Hi there!

I’m Michael Schramm

github.com/wodka
twitter.com/wodkamichi

born in Salzburg

studying in Vienna

php development since 2003

co-founder of mymarket.io
e-commerce api solution

PHP beyond servers

by Michael Schramm

last few years

PHP CGI (Apache)

PHP FPM (Apache, NginX)

PHP as a Service

Apache

CGI module

PHP inside of Apache

one memory pool

change requires Apache restart

every request triggers php!!!

Apache

FPM

PHP has own service

Apache is a Proxy for PHP-FPM

nice but not perfect

NginX

FPM

PHP has own service

NginX is a Proxy for PHP-FPM

chain for requests (try_files)

only php if needed

further improvements through microcache (cache for 1sec)

https://www.nginx.com/blog/benefits-of-microcaching-nginx/
http://nginx.org/en/docs/http/ngx_http_core_module.html#try_files

all have ONE drawback

maintaining infrastructure is hard

microservices

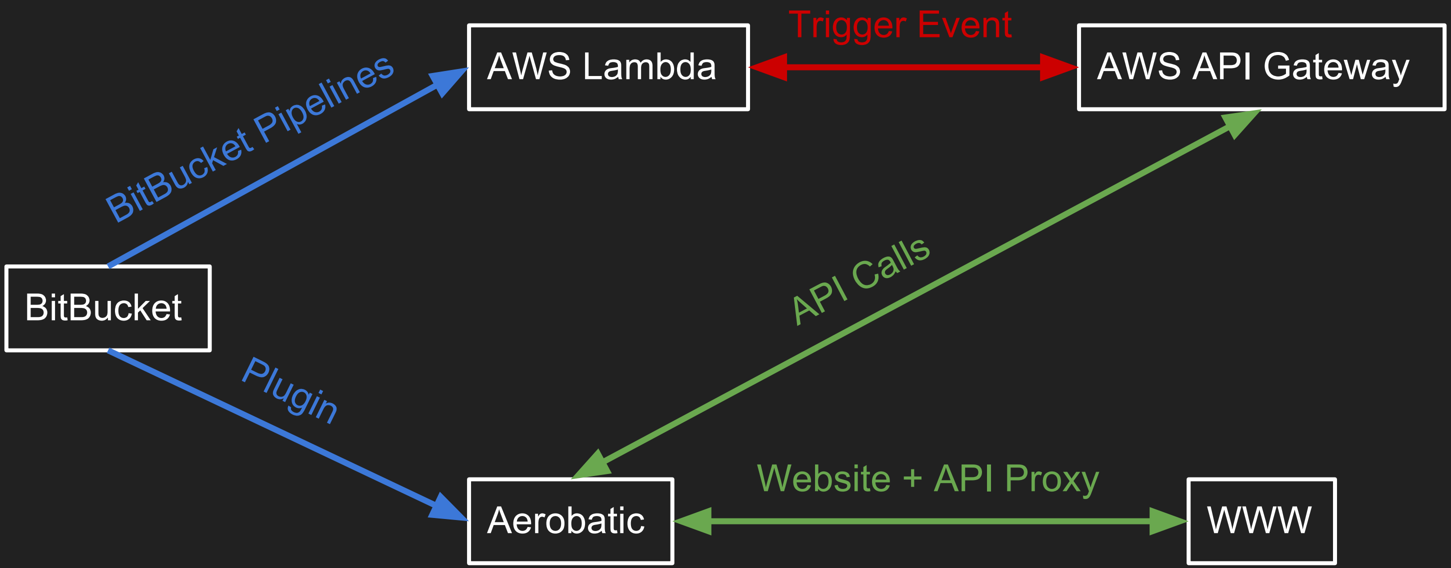
split your app

Create small modular components

Each component should have a well defined interface

One component one task

What we want to do:



This setup does not incur charges for a year

(as long as you stick to the aws free tier)

Backend

repository with php code and deploy logic

for lambda one zip with source and binary is required

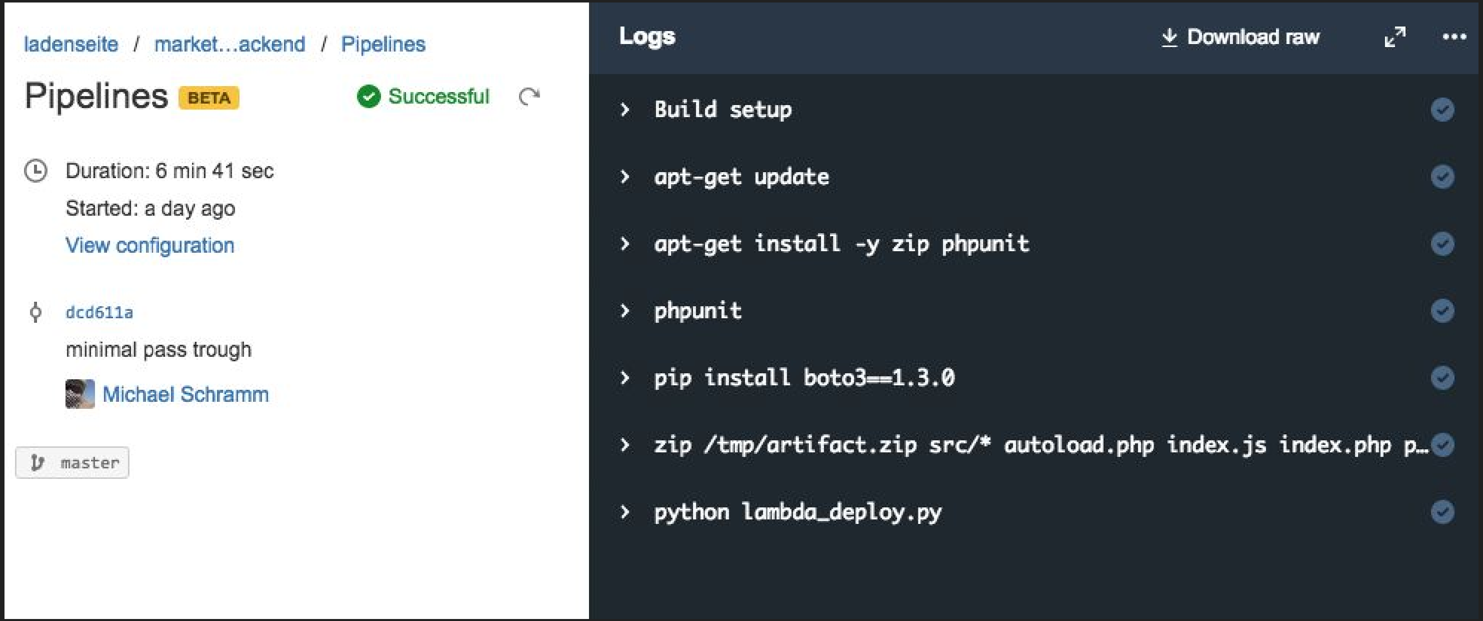
for us php 5.6 with curl,json,libxml and gd (building a new one is a bit complicated)

every push will build a new version of our “function”

https://github.com/awspilot/aws-lambda-php-template

BitBucket Pipelines

“Build, test and deploy from Bitbucket”



ladenseite / market...ackend / Pipelines

Pipelines BETA ✓ Successful

Duration: 6 min 41 sec

Started: a day ago

View configuration

dcd611a

minimal pass trough

Michael Schramm

master

Logs — Download raw

```
› Build setup                                                                 ✓
› apt-get update                                                              ✓
› apt-get install -y zip phpunit                                              ✓
› phpunit                                                                     ✓
› pip install boto3==1.3.0                                                    ✓
› zip /tmp/artifact.zip src/* autoload.php index.js index.php p…              ✓
› python lambda_deploy.py                                                     ✓
```

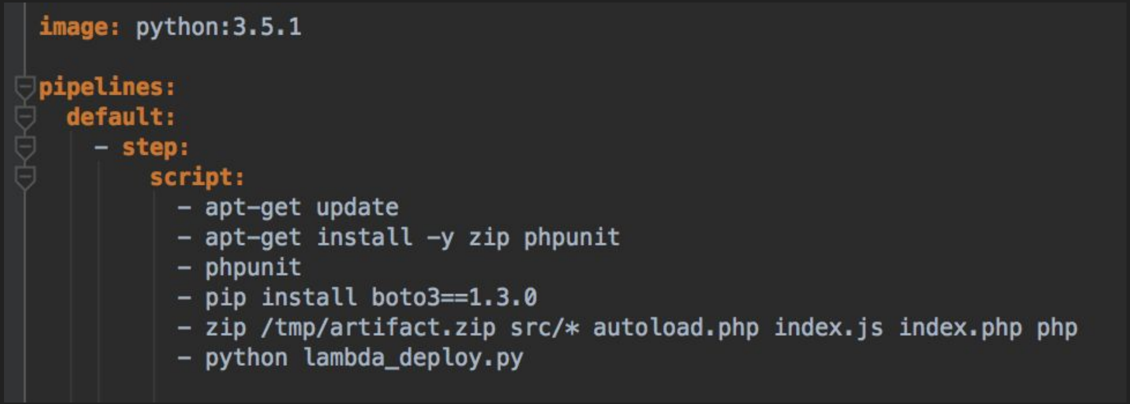
https://bitbucket.org/product/features/pipelines

BitBucket Pipelines

configuration similar to other CI solutions

image is a dockerhub id

pipelines can be separated by branch with different steps (one instance per step)



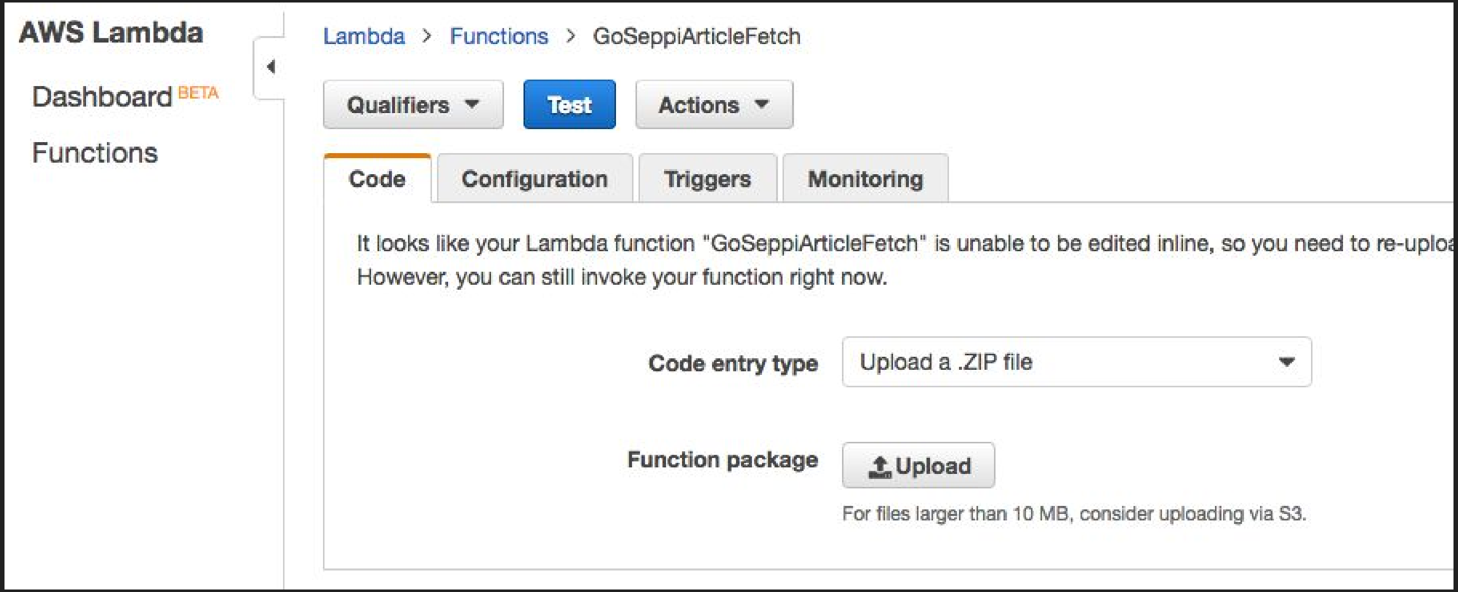
```
image: python:3.5.1

pipelines:
  default:
    - step:
        script:
          - apt-get update
          - apt-get install -y zip phpunit
          - phpunit
          - pip install boto3==1.3.0
          - zip /tmp/artifact.zip src/* autoload.php index.js index.php php
          - python lambda_deploy.py
```

https://bitbucket.org/awslabs/aws-lambda-bitbucket-pipelines-python

AWS Lambda

“Run code without thinking about servers. Pay for only the compute time you consume.”



It looks like your Lambda function "GoSeppiArticleFetch" is unable to be edited inline, so you need to re-uploa
However, you can still invoke your function right now.

Code entry type: Upload a .ZIP file

Function package: Upload

For files larger than 10 MB, consider uploading via S3.

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

AWS Lambda

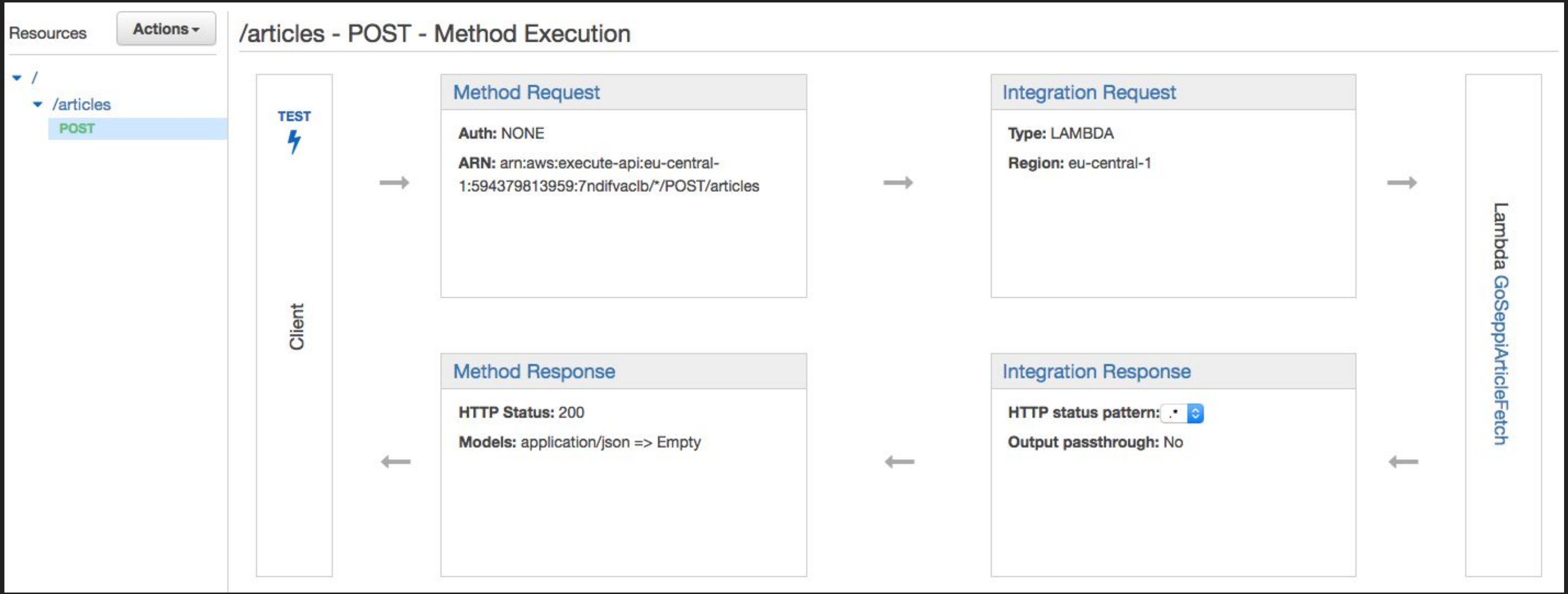
quite simple

1. create a new function
2. you can either edit it online or upload a zip for complex setups
3. setup triggers (like API Gateway)

there are other use cases like: push images to s3 from your application, on creation of images resize those in lambda to have them distributed through cloudfront

AWS API Gateway

“is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale.”



/articles - POST - Method Execution

Method Request

Auth: NONE

ARN: arn:aws:execute-api:eu-central-1:594379813959:7ndifvaclb/*/POST/articles

Integration Request

Type: LAMBDA

Region: eu-central-1

Method Response

HTTP Status: 200

Models: application/json => Empty

Integration Response

HTTP status pattern: .*

Output passthrough: No

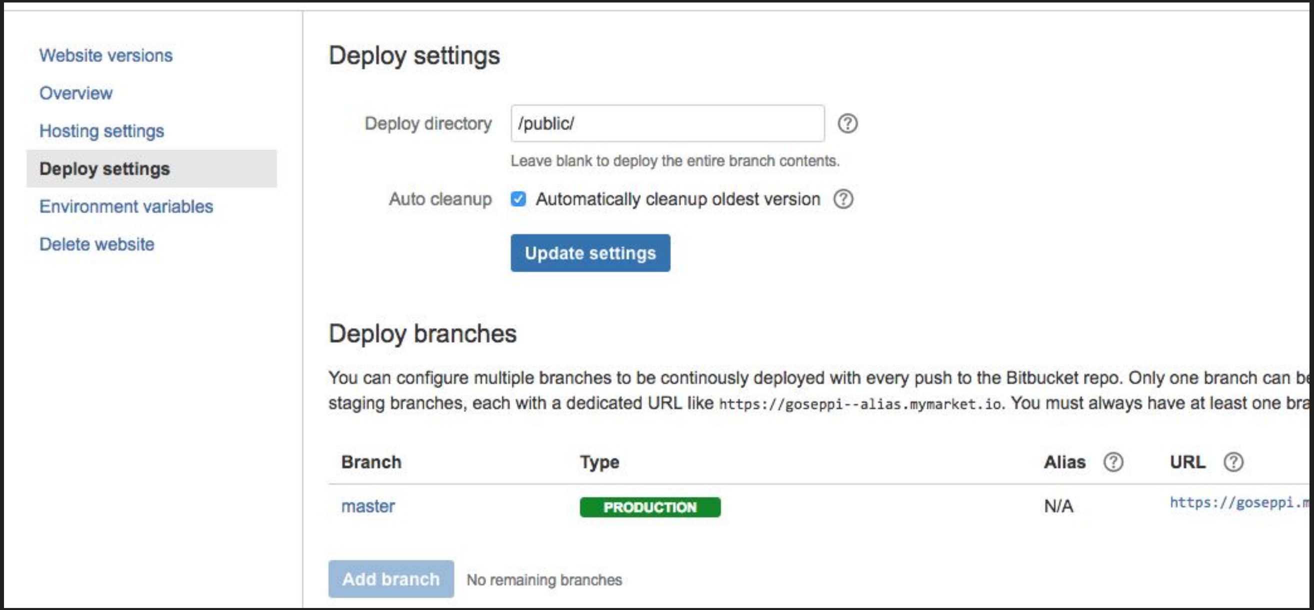
https://aws.amazon.com/api-gateway/

Let’s make a change and push it

How it works together

Aerobatic.com

“The easiest way to build, deploy, and host your static HTML website in Bitbucket”



Deploy settings

Deploy directory: /public/

Leave blank to deploy the entire branch contents.

Auto cleanup: ☑ Automatically cleanup oldest version

Deploy branches

You can configure multiple branches to be continously deployed with every push to the Bitbucket repo. Only one branch can be
staging branches, each with a dedicated URL like `https://goseppi--alias.mymarket.io`. You must always have at least one bra

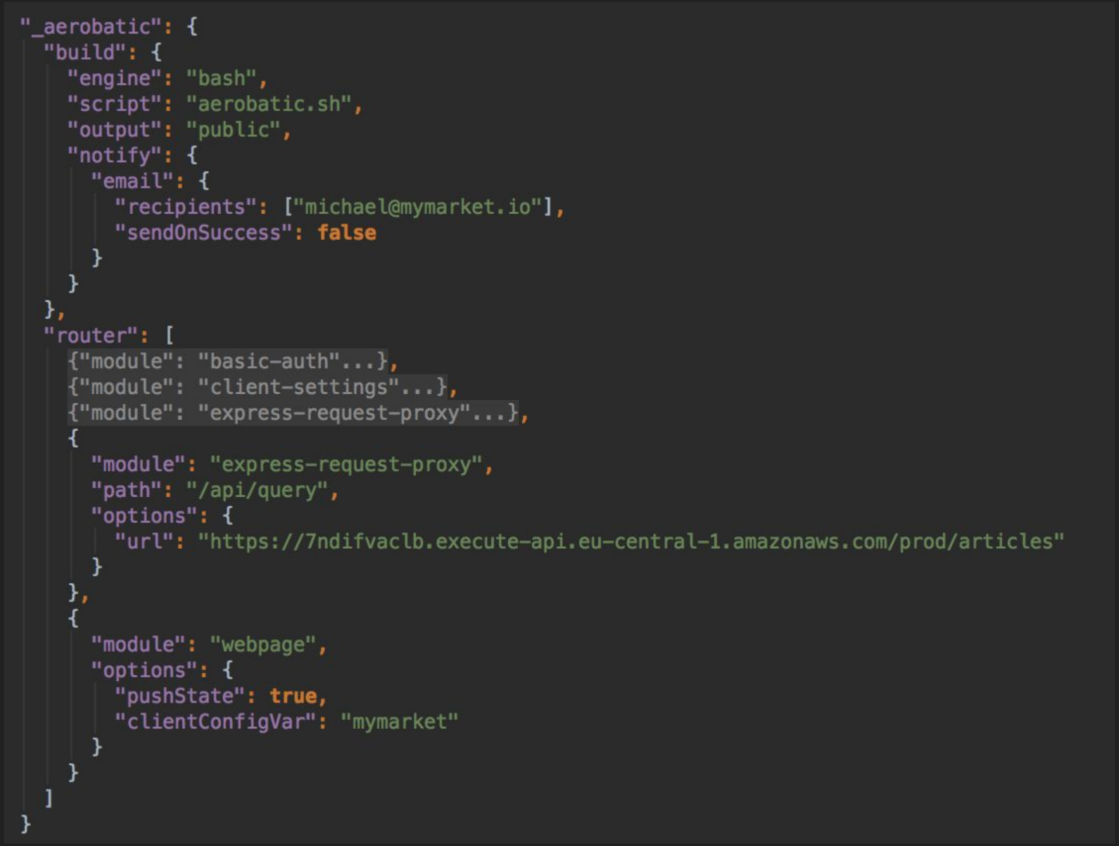
Branch	Type	Alias	URL
master	PRODUCTION	N/A	https://goseppi.m

No remaining branches

https://www.aerobatic.com/

Aerobatic.com

configuration in package.json

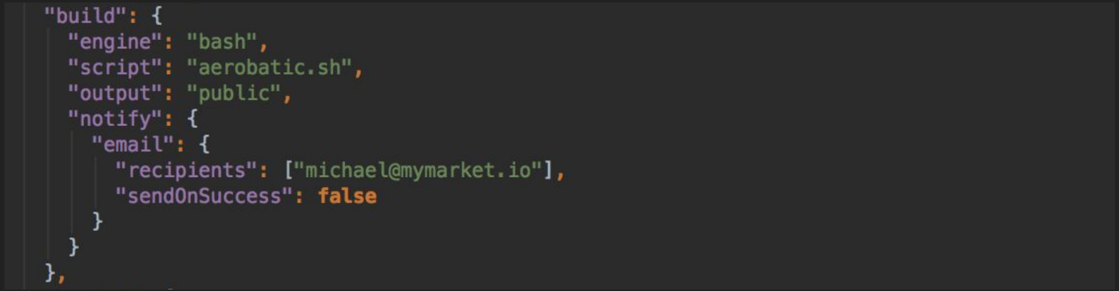


```
"_aerobatic": {
  "build": {
    "engine": "bash",
    "script": "aerobatic.sh",
    "output": "public",
    "notify": {
      "email": {
        "recipients": ["michael@mymarket.io"],
        "sendOnSuccess": false
      }
    }
  },
  "router": [
    {"module": "basic-auth"...},
    {"module": "client-settings"...},
    {"module": "express-request-proxy"...},
    {
      "module": "express-request-proxy",
      "path": "/api/query",
      "options": {
        "url": "https://7ndifvaclb.execute-api.eu-central-1.amazonaws.com/prod/articles"
      }
    },
    {
      "module": "webpage",
      "options": {
        "pushState": true,
        "clientConfigVar": "mymarket"
      }
    }
  ]
}
```

Aerobatic.com - build

engine can be one of: bash, npm, hugo, jekyll, middleman

bash executes one script and takes the output to deploy



```
"build": {
  "engine": "bash",
  "script": "aerobatic.sh",
  "output": "public",
  "notify": {
    "email": {
      "recipients": ["michael@mymarket.io"],
      "sendOnSuccess": false
    }
  }
},
```

https://www.aerobatic.com/docs/automated-builds

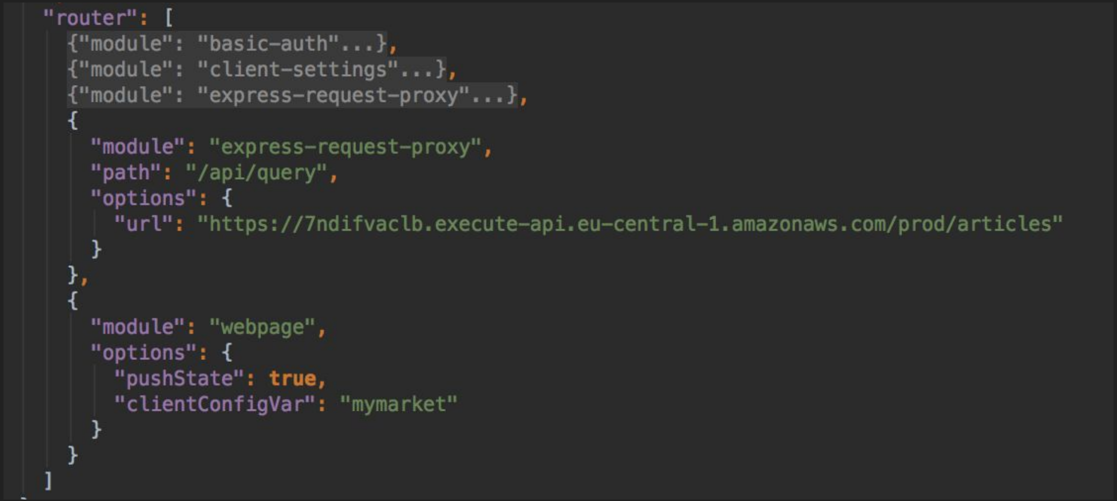
Aerobatic.com - router

router is a list of modules

webpage: passes all calls to index.html in root

express-request-proxy: proxy service, enrich request with additional headers, etc

client-settings: environment specific values that are added to html



```
"router": [
  {"module": "basic-auth"...},
  {"module": "client-settings"...},
  {"module": "express-request-proxy"...},
  {
    "module": "express-request-proxy",
    "path": "/api/query",
    "options": {
      "url": "https://7ndifvaclb.execute-api.eu-central-1.amazonaws.com/prod/articles"
    }
  },
  {
    "module": "webpage",
    "options": {
      "pushState": true,
      "clientConfigVar": "mymarket"
    }
  }
]
```

https://www.aerobatic.com/docs/http-proxy show the template

questions?

Slides available on https://github.com/viennaphp