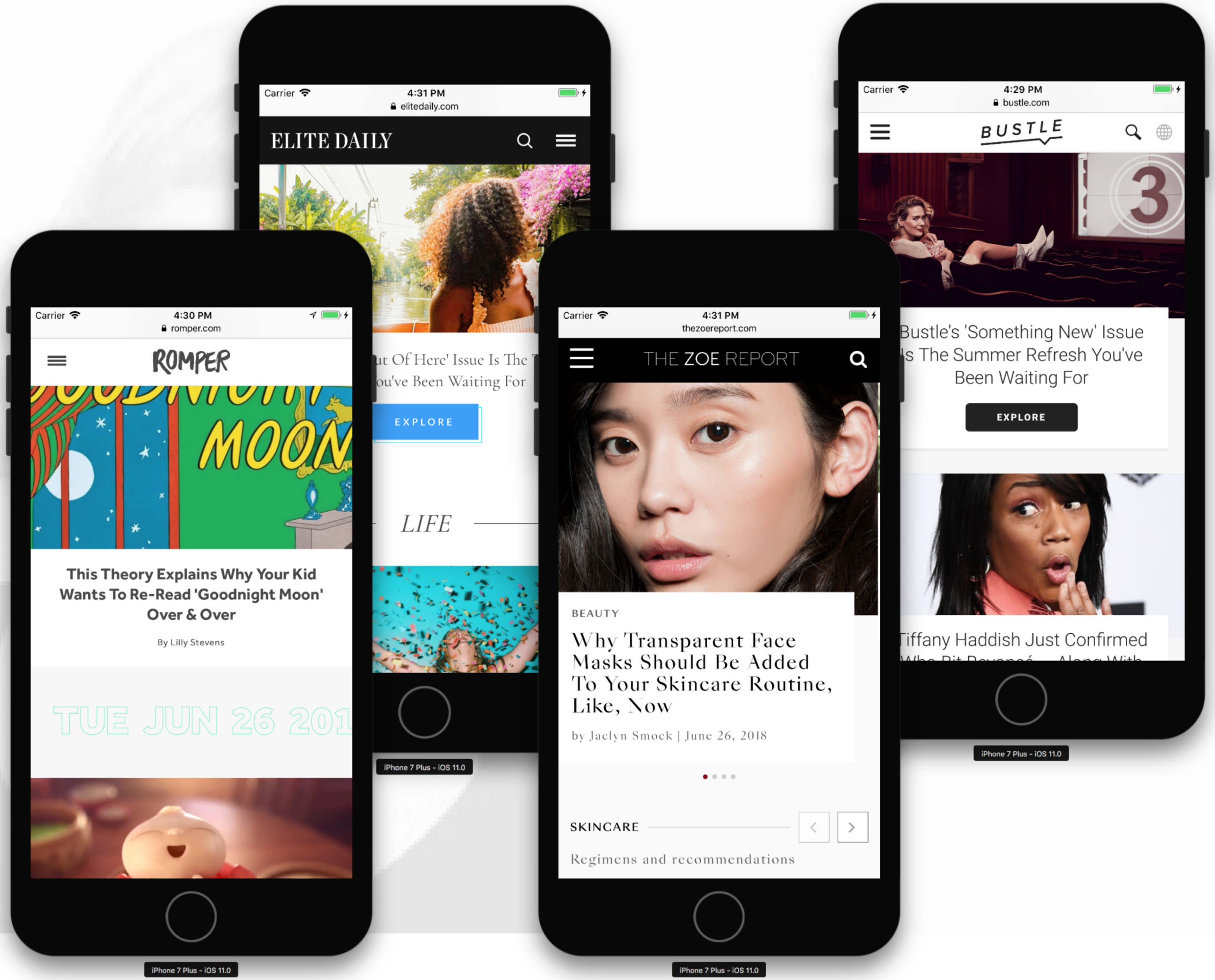
Hi

CATAN
CATAN



Just trying to take a nap

BUSTLE DIGITAL GROUP



Carrier 4:30 PM
romper.com

ROMPER

This Theory Explains Why Your Kid Wants To Re-Read 'Goodnight Moon' Over & Over

By Lilly Stevens

TUE JUN 26 201

iPhone 7 Plus - iOS 11.0

Carrier 4:31 PM
elitedaily.com

ELITE DAILY

ut Of Here' Issue Is The
ou've Been Waiting For

EXPLORE

LIFE

iPhone 7 Plus - iOS 11.0

Carrier 4:31 PM
thezoereport.com

THE ZOE REPORT

BEAUTY

Why Transparent Face Masks Should Be Added To Your Skincare Routine, Like, Now

by Jaclyn Smock | June 26, 2018

SKINCARE

Regimens and recommendations

iPhone 7 Plus - iOS 11.0

Carrier 4:29 PM
bustle.com

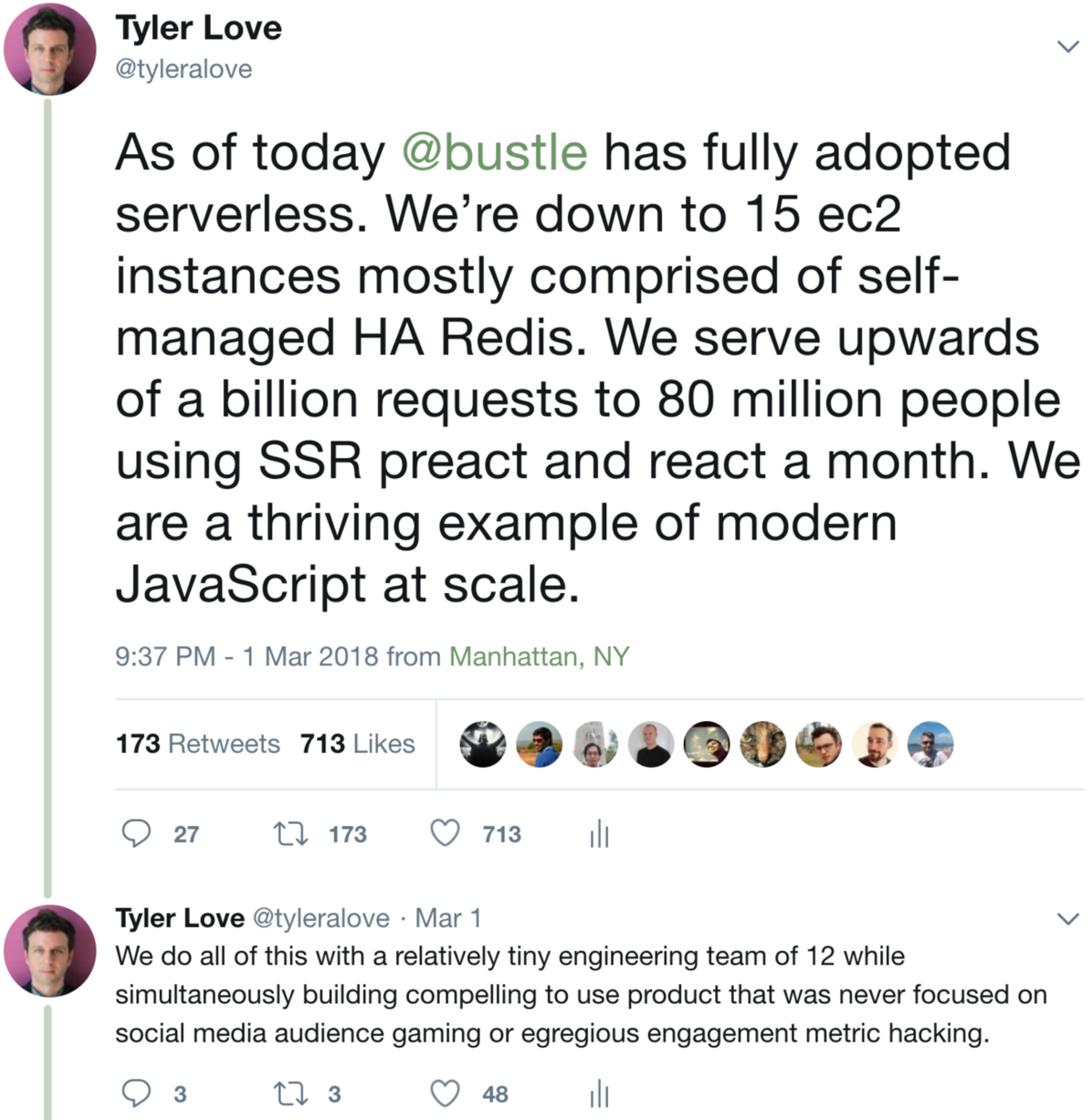
BUSTLE

Bustle's 'Something New' Issue Is The Summer Refresh You've Been Waiting For

EXPLORE

Tiffany Haddish Just Confirmed

iPhone 7 Plus - iOS 11.0



Tyler Love
@tyleralove

As of today @bustle has fully adopted serverless. We're down to 15 ec2 instances mostly comprised of self-managed HA Redis. We serve upwards of a billion requests to 80 million people using SSR preact and react a month. We are a thriving example of modern JavaScript at scale.

9:37 PM - 1 Mar 2018 from Manhattan, NY

173 Retweets **713** Likes

27 173 713

Tyler Love @tyleralove · Mar 1

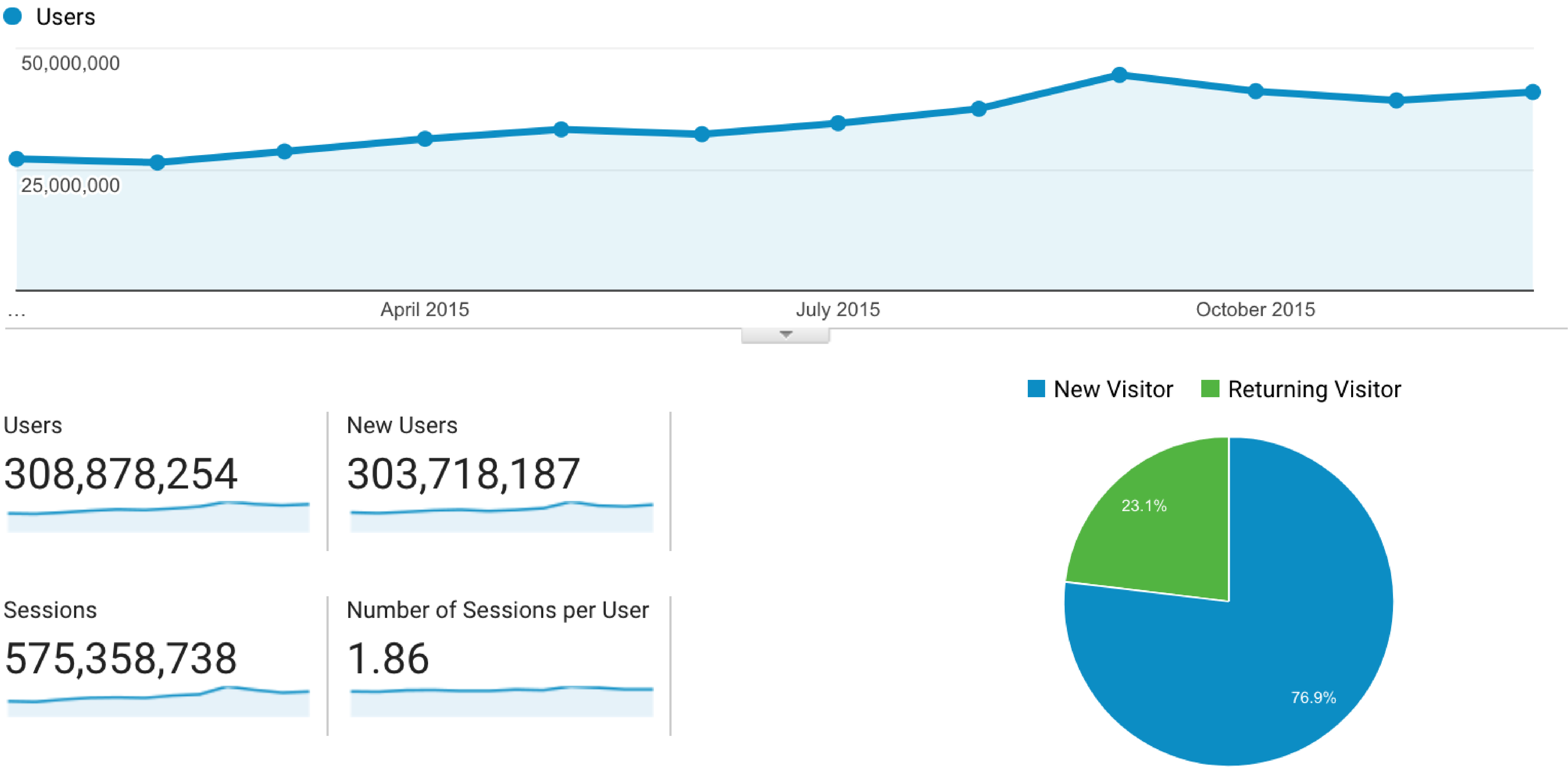
We do all of this with a relatively tiny engineering team of 12 while simultaneously building compelling to use product that was never focused on social media audience gaming or egregious engagement metric hacking.

3 3 48

Adoption

> *“ They seed in the upper branches of a fig tree and gradually work their way down the tree until they root in the soil. Over many years they grow into fantastic and beautiful shapes, meanwhile strangling and killing the tree that was their host.*
>
> Martin Fowler

Complexity

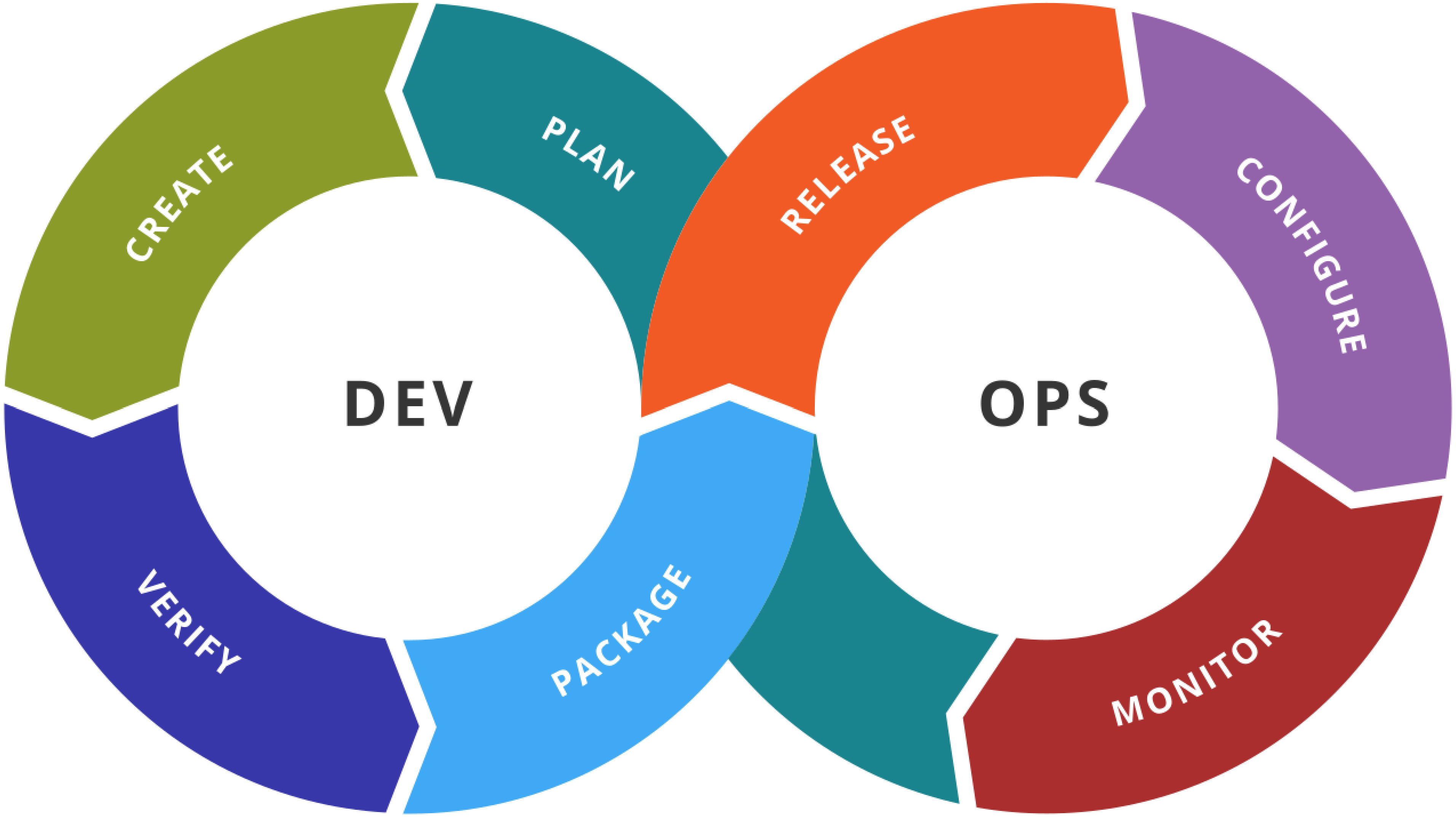


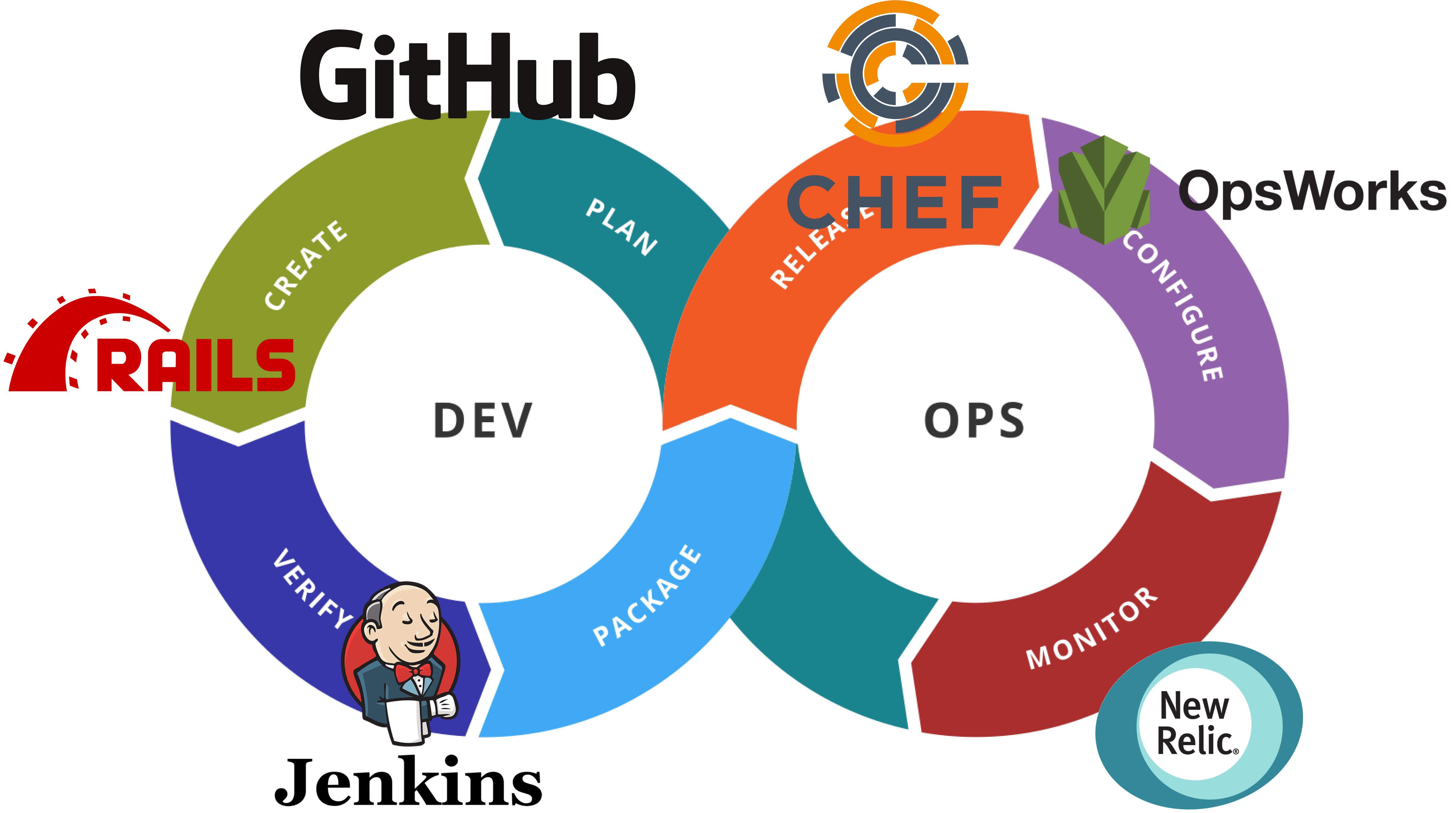
Users
308,878,254

New Users
303,718,187

Sessions
575,358,738

Number of Sessions per User
1.86


CREATE
PLAN
RELEASE
CONFIGURE
DEV
OPS
VERIFY
PACKAGE
MONITOR


GitHub
CHEF
OpsWorks
RAILS
CREATE
PLAN
RELEASE
CONFIGURE
DEV
OPS
VERIFY
PACKAGE
MONITOR
Jenkins
New Relic.

Jenkins

Challenges

- Installed on an EC2 instance
- Automated with Chef and AWS OpsWorks
- Authorization and access management
- Unique environment for running our applications

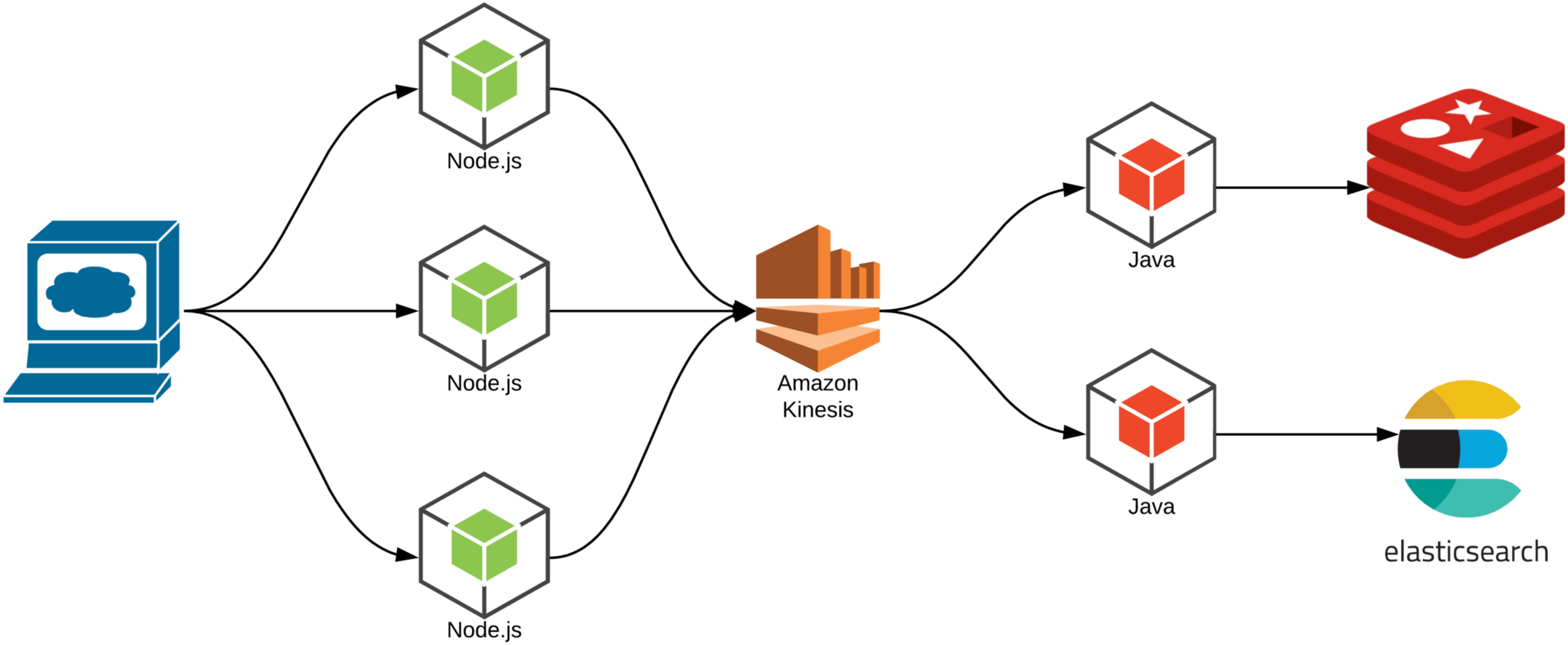
Challenges

- Expensive for every line
- 100% Domain knowledge coding
- Slow development cycle
- Didn't do everything we wanted it to
- Hard to incrementally adopt

Challenges

- Expensive.
- Layers of marketing people for a dev tool

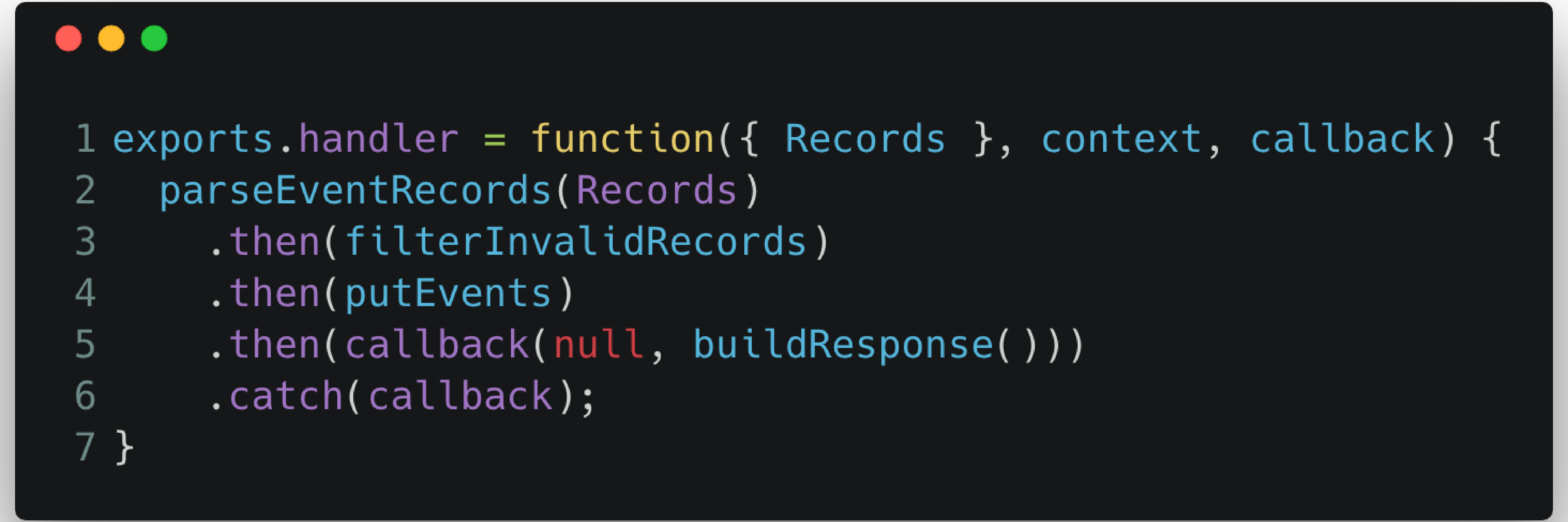
Client / Browser Collectors Stream Processors Metrics



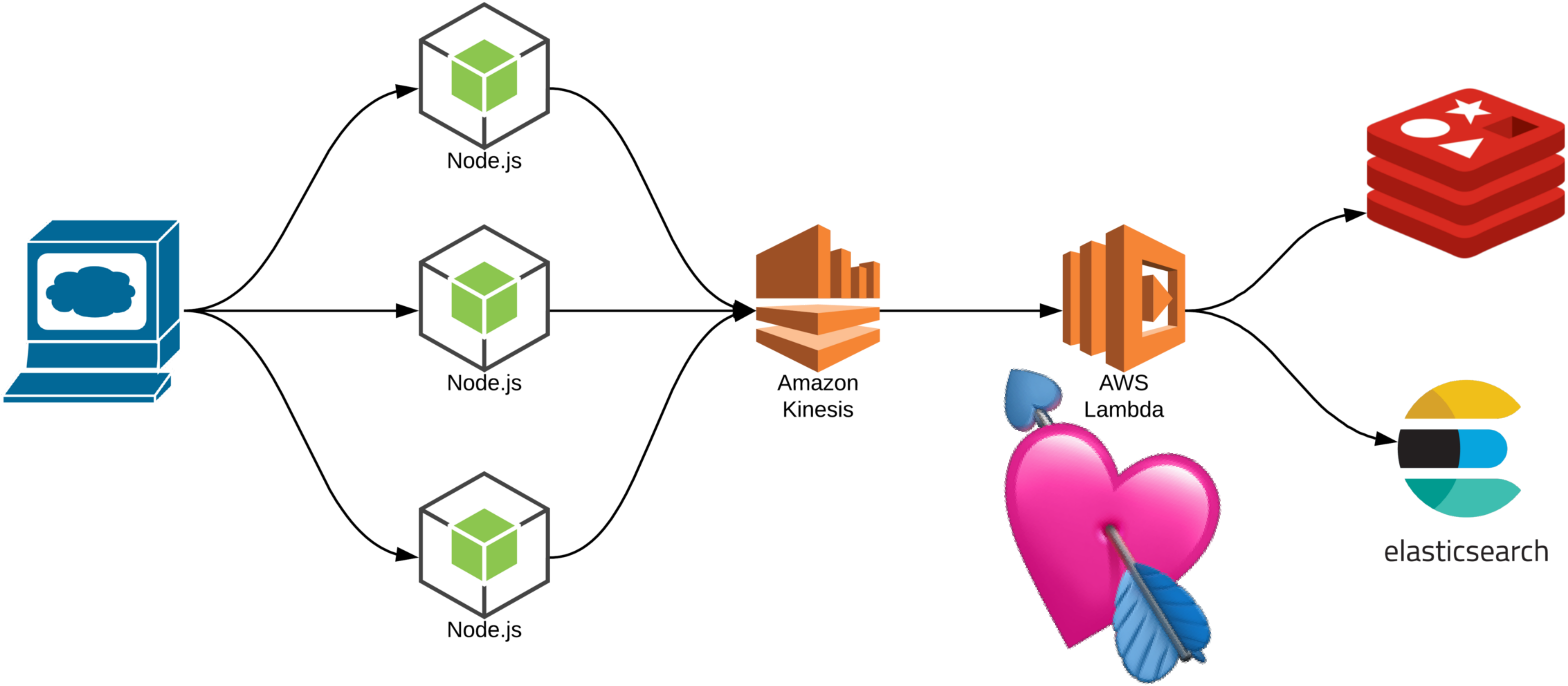
WERNER VOGELS
CTO, AMAZON.COM

AWS OpsWorks

Java



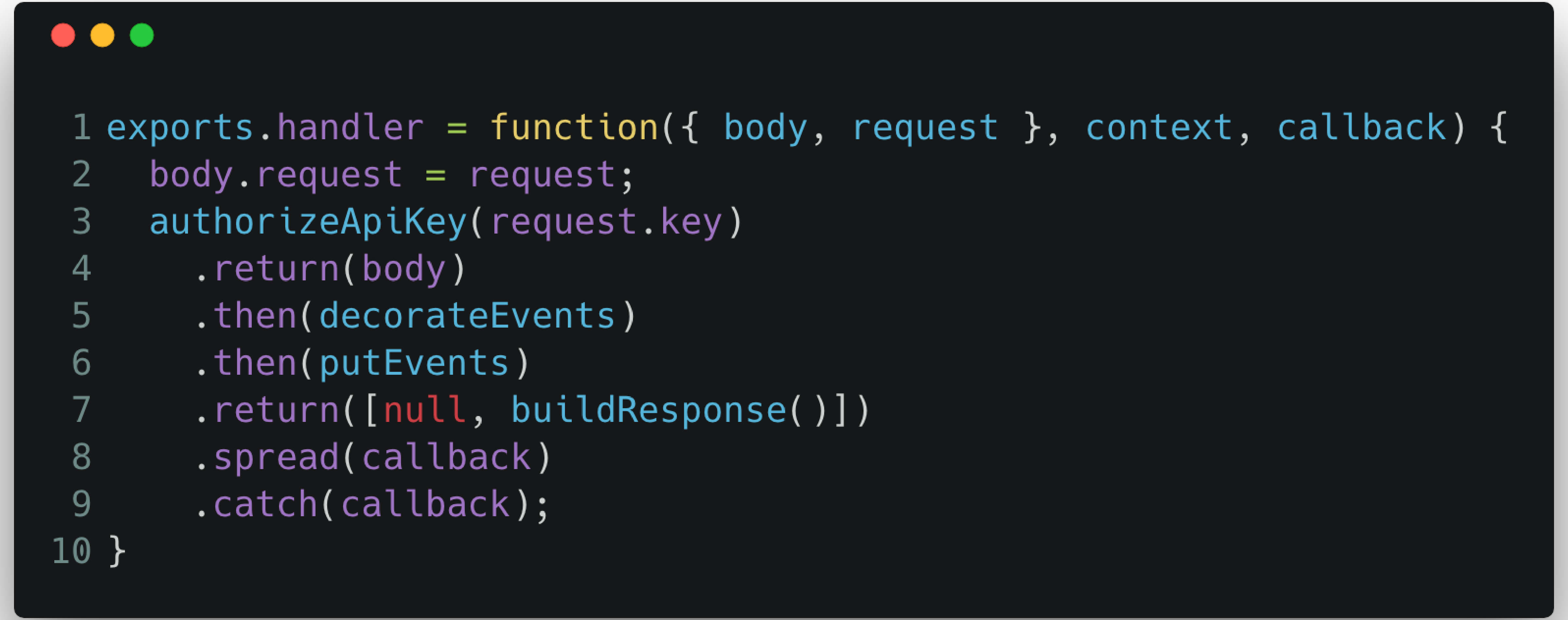
```javascript
exports.handler = function({ Records }, context, callback) {
  parseEventRecords(Records)
    .then(filterInvalidRecords)
    .then(putEvents)
    .then(callback(null, buildResponse()))
    .catch(callback);
}
```


Node.js
Node.js
Node.js
Amazon Kinesis
AWS Lambda
elasticsearch

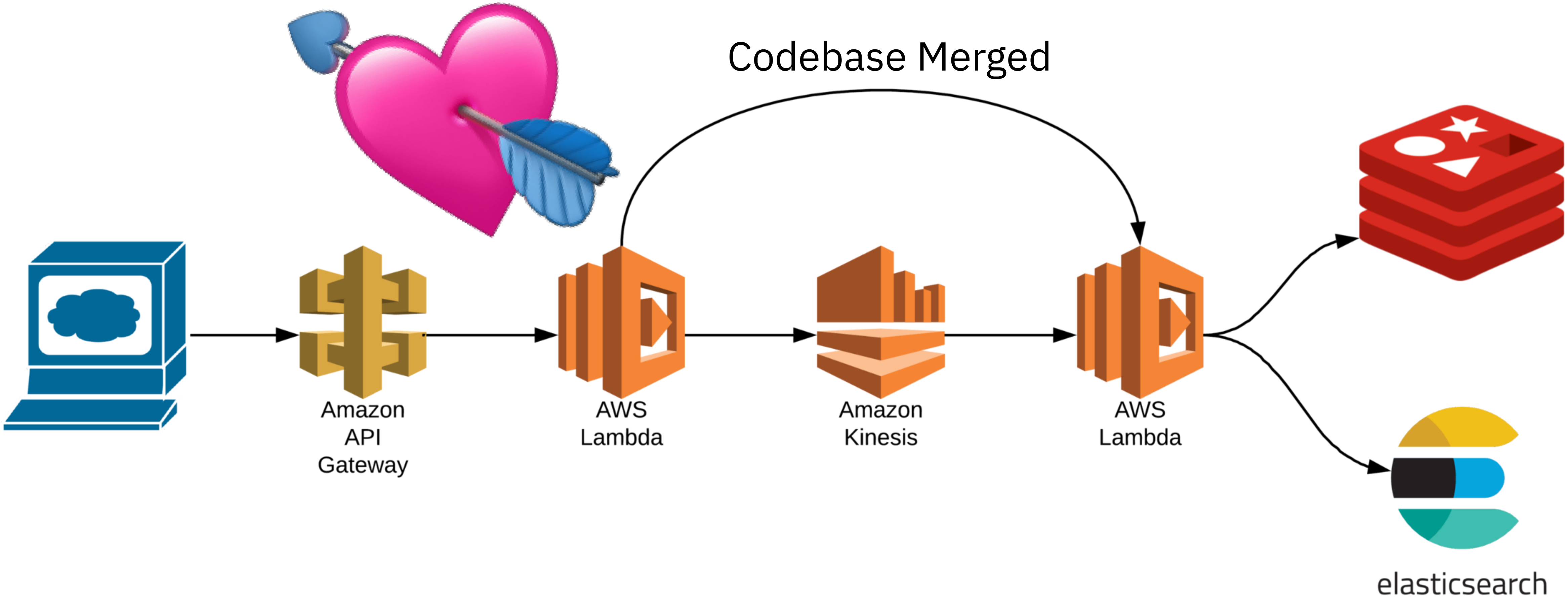
ructure in Min
Add Ne

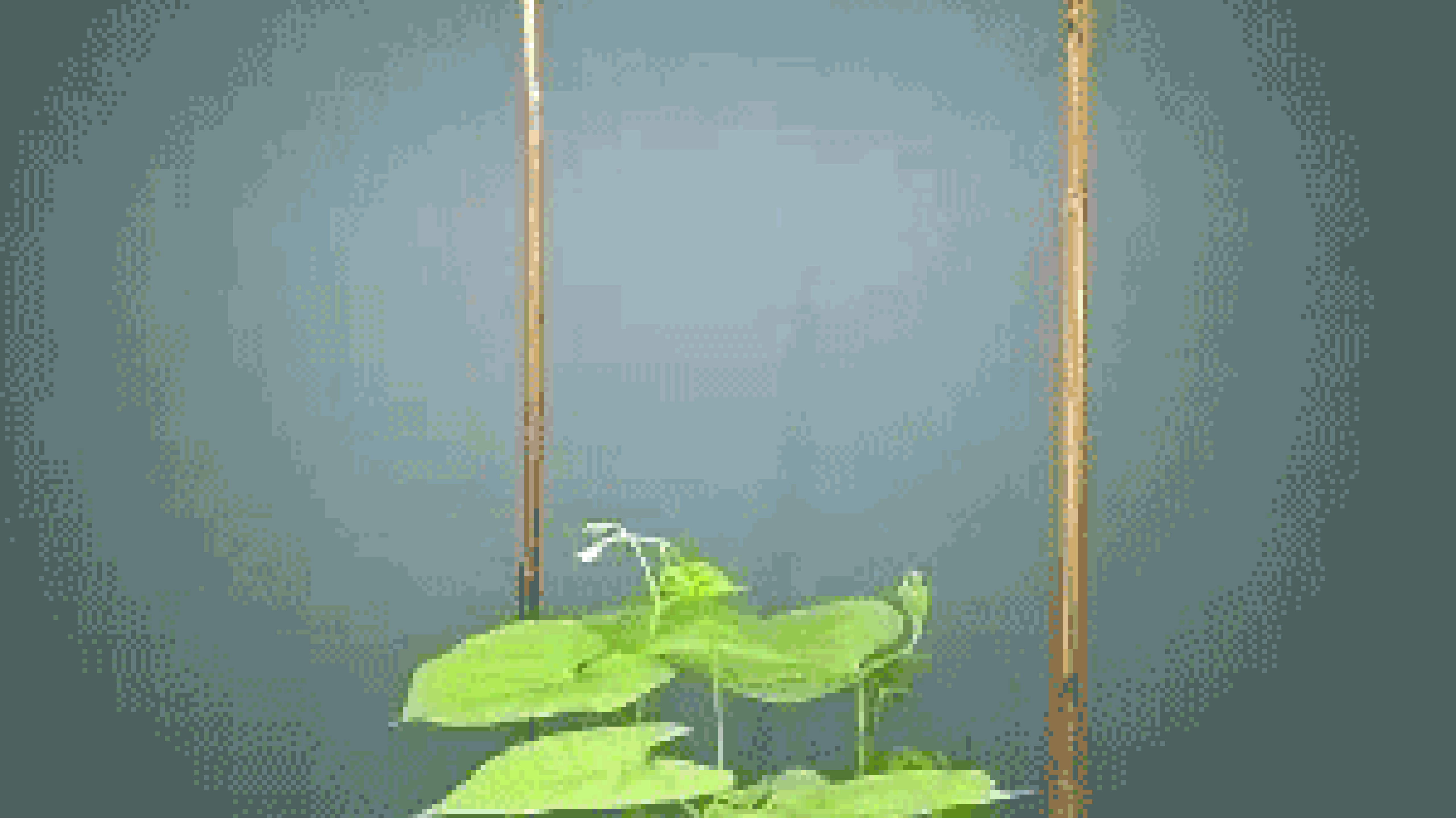
AWS OpsWorks

Node.js

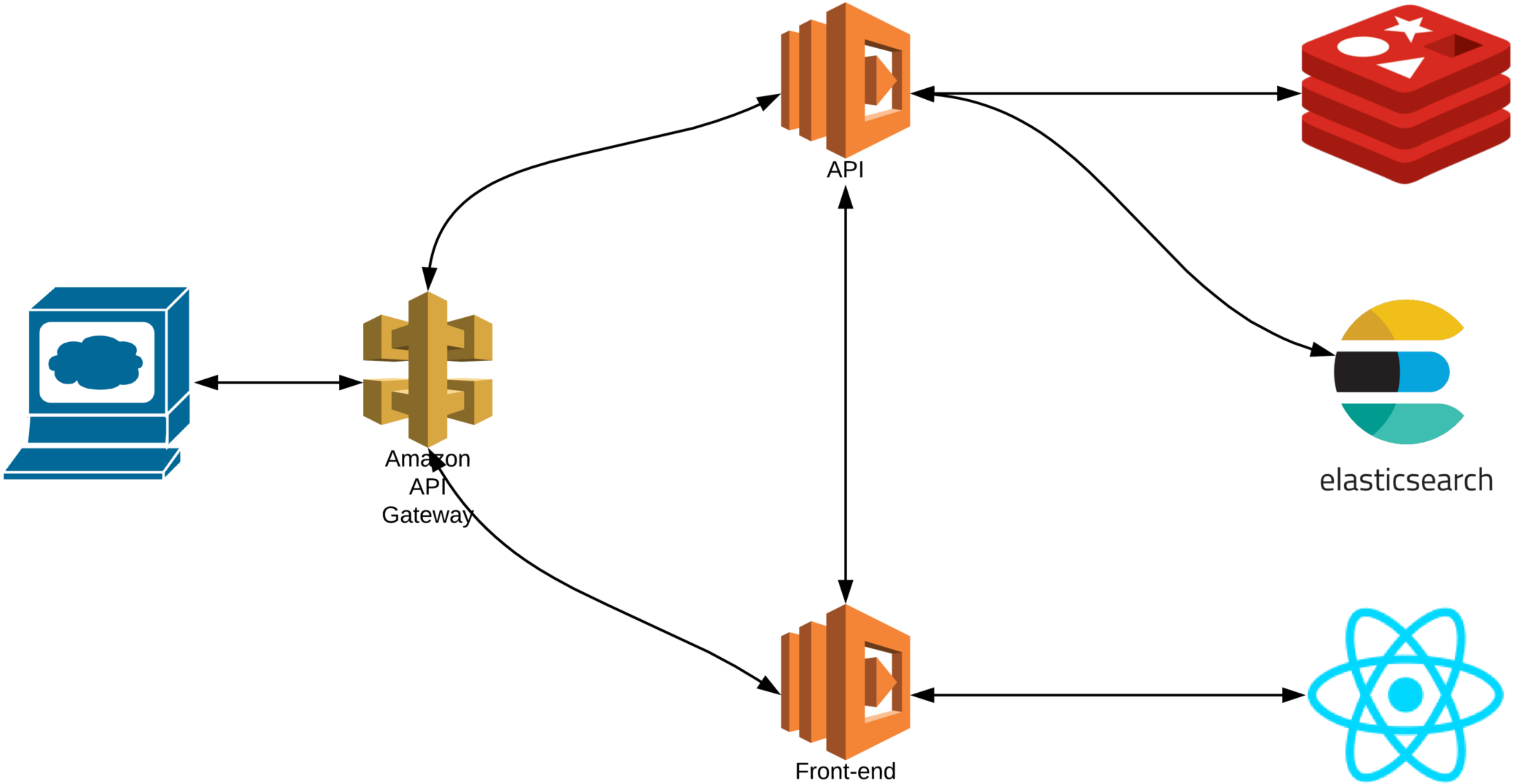


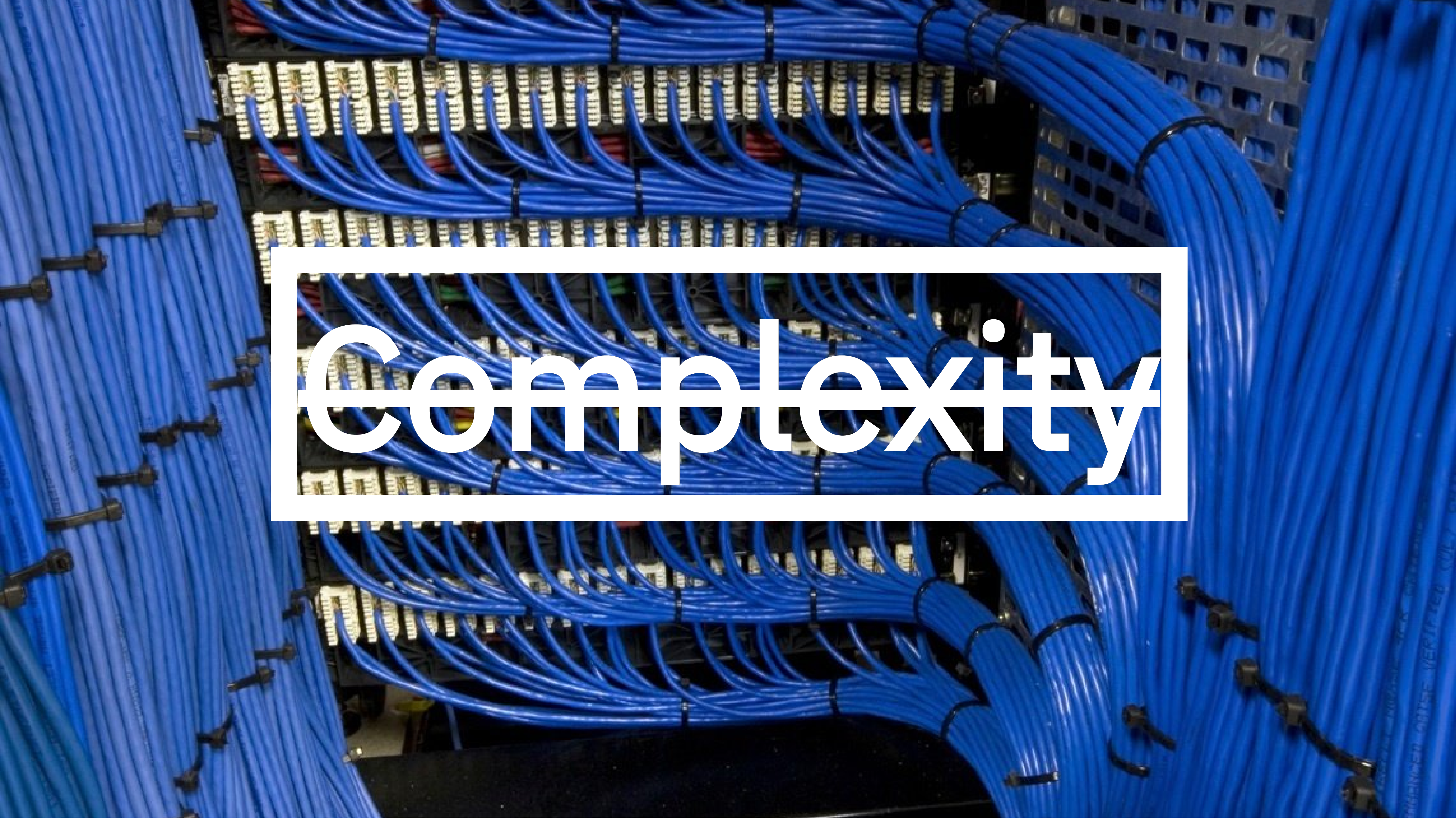
```
exports.handler = function({ body, request }, context, callback) {
  body.request = request;
  authorizeApiKey(request.key)
    .return(body)
    .then(decorateEvents)
    .then(putEvents)
    .return([null, buildResponse()])
    .spread(callback)
    .catch(callback);
}
```


Codebase Merged
Amazon API Gateway
AWS Lambda
Amazon Kinesis
AWS Lambda
elasticsearch



ROMPER


API
Amazon
API
Gateway
elasticsearch
Front-end



~~Complexity~~

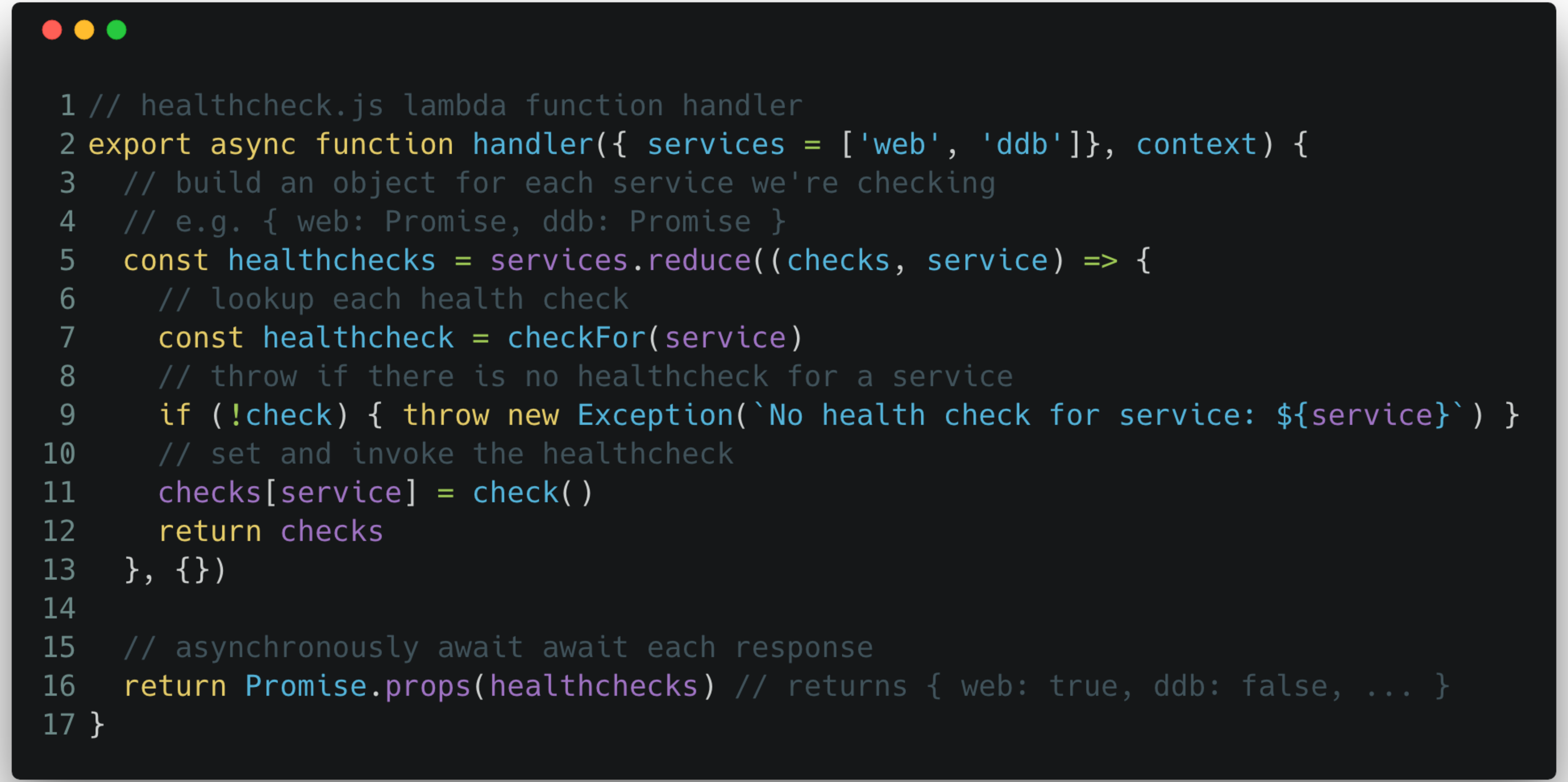
BUSTLE

TypeSet

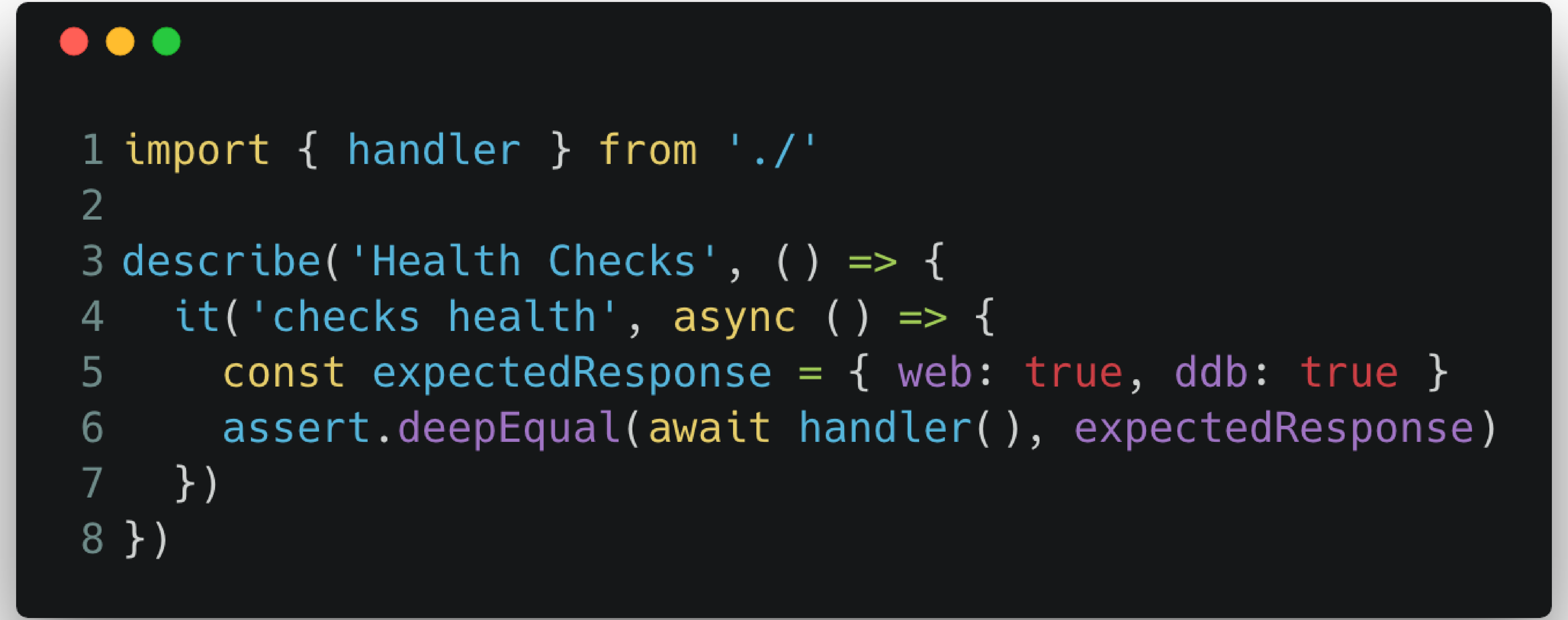
Testing

Portability

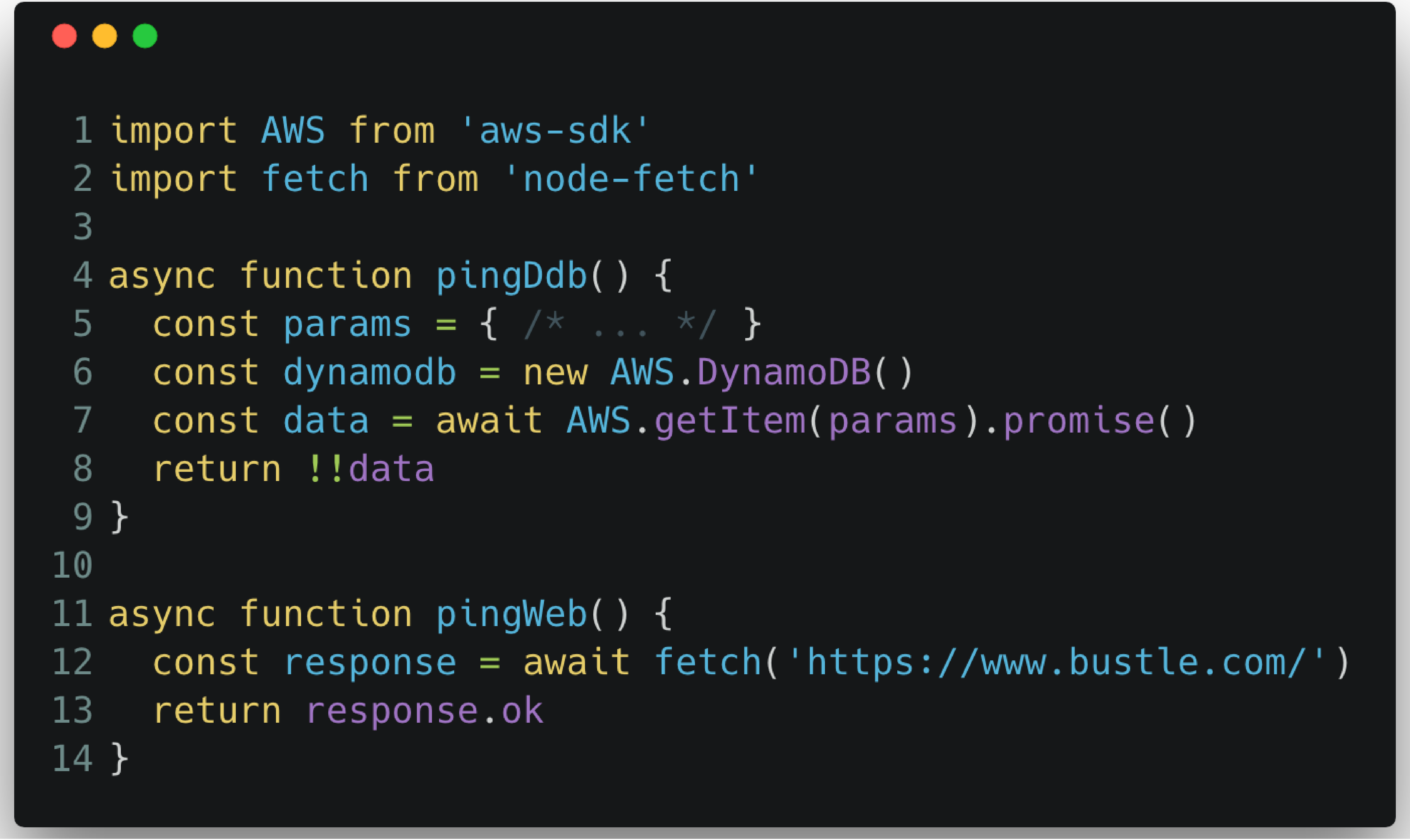
Context & Isolation



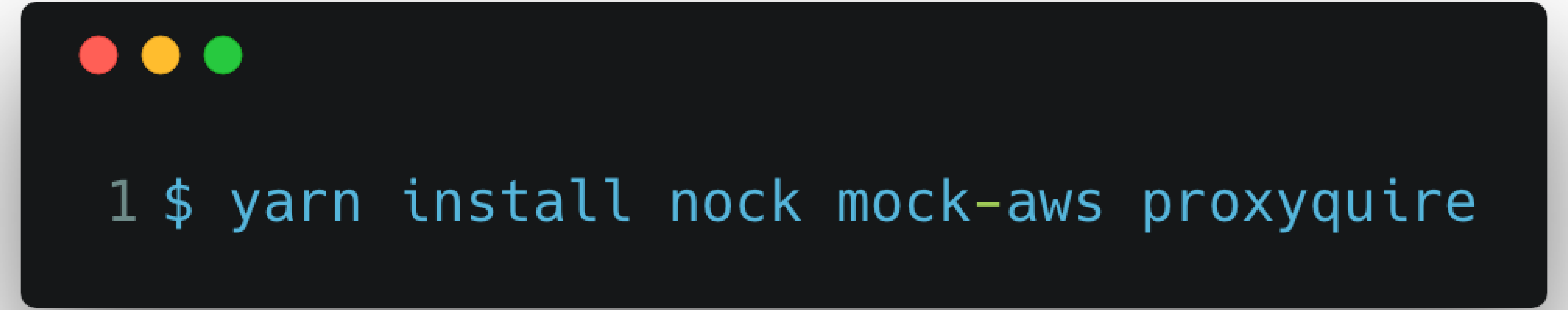
```javascript
// healthcheck.js lambda function handler
export async function handler({ services = ['web', 'ddb']}, context) {
  // build an object for each service we're checking
  // e.g. { web: Promise, ddb: Promise }
  const healthchecks = services.reduce((checks, service) => {
    // lookup each health check
    const healthcheck = checkFor(service)
    // throw if there is no healthcheck for a service
    if (!check) { throw new Exception(`No health check for service: ${service}`) }
    // set and invoke the healthcheck
    checks[service] = check()
    return checks
  }, {})

  // asynchronously await await each response
  return Promise.props(healthchecks) // returns { web: true, ddb: false, ... }
}
```



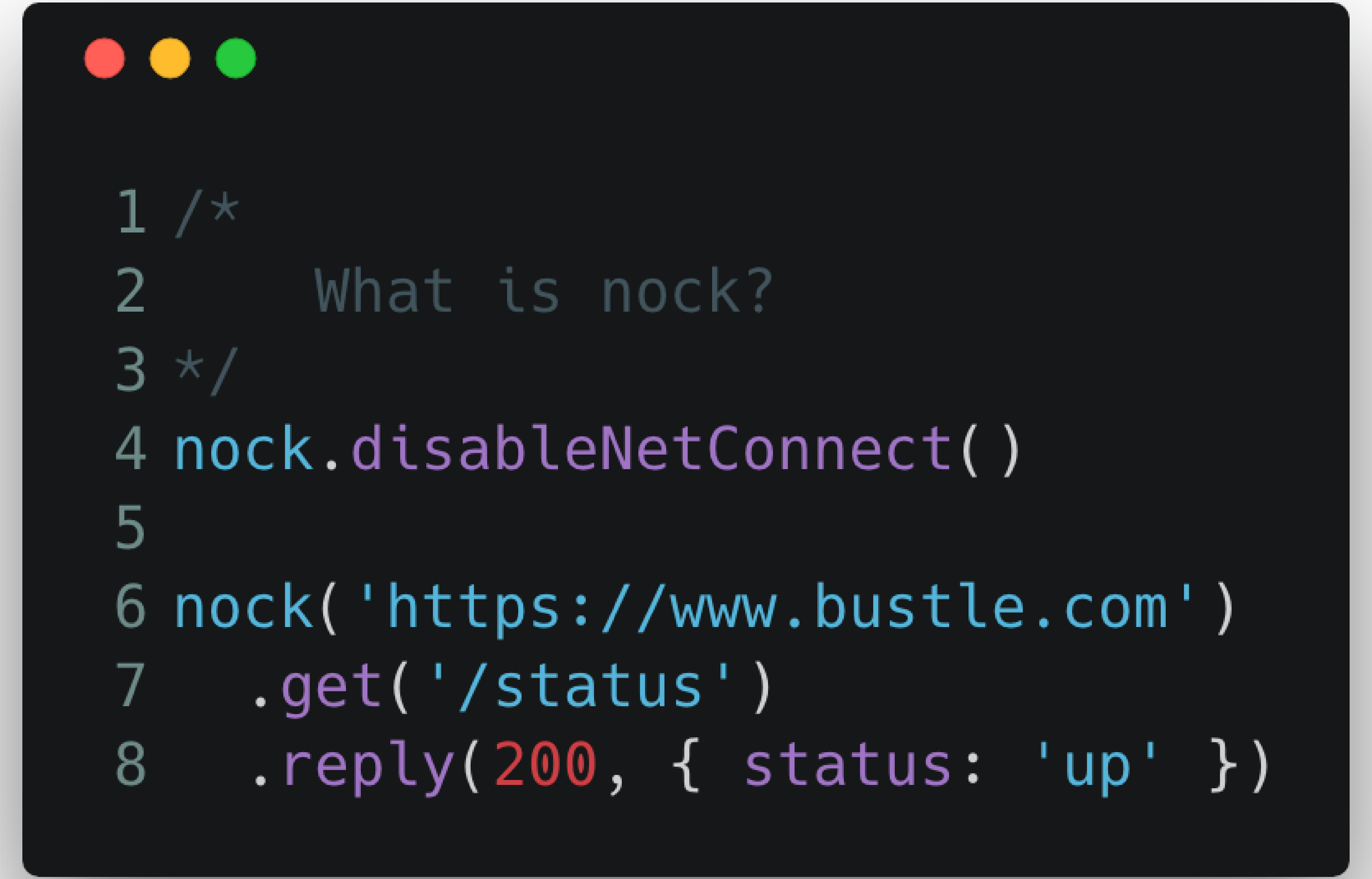
```javascript
import { handler } from './'

describe('Health Checks', () => {
  it('checks health', async () => {
    const expectedResponse = { web: true, ddb: true }
    assert.deepEqual(await handler(), expectedResponse)
  })
})
```



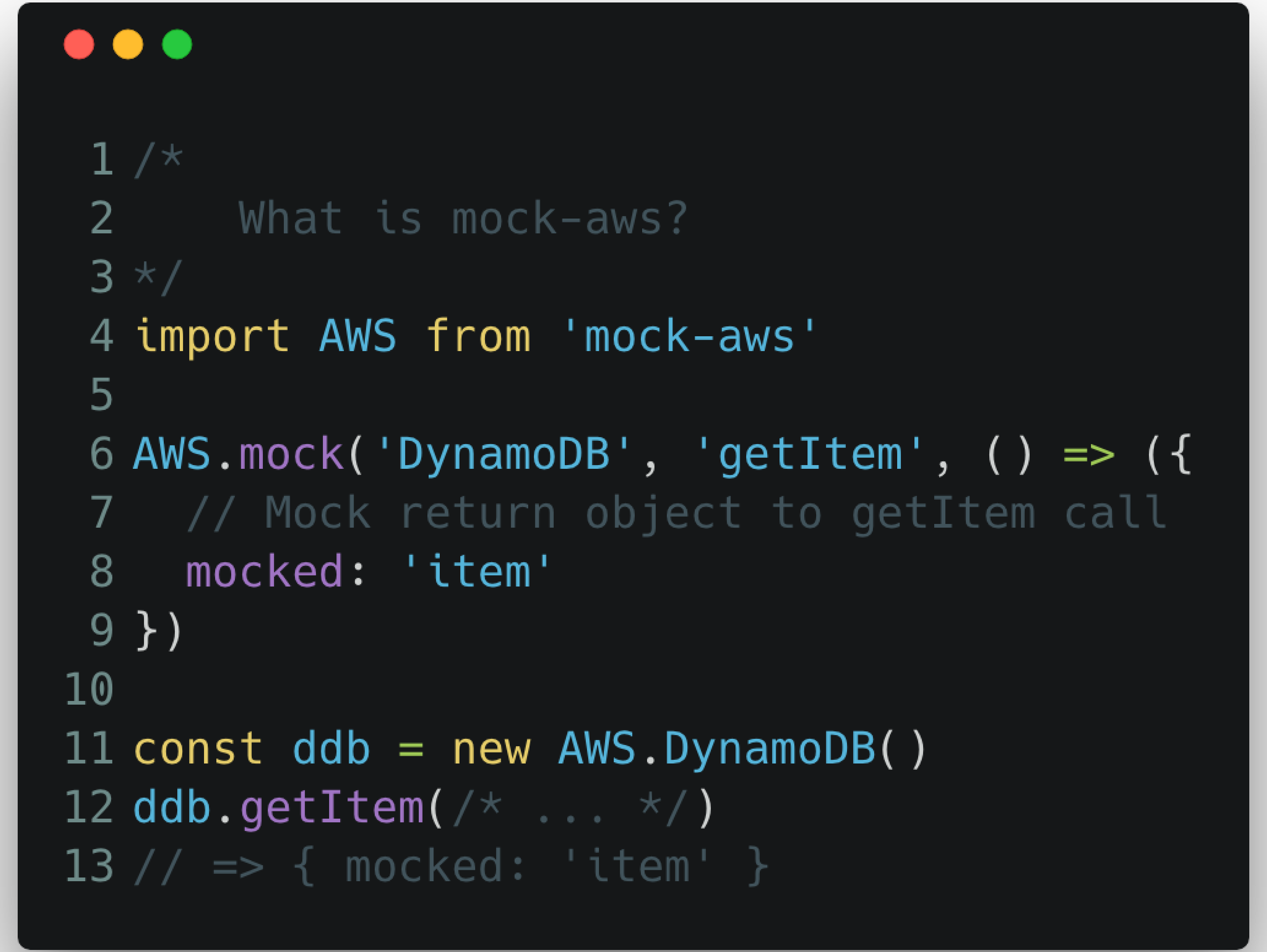
```javascript
import AWS from 'aws-sdk'
import fetch from 'node-fetch'

async function pingDdb() {
  const params = { /* ... */ }
  const dynamodb = new AWS.DynamoDB()
  const data = await AWS.getItem(params).promise()
  return !!data
}

async function pingWeb() {
  const response = await fetch('https://www.bustle.com/')
  return response.ok
}
```



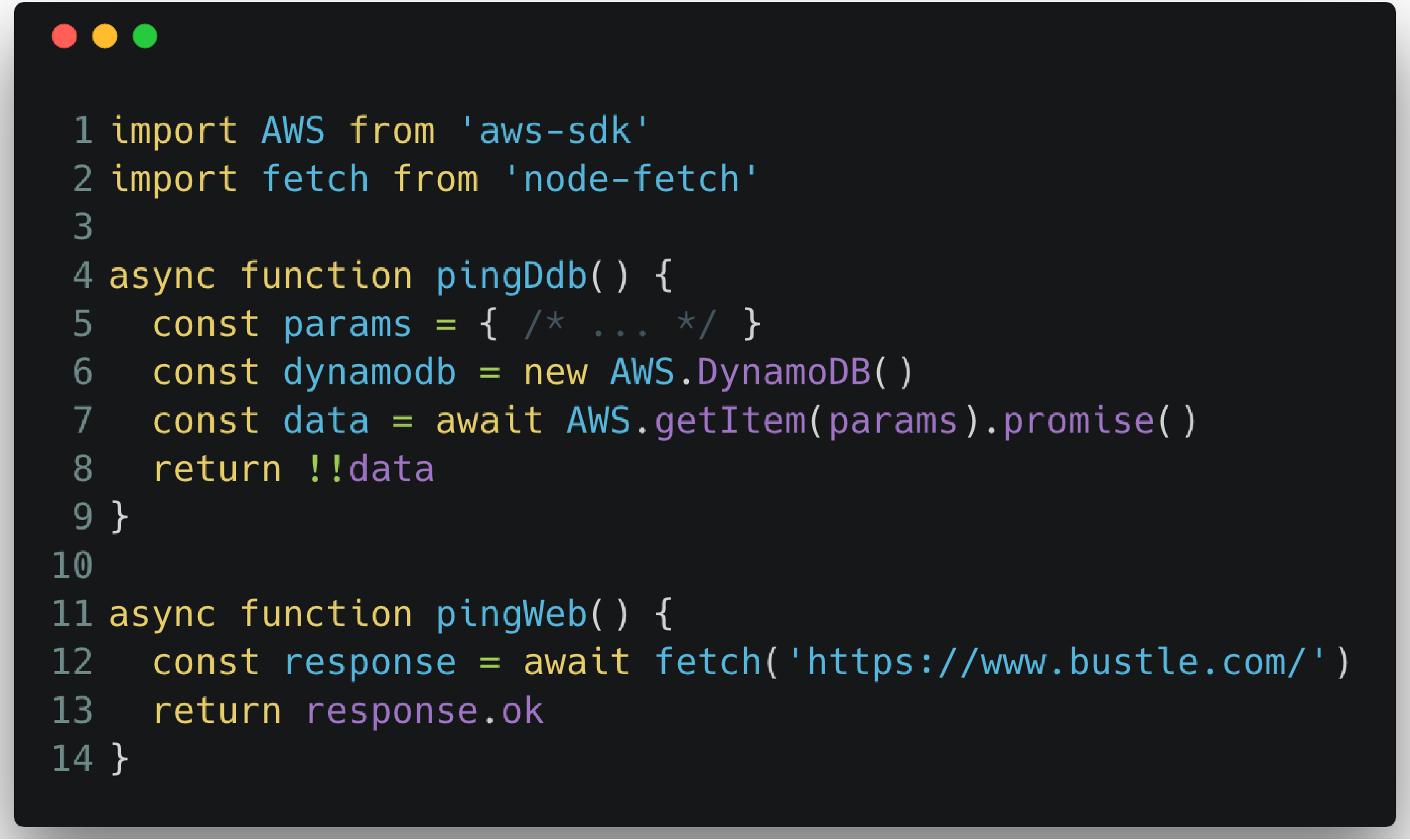
```
$ yarn install nock mock-aws proxyquire
```



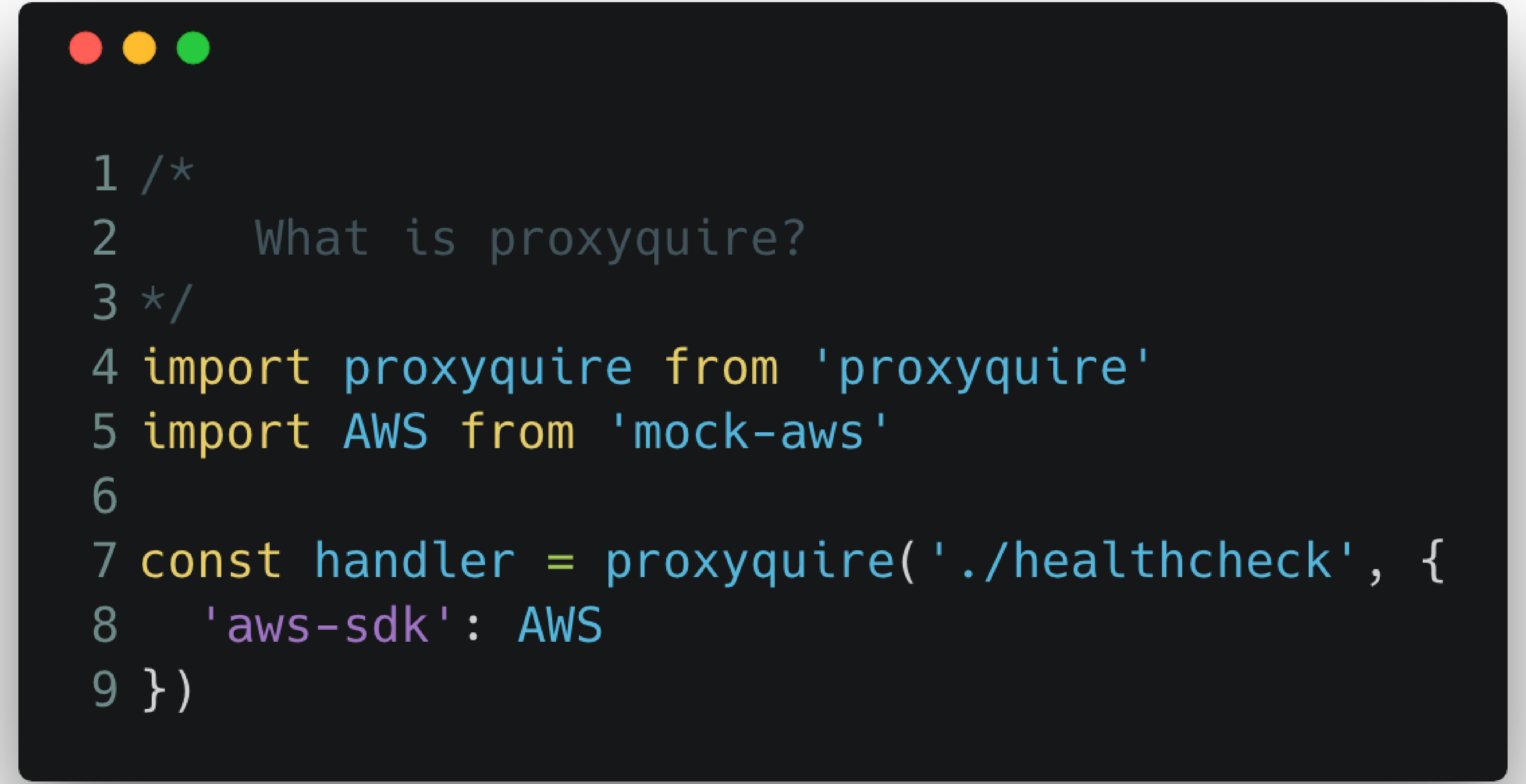
```
/*
    What is nock?
*/
nock.disableNetConnect()

nock('https://www.bustle.com')
  .get('/status')
  .reply(200, { status: 'up' })
```



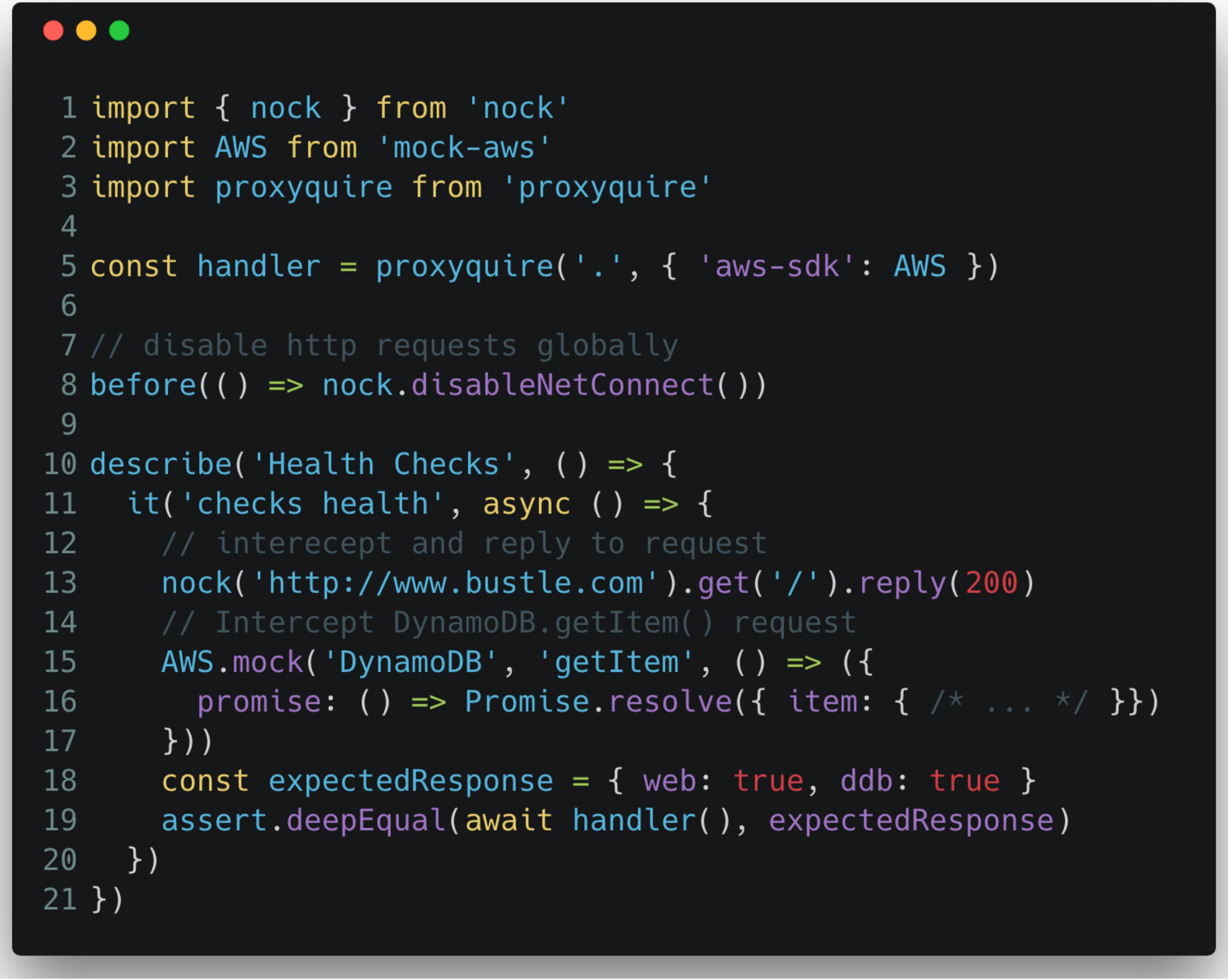
```
/*
    What is mock-aws?
*/
import AWS from 'mock-aws'

AWS.mock('DynamoDB', 'getItem', () => ({
  // Mock return object to getItem call
  mocked: 'item'
})

const ddb = new AWS.DynamoDB()
ddb.getItem(/* ... */)
// => { mocked: 'item' }
```



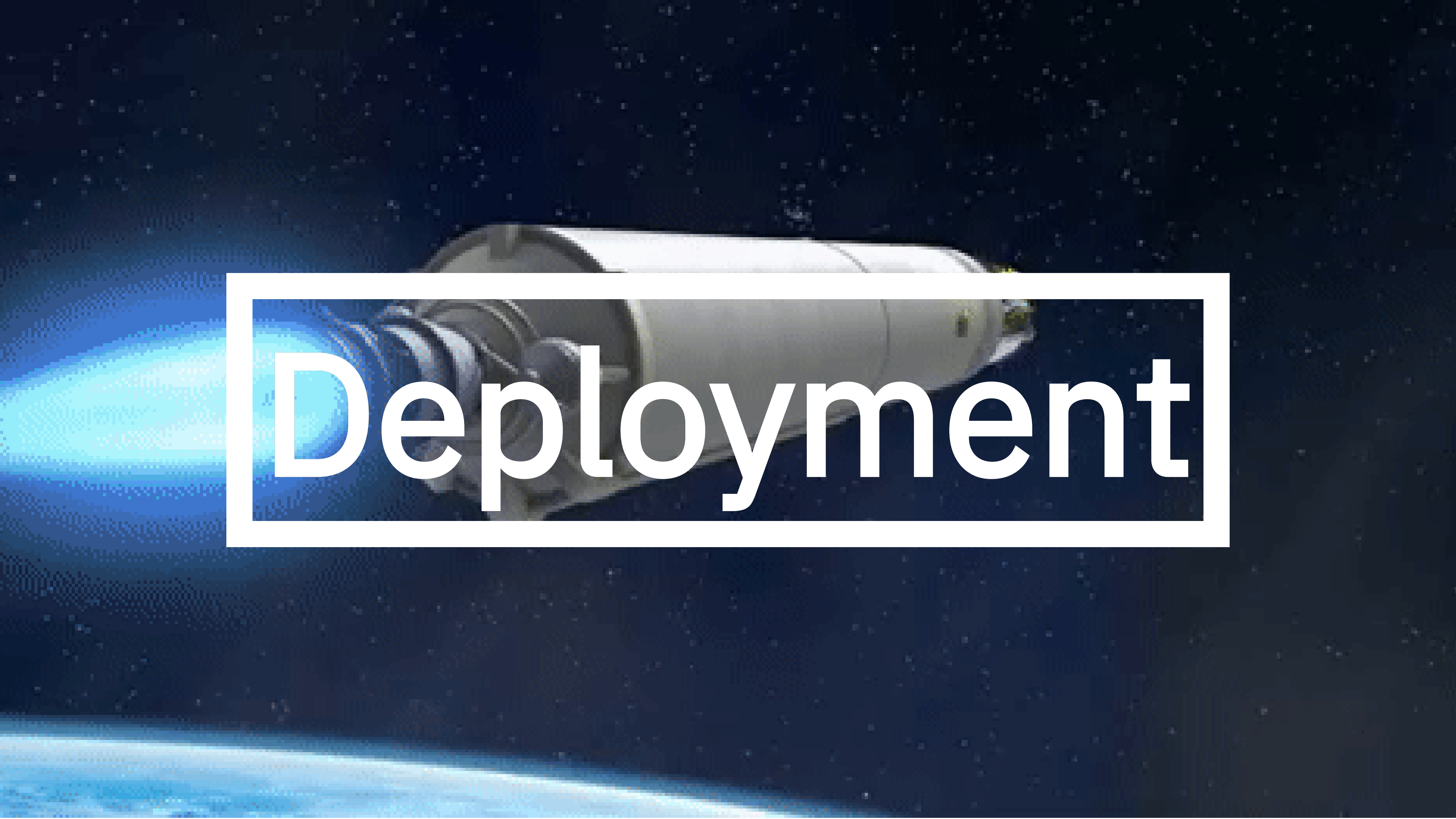
```
import AWS from 'aws-sdk'
import fetch from 'node-fetch'

async function pingDdb() {
  const params = { /* ... */ }
  const dynamodb = new AWS.DynamoDB()
  const data = await AWS.getItem(params).promise()
  return !!data
}

async function pingWeb() {
  const response = await fetch('https://www.bustle.com/')
  return response.ok
}
```



```
/*
    What is proxyquire?
*/
import proxyquire from 'proxyquire'
import AWS from 'mock-aws'

const handler = proxyquire('./healthcheck', {
  'aws-sdk': AWS
})
```

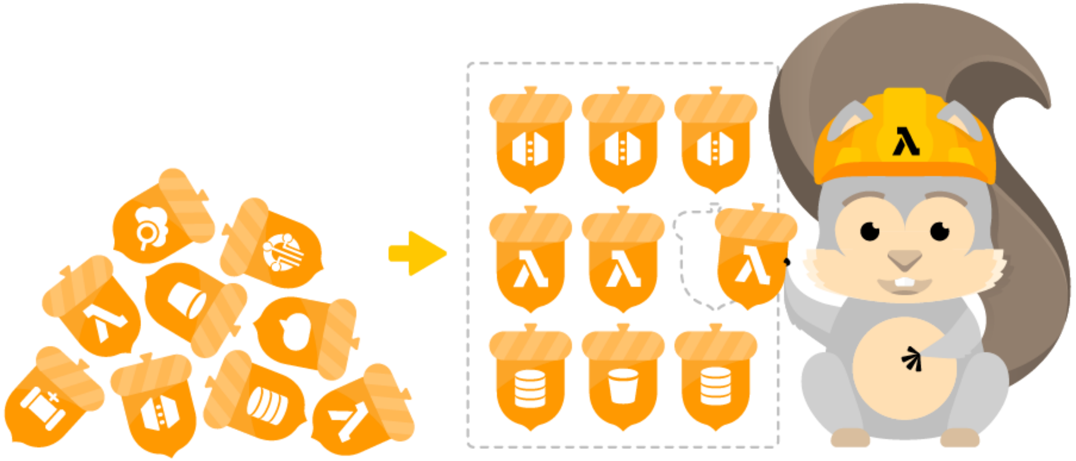


```
import { nock } from 'nock'
import AWS from 'mock-aws'
import proxyquire from 'proxyquire'

const handler = proxyquire('.', { 'aws-sdk': AWS })

// disable http requests globally
before(() => nock.disableNetConnect())

describe('Health Checks', () => {
  it('checks health', async () => {
    // interecept and reply to request
    nock('http://www.bustle.com').get('/').reply(200)
    // Intercept DynamoDB.getItem() request
    AWS.mock('DynamoDB', 'getItem', () => ({
      promise: () => Promise.resolve({ item: { /* ... */ }})
    }))
    const expectedResponse = { web: true, ddb: true }
    assert.deepEqual(await handler(), expectedResponse)
  })
})
```

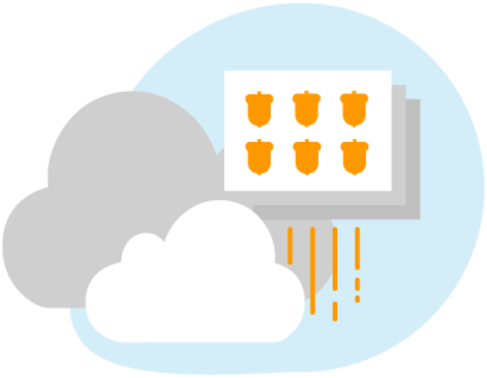


Deployment

MEET SAM.



USE SAM TO BUILD TEMPLATES THAT DEFINE YOUR SERVERLESS APPLICATIONS.

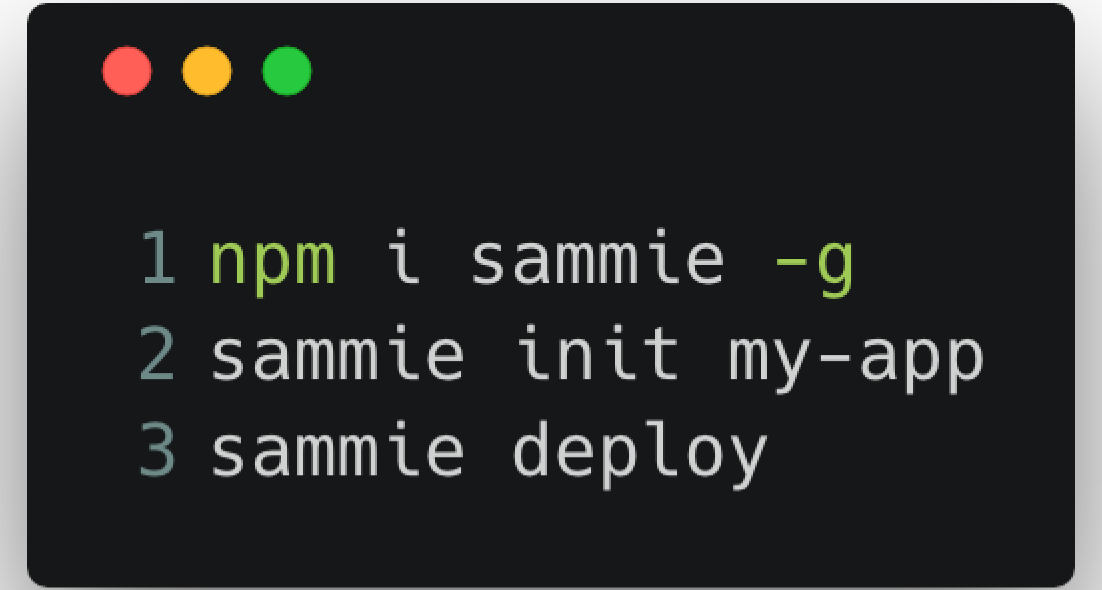


DEPLOY YOUR SAM TEMPLATE WITH AWS CLOUDFORMATION.

SAMMIE

Serverless Application Model Made Infinitely Easier

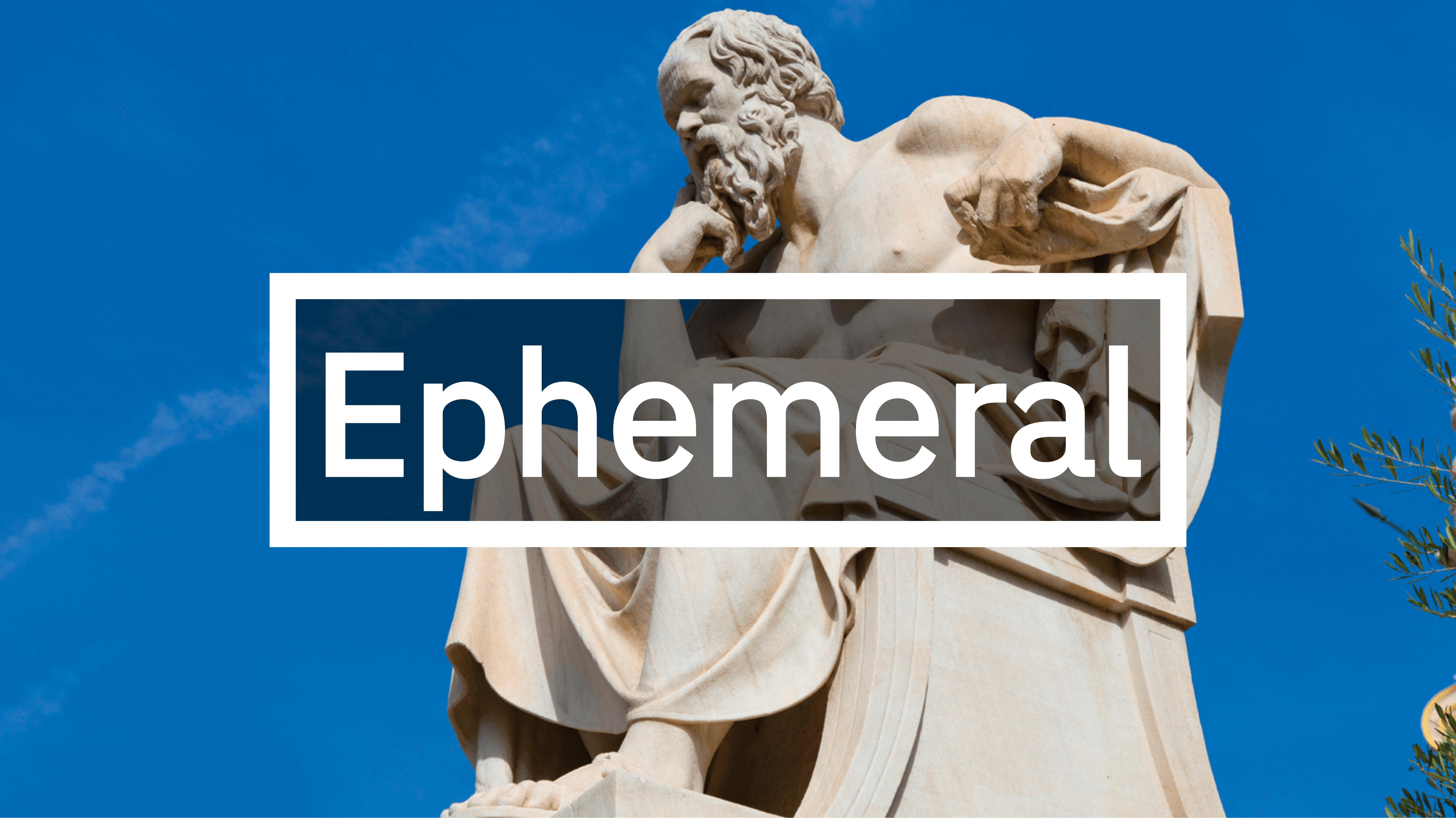
github.com/bustle/sammie



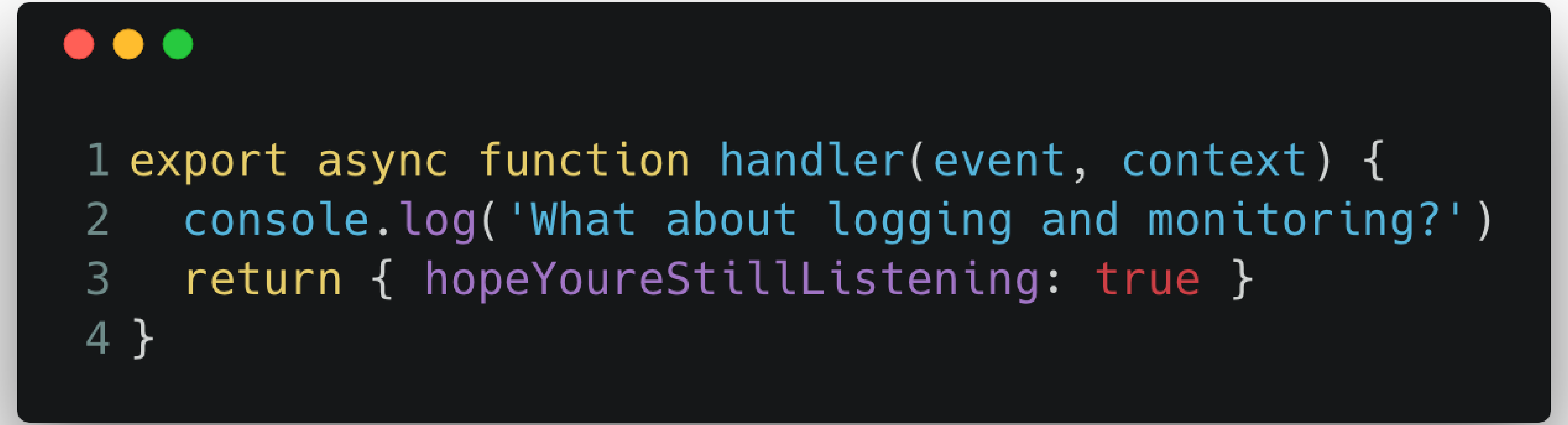
```
npm i sammie -g
sammie init my-app
sammie deploy
```

Logging

Convention

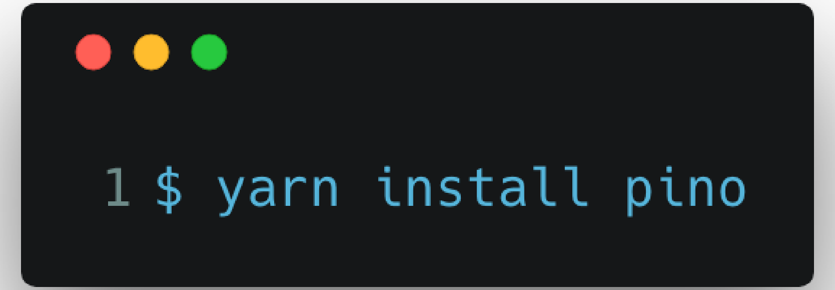


Ephemeral

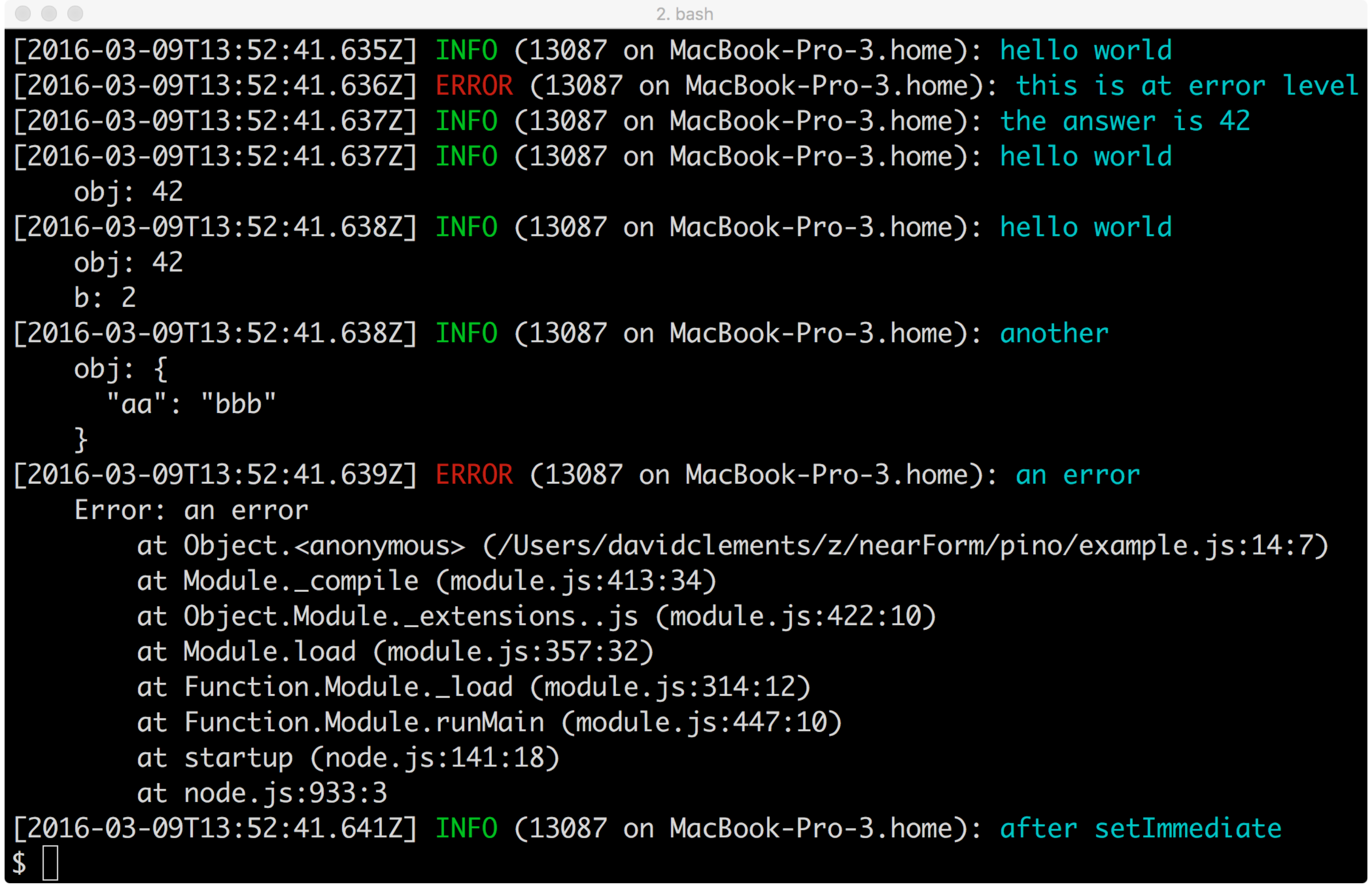


```
export async function handler(event, context) {
  console.log('What about logging and monitoring?')
  return { hopeYoureStillListening: true }
}
```

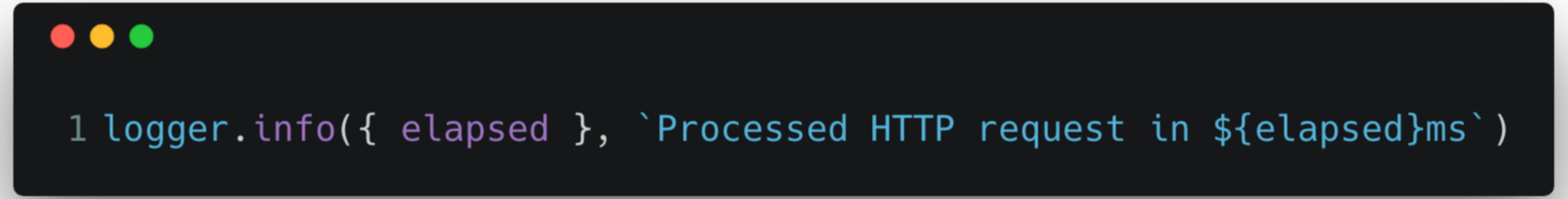
pino



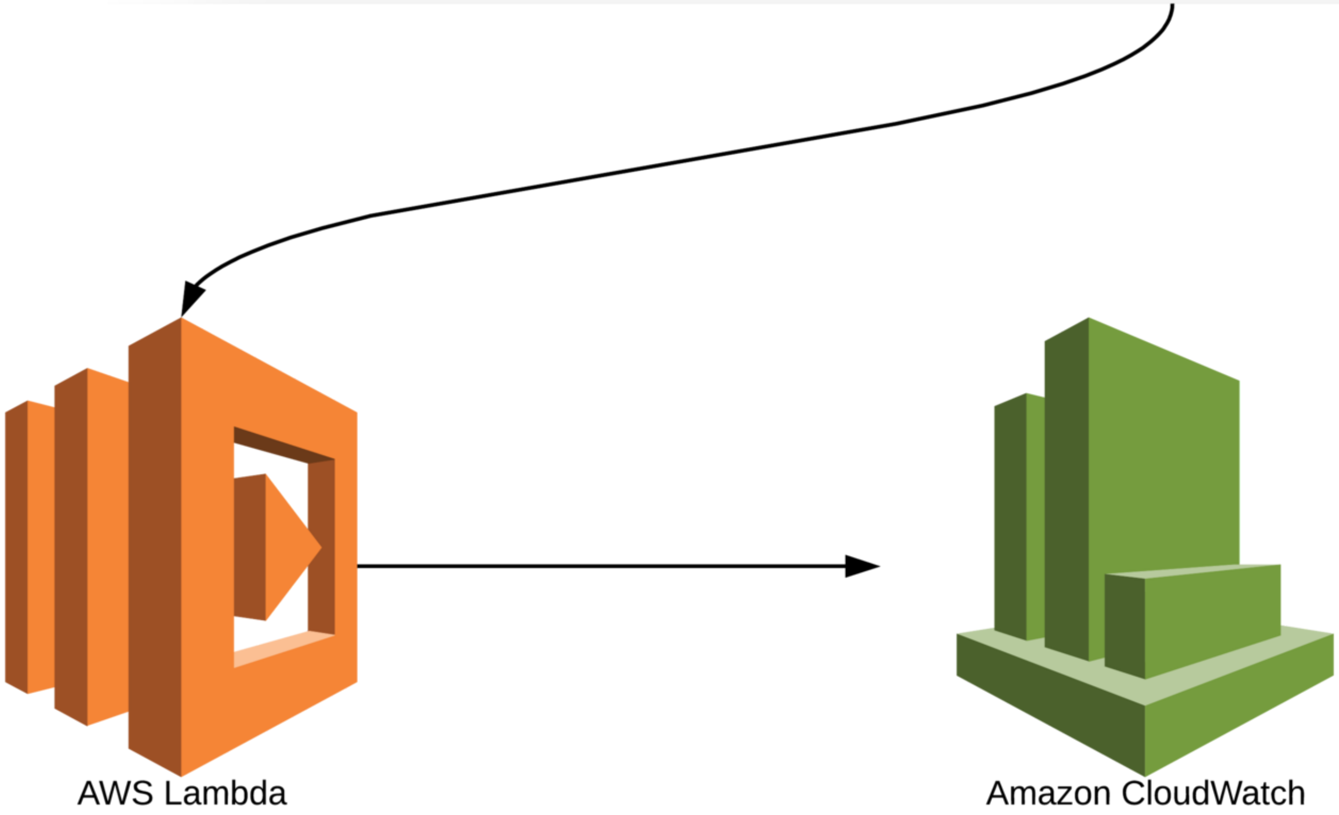
```
$ yarn install pino
```

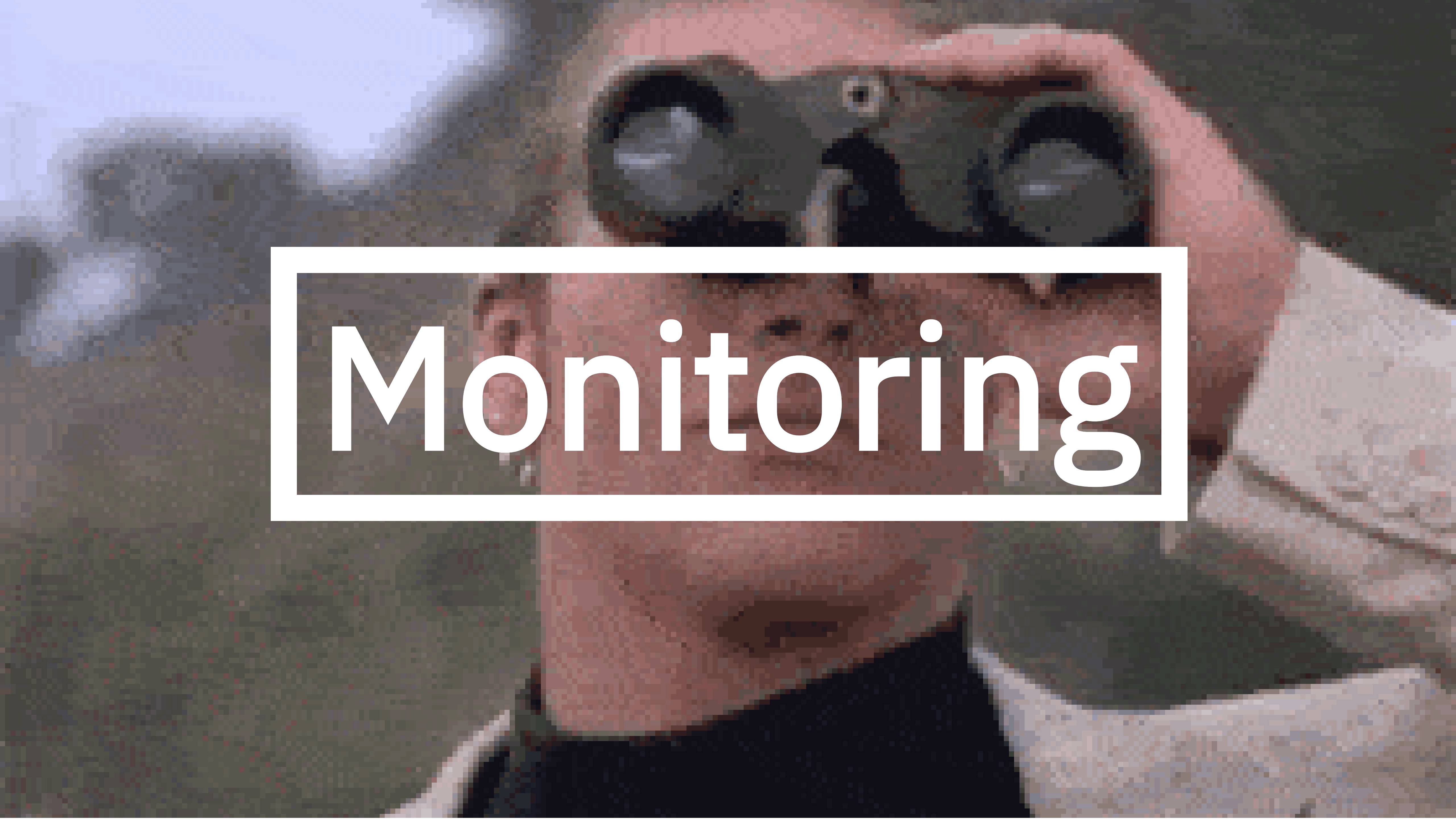


```
[2016-03-09T13:52:41.635Z] INFO (13087 on MacBook-Pro-3.home): hello world
[2016-03-09T13:52:41.636Z] ERROR (13087 on MacBook-Pro-3.home): this is at error level
[2016-03-09T13:52:41.637Z] INFO (13087 on MacBook-Pro-3.home): the answer is 42
[2016-03-09T13:52:41.637Z] INFO (13087 on MacBook-Pro-3.home): hello world
    obj: 42
[2016-03-09T13:52:41.638Z] INFO (13087 on MacBook-Pro-3.home): hello world
    obj: 42
    b: 2
[2016-03-09T13:52:41.638Z] INFO (13087 on MacBook-Pro-3.home): another
    obj: {
      "aa": "bbb"
    }
[2016-03-09T13:52:41.639Z] ERROR (13087 on MacBook-Pro-3.home): an error
    Error: an error
        at Object.<anonymous> (/Users/davidclements/z/nearForm/pino/example.js:14:7)
        at Module._compile (module.js:413:34)
        at Object.Module._extensions..js (module.js:422:10)
        at Module.load (module.js:357:32)
        at Function.Module._load (module.js:314:12)
        at Function.Module.runMain (module.js:447:10)
        at startup (node.js:141:18)
        at node.js:933:3
[2016-03-09T13:52:41.641Z] INFO (13087 on MacBook-Pro-3.home): after setImmediate
$ 
```

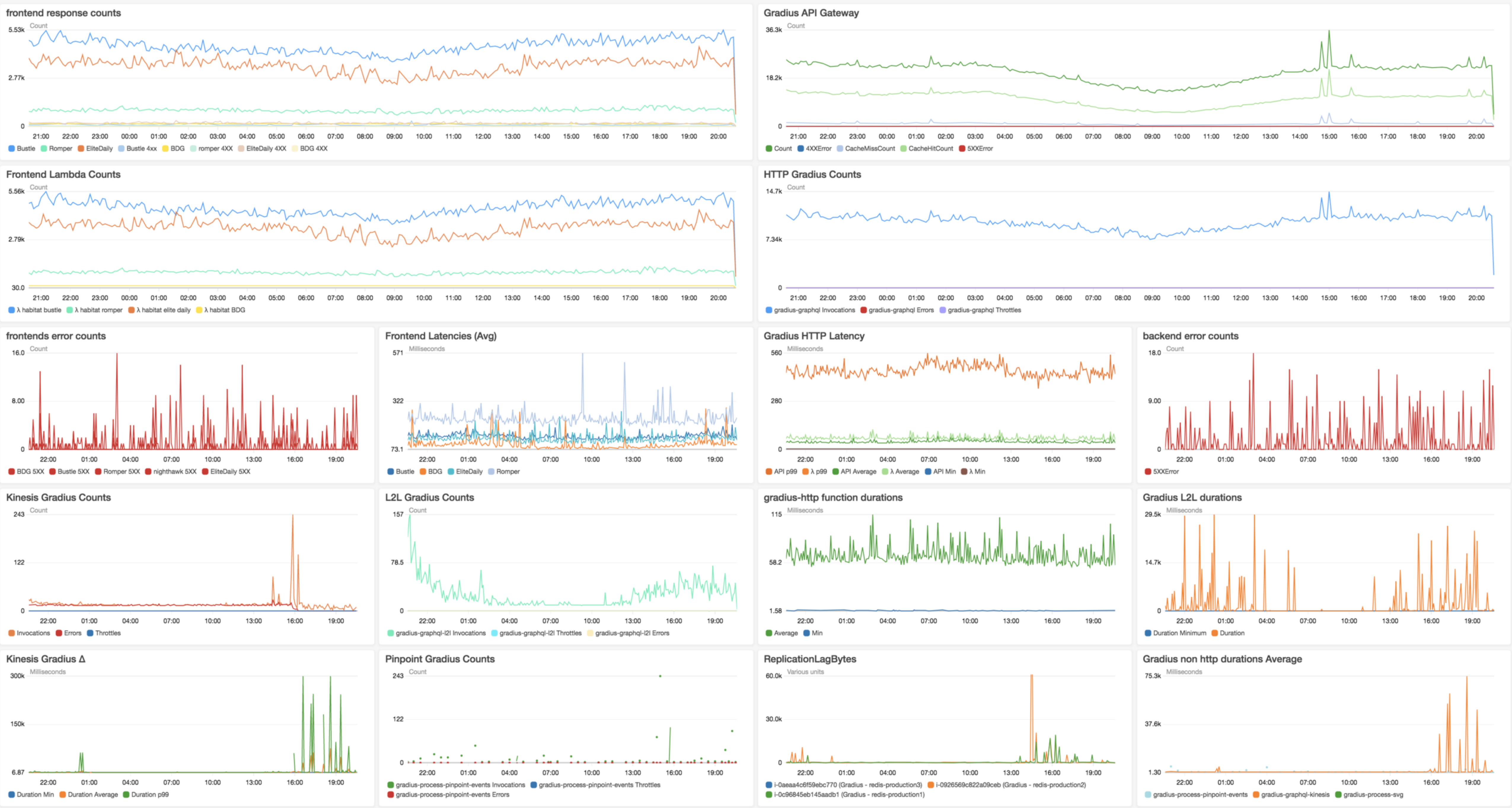


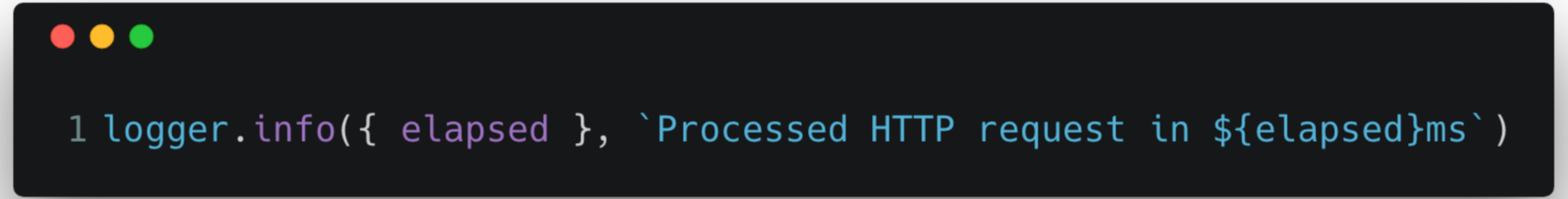
```
logger.info({ elapsed }, `Processed HTTP request in ${elapsed}ms`)
```



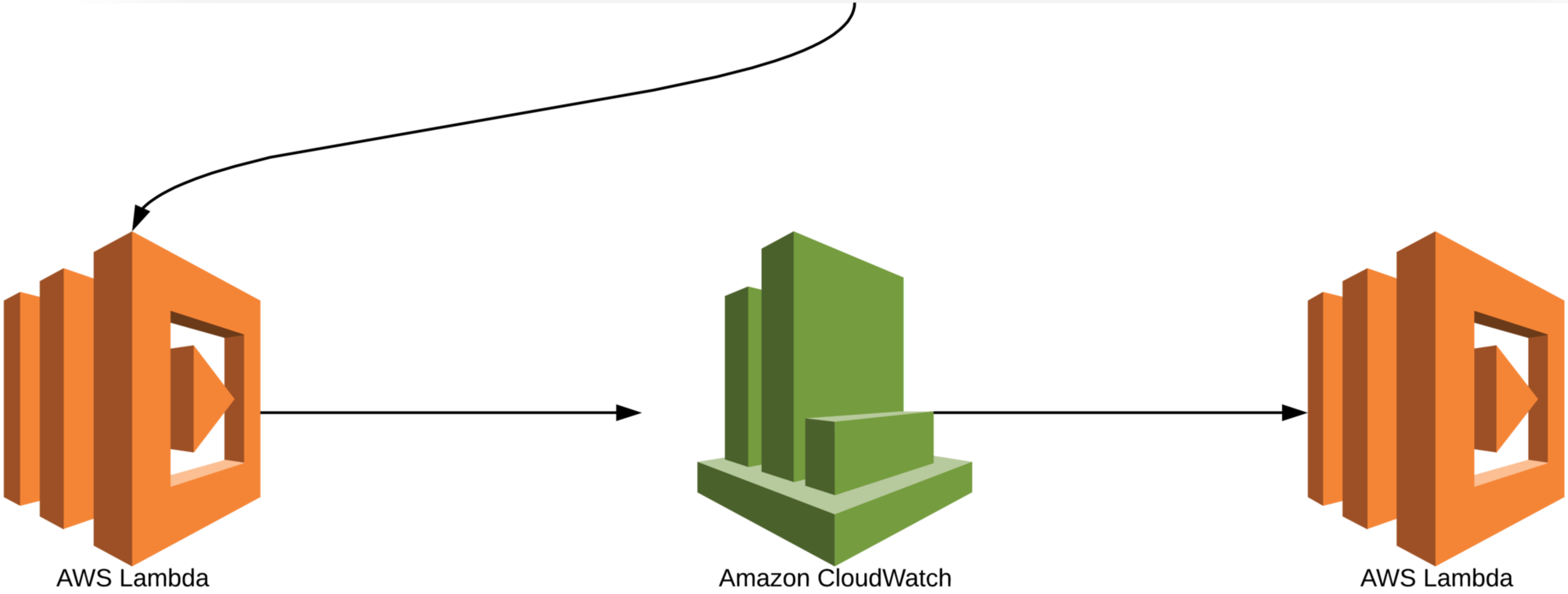


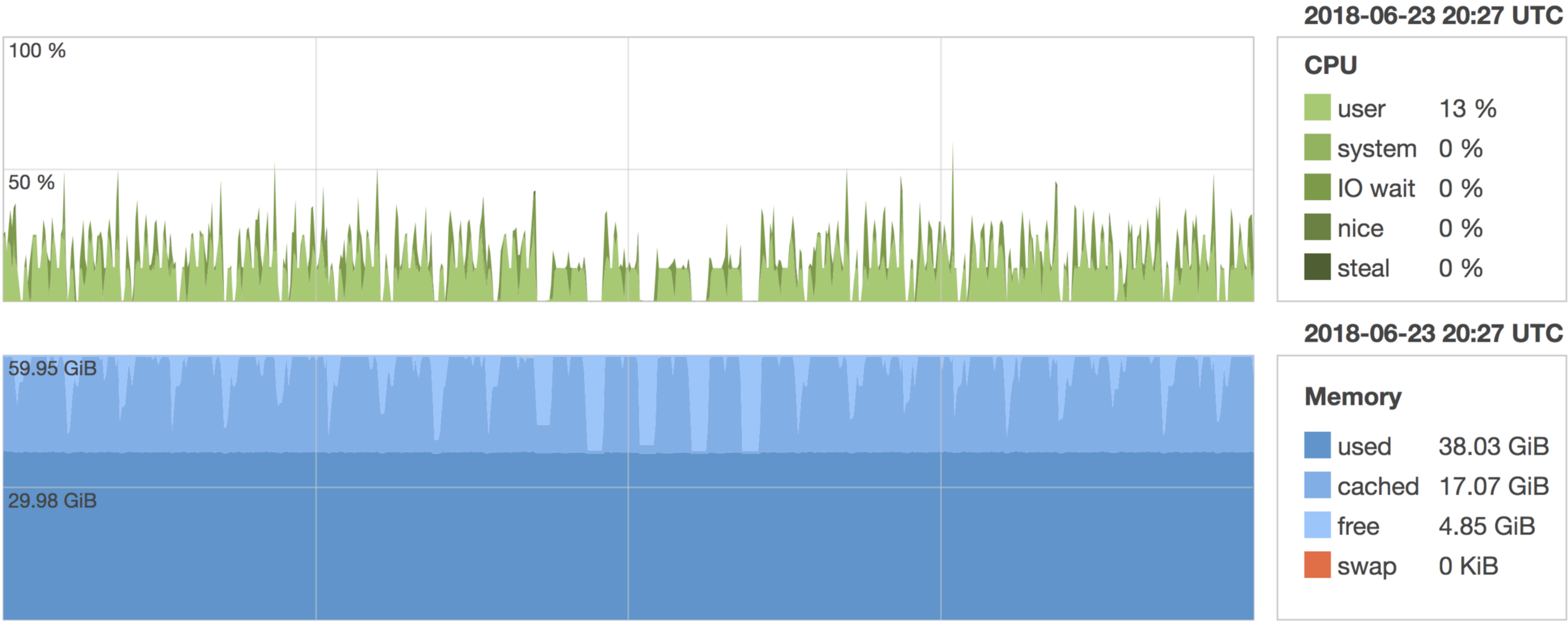
Monitoring


frontend response counts
Count
5.53k
2.77k
0
21:00 22:00 23:00 00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00 12:00 13:00 14:00 15:00 16:00 17:00 18:00 19:00 20:00
Bustle Romper EliteDaily Bustle 4xx BDG romper 4XX EliteDaily 4XX BDG 4XX
Gradius API Gateway
Count
36.3k
18.2k
0
21:00 22:00 23:00 00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00 12:00 13:00 14:00 15:00 16:00 17:00 18:00 19:00 20:00
Count 4XXError CacheMissCount CacheHitCount 5XXError
Frontend Lambda Counts
Count
5.56k
2.79k
30.0
21:00 22:00 23:00 00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00 12:00 13:00 14:00 15:00 16:00 17:00 18:00 19:00 20:00
λ habitat bustle λ habitat romper λ habitat elite daily λ habitat BDG
HTTP Gradius Counts
Count
14.7k
7.34k
0
21:00 22:00 23:00 00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00 12:00 13:00 14:00 15:00 16:00 17:00 18:00 19:00 20:00
gradius-graphql Invocations gradius-graphql Errors gradius-graphql Throttles
frontends error counts
Count
16.0
8.00
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
BDG 5XX Bustle 5XX Romper 5XX nighthawk 5XX EliteDaily 5XX
Frontend Latencies (Avg)
Milliseconds
571
322
73.1
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
Bustle BDG EliteDaily Romper
Gradius HTTP Latency
Milliseconds
560
280
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
API p99 λ p99 API Average λ Average API Min λ Min
backend error counts
Count
18.0
9.00
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
5XXError
Kinesis Gradius Counts
Count
243
122
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
Invocations Errors Throttles
L2L Gradius Counts
Count
157
78.5
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
gradius-graphql-l2l Invocations gradius-graphql-l2l Throttles gradius-graphql-l2l Errors
gradius-http function durations
Milliseconds
115
58.2
1.58
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
Average Min
Gradius L2L durations
Milliseconds
29.5k
14.7k
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
Duration Minimum Duration
Kinesis Gradius Δ
Milliseconds
300k
150k
6.87
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
Duration Min Duration Average Duration p99
Pinpoint Gradius Counts
Count
243
122
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
gradius-process-pinpoint-events Invocations gradius-process-pinpoint-events Throttles
gradius-process-pinpoint-events Errors
ReplicationLagBytes
Various units
60.0k
30.0k
0
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
i-0aeaa4c6f59ebc770 (Gradius - redis-production3) i-0926569c822a09ceb (Gradius - redis-production2)
i-0c96845eb145aadb1 (Gradius - redis-production1)
Gradius non http durations Average
Milliseconds
75.3k
37.6k
1.30
22:00 01:00 04:00 07:00 10:00 13:00 16:00 19:00
gradius-process-pinpoint-events gradius-graphql-kinesis gradius-process-svg



```
logger.info({ elapsed }, `Processed HTTP request in ${elapsed}ms`)
```





2018-06-23 20:27 UTC

CPU

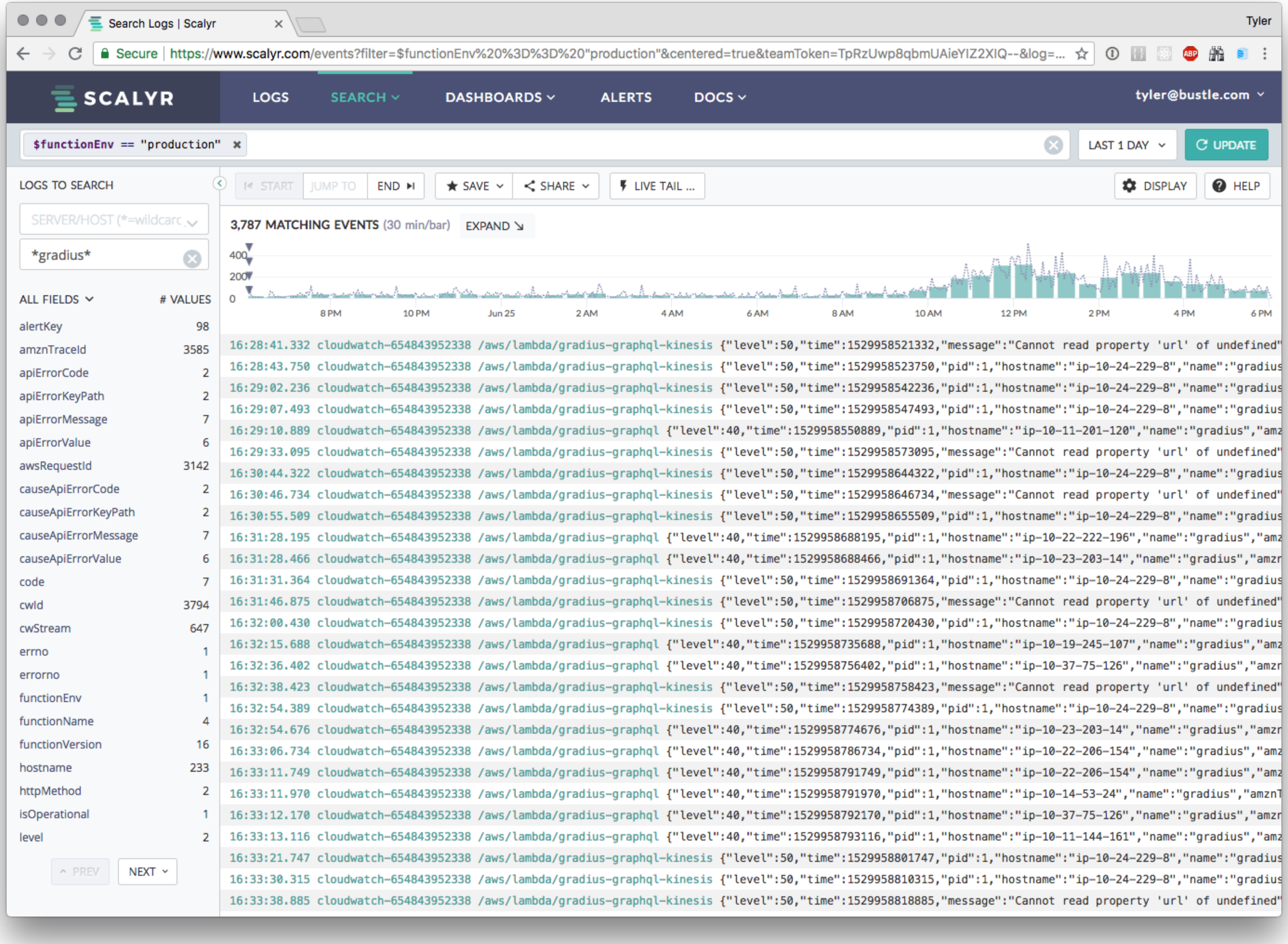
🟩	user	13 %
🟩	system	0 %
🟩	IO wait	0 %
🟩	nice	0 %
⬛	steal	0 %

2018-06-23 20:27 UTC

Memory

🟦	used	38.03 GiB
🟦	cached	17.07 GiB
🟦	free	4.85 GiB
🟧	swap	0 KiB

SCALYR



`$functionEnv == "production"`

LOGS TO SEARCH

SERVER/HOST (*=wildcard)

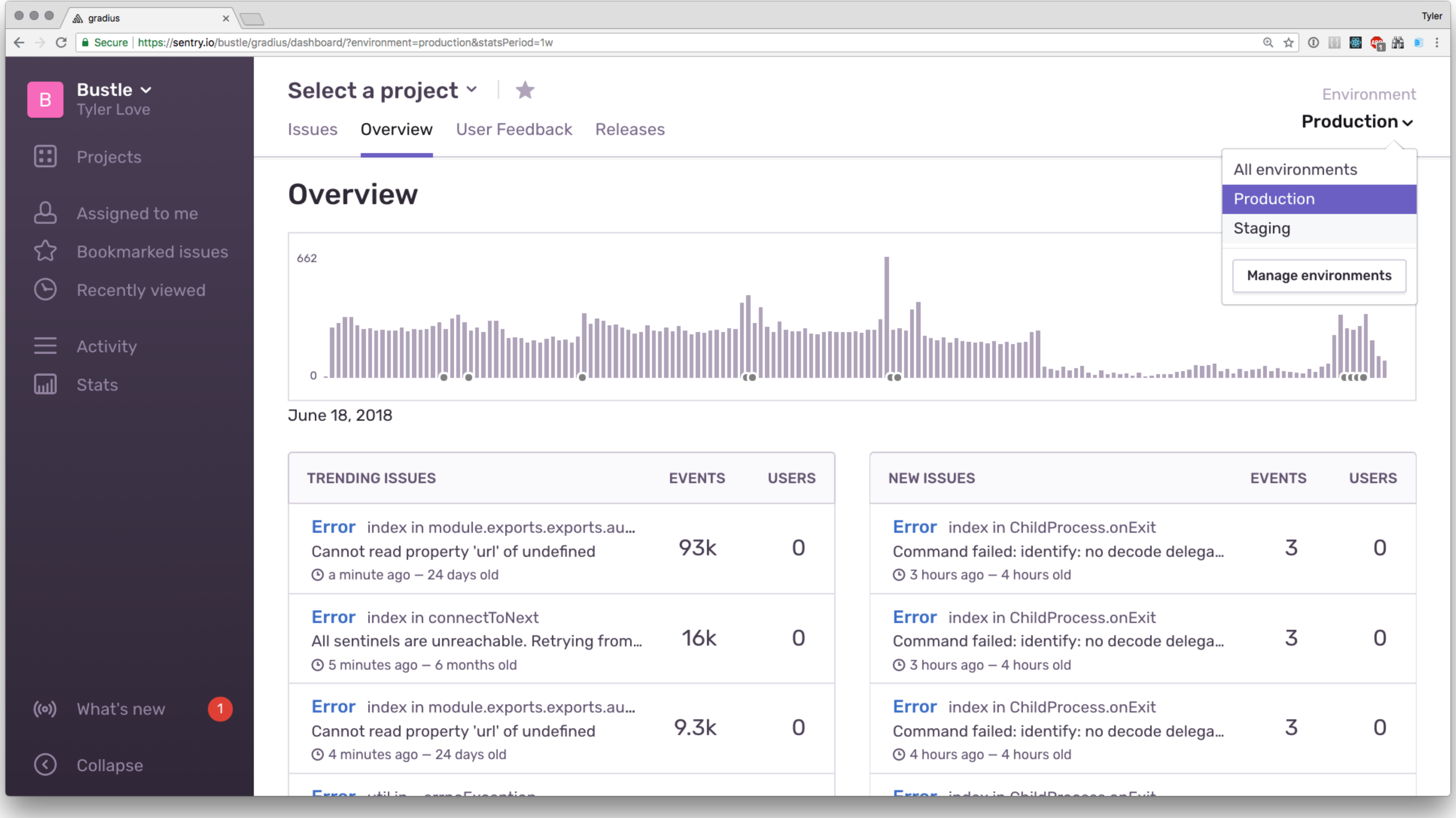
gradius

3,787 MATCHING EVENTS (30 min/bar)

ALL FIELDS	# VALUES
alertKey	98
amznTraceId	3585
apiErrorCode	2
apiErrorKeyPath	2
apiErrorMessage	7
apiErrorValue	6
awsRequestId	3142
causeApiErrorCode	2
causeApiErrorKeyPath	2
causeApiErrorMessage	7
causeApiErrorValue	6
code	7
cwId	3794
cwStream	647
errno	1
errorno	1
functionEnv	1
functionName	4
functionVersion	16
hostname	233
httpMethod	2
isOperational	1
level	2

```
16:28:41.332 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958521332,"message":"Cannot read property 'url' of undefined"
16:28:43.750 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958523750,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:29:02.236 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958542236,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:29:07.493 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958547493,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:29:10.889 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958550889,"pid":1,"hostname":"ip-10-11-201-120","name":"gradius","amz
16:29:33.095 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958573095,"message":"Cannot read property 'url' of undefined"
16:30:44.322 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958644322,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:30:46.734 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958646734,"message":"Cannot read property 'url' of undefined"
16:30:55.509 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958655509,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:31:28.195 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958688195,"pid":1,"hostname":"ip-10-22-222-196","name":"gradius","amz
16:31:28.466 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958688466,"pid":1,"hostname":"ip-10-23-203-14","name":"gradius","amzr
16:31:31.364 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958691364,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:31:46.875 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958706875,"message":"Cannot read property 'url' of undefined"
16:32:00.430 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958720430,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:32:15.688 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958735688,"pid":1,"hostname":"ip-10-19-245-107","name":"gradius","amz
16:32:36.402 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958756402,"pid":1,"hostname":"ip-10-37-75-126","name":"gradius","amzr
16:32:38.423 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958758423,"message":"Cannot read property 'url' of undefined"
16:32:54.389 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958774389,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:32:54.676 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958774676,"pid":1,"hostname":"ip-10-23-203-14","name":"gradius","amzr
16:33:06.734 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958786734,"pid":1,"hostname":"ip-10-22-206-154","name":"gradius","amz
16:33:11.749 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958791749,"pid":1,"hostname":"ip-10-22-206-154","name":"gradius","amz
16:33:11.970 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958791970,"pid":1,"hostname":"ip-10-14-53-24","name":"gradius","amznT
16:33:12.170 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958792170,"pid":1,"hostname":"ip-10-37-75-126","name":"gradius","amzr
16:33:13.116 cloudwatch-654843952338 /aws/lambda/gradius-graphql {"level":40,"time":1529958793116,"pid":1,"hostname":"ip-10-11-144-161","name":"gradius","amz
16:33:21.747 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958801747,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:33:30.315 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958810315,"pid":1,"hostname":"ip-10-24-229-8","name":"gradius
16:33:38.885 cloudwatch-654843952338 /aws/lambda/gradius-graphql-kinesis {"level":50,"time":1529958818885,"message":"Cannot read property 'url' of undefined"
```

SENTRY



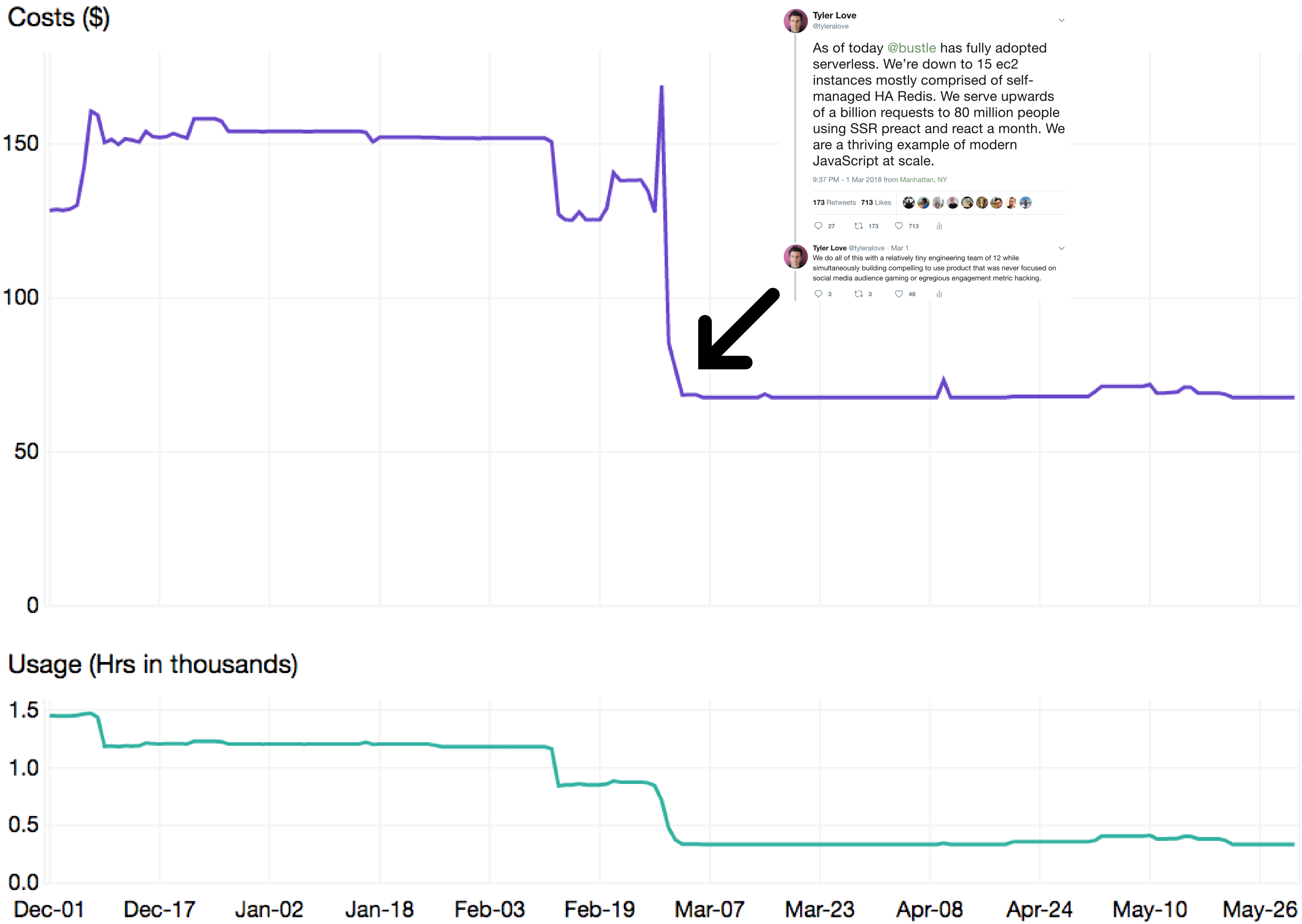
Overview

June 18, 2018

TRENDING ISSUES	EVENTS	USERS
Error index in module.exports.exports.au... Cannot read property 'url' of undefined a minute ago — 24 days old	93k	0
Error index in connectToNext All sentinels are unreachable. Retrying from... 5 minutes ago — 6 months old	16k	0
Error index in module.exports.exports.au... Cannot read property 'url' of undefined 4 minutes ago — 24 days old	9.3k	0
Error util in _errnoException		

NEW ISSUES	EVENTS	USERS
Error index in ChildProcess.onExit Command failed: identify: no decode delega... 3 hours ago — 4 hours old	3	0
Error index in ChildProcess.onExit Command failed: identify: no decode delega... 3 hours ago — 4 hours old	3	0
Error index in ChildProcess.onExit Command failed: identify: no decode delega... 4 hours ago — 4 hours old	3	0
Error index in ChildProcess.onExit		

Cost



Tyler Love
@tyleralove

As of today @bustle has fully adopted serverless. We're down to 15 ec2 instances mostly comprised of self-managed HA Redis. We serve upwards of a billion requests to 80 million people using SSR preact and react a month. We are a thriving example of modern JavaScript at scale.

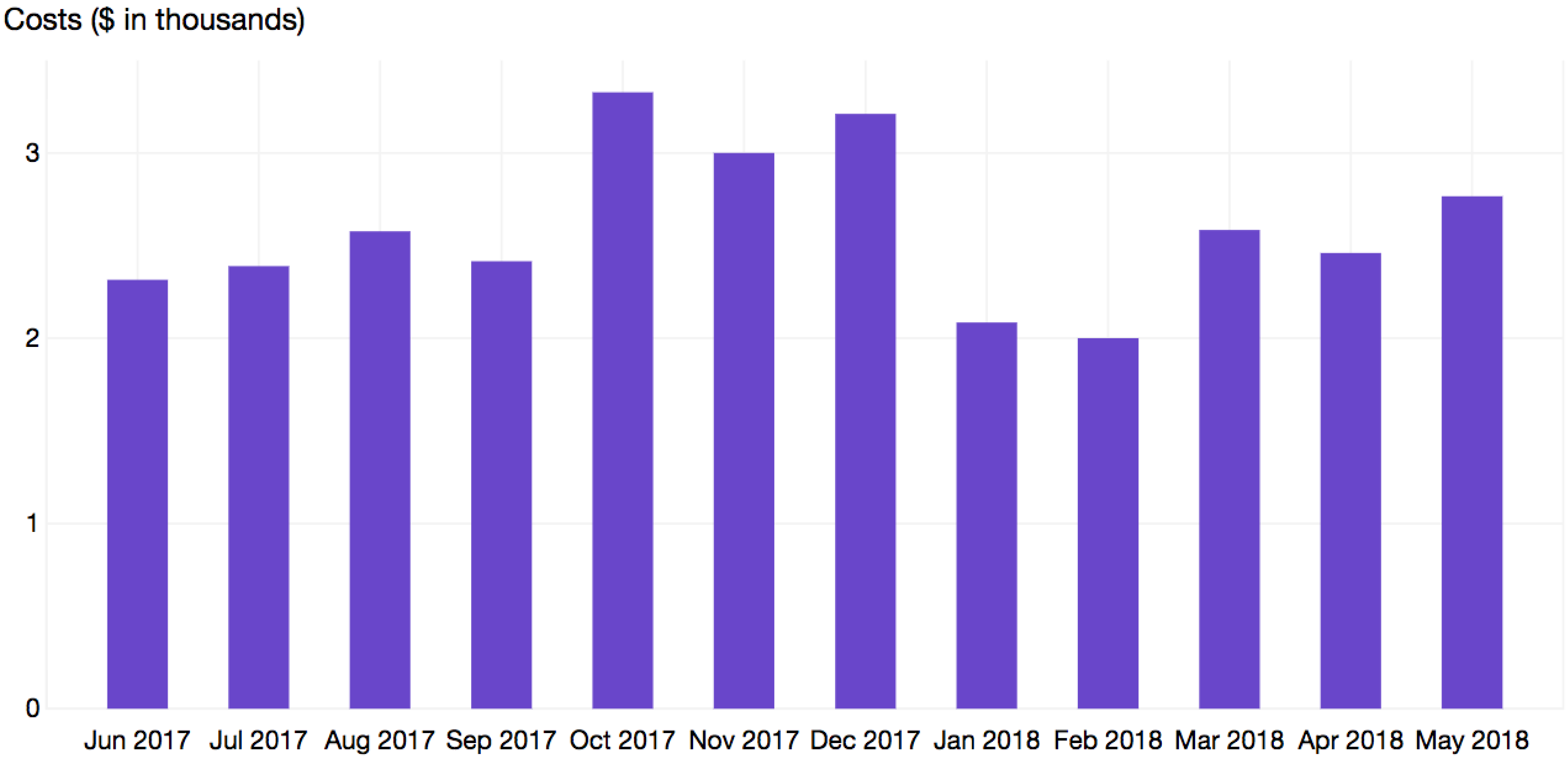
9:37 PM - 1 Mar 2018 from Manhattan, NY

173 Retweets 713 Likes

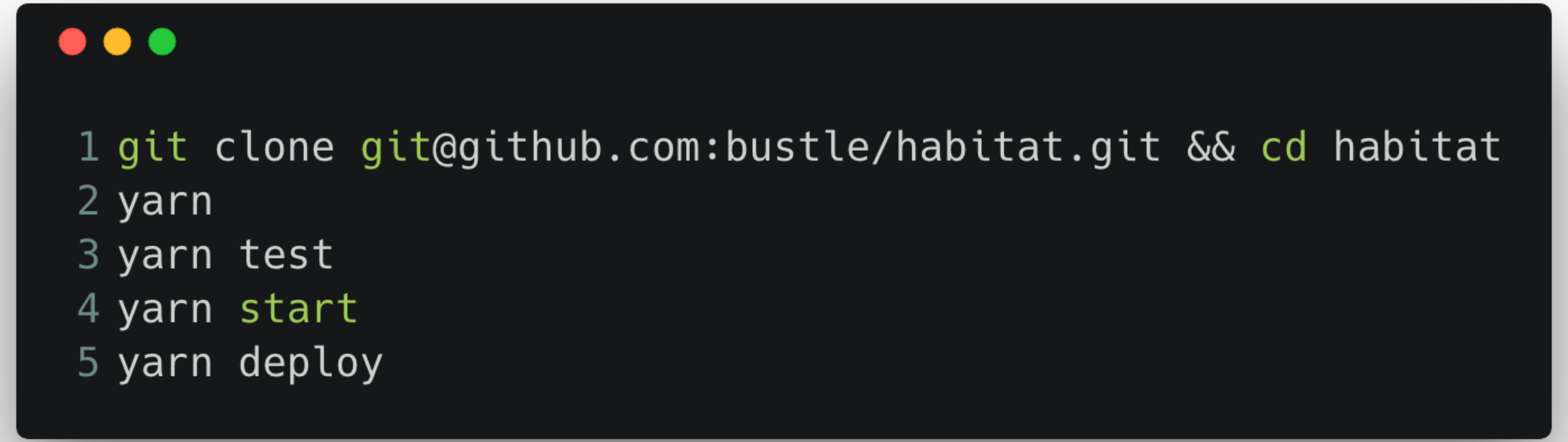
27 173 713

Tyler Love @tyleralove · Mar 1
We do all of this with a relatively tiny engineering team of 12 while simultaneously building compelling to use product that was never focused on social media audience gaming or egregious engagement metric hacking.

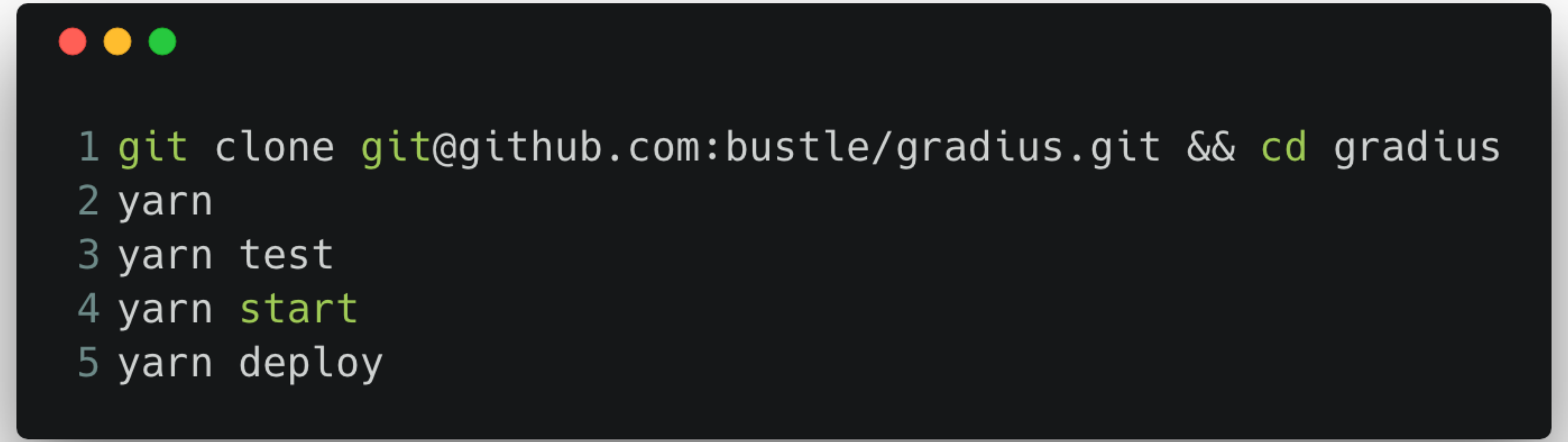
3 3 48


Costs ($ in thousands)
3
2
1
0
Jun 2017
Jul 2017
Aug 2017
Sep 2017
Oct 2017
Nov 2017
Dec 2017
Jan 2018
Feb 2018
Mar 2018
Apr 2018
May 2018

Merging Disciplines



```
git clone git@github.com:bustle/habitat.git && cd habitat
yarn
yarn test
yarn start
yarn deploy
```



```
git clone git@github.com:bustle/gradius.git && cd gradius
yarn
yarn test
yarn start
yarn deploy
```

@tyleralove