OPENFAAS

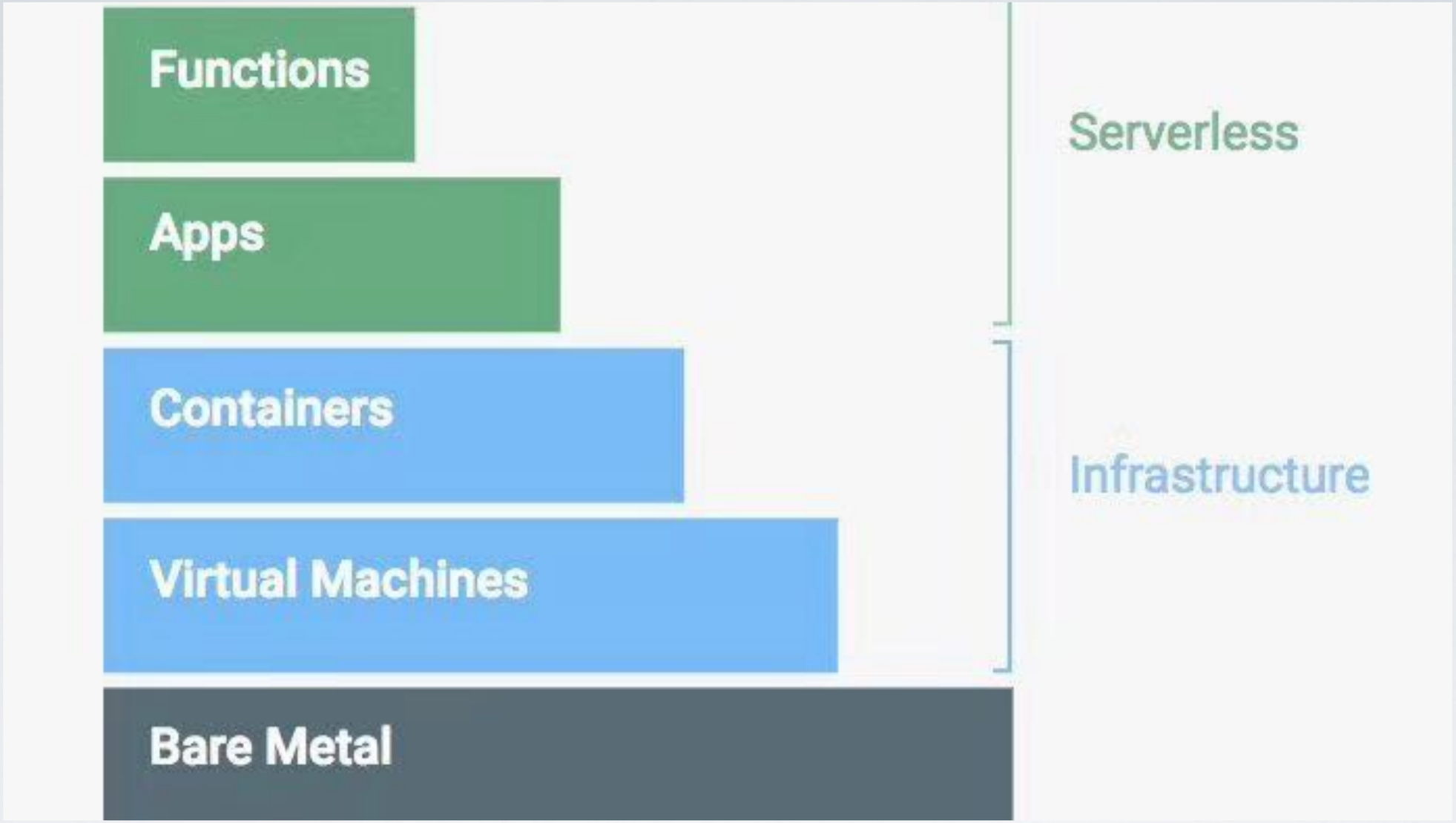
Serverless Functions Made Simple

“今天大多数公司在开发应用程序并将其部署在服务器上的时候，无论是选择公有云还是私有的数据中心，都需要提前了解究竟需要多少台服务器、多大容量的存储和数据库的功能等。并需要部署运行应用程序和依赖的软件到基础设施之上。假设我们不想在这些细节上花费精力，是否有一种简单的架构模型能够满足我们这种想法？这个答案已经存在，这就是今天软件架构世界中新鲜但是很热门的一个话题——Serverless（无服务器）架构。”

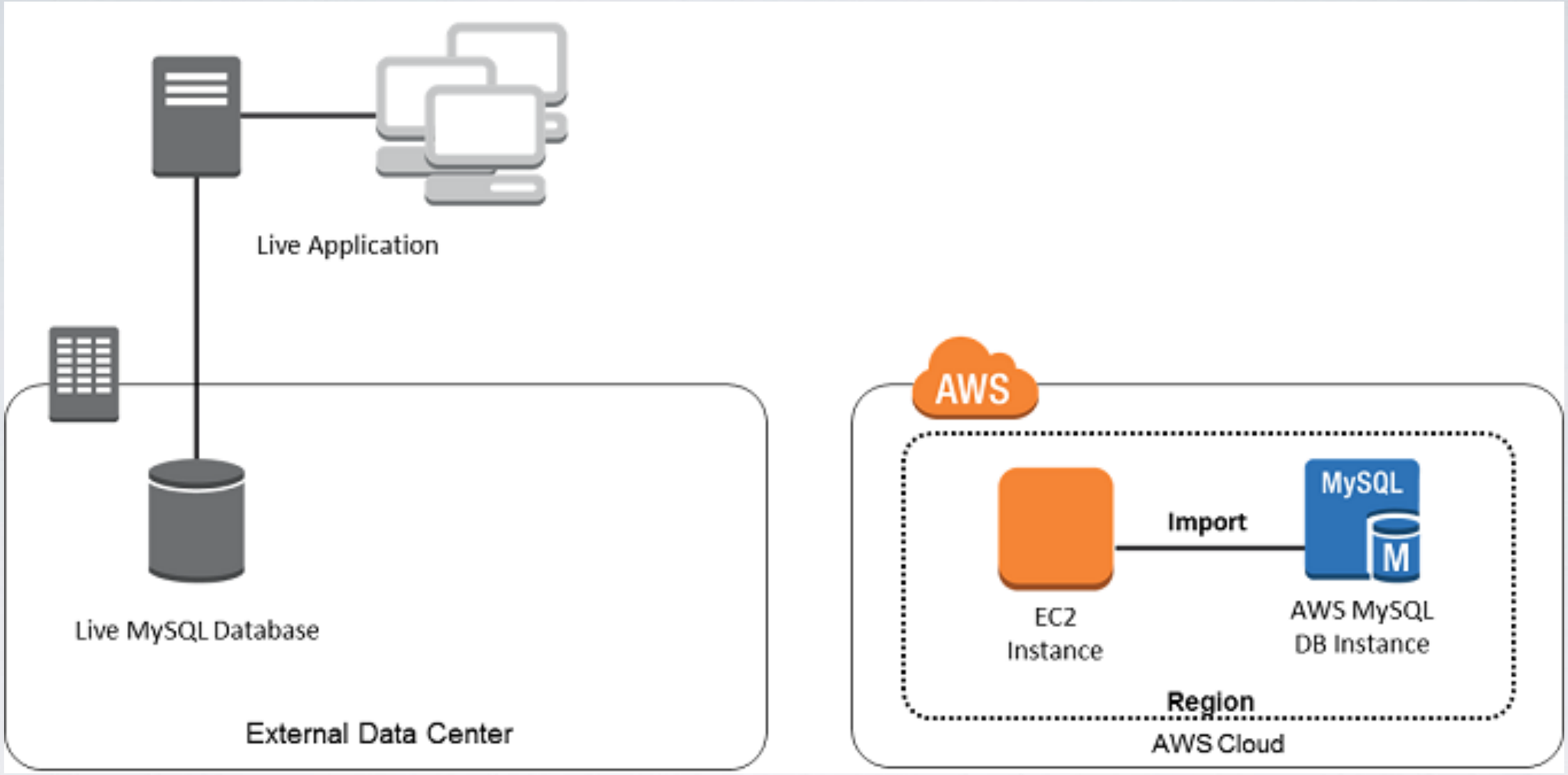
——AWS 费良宏

什么是SERVERLESS?

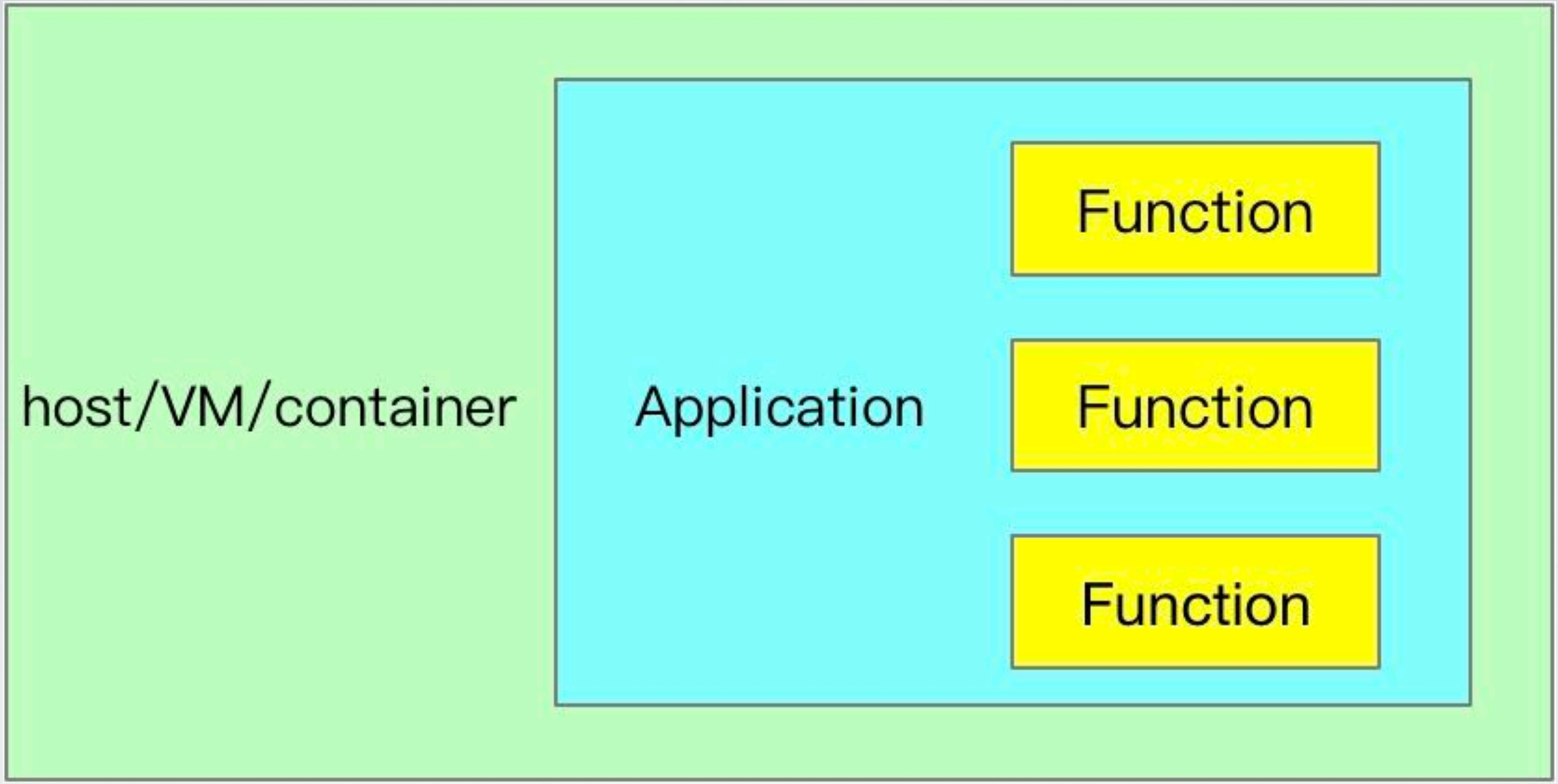
- IaaS： Infrastructure as a Service
  - IaaS从本质上讲是服务器租赁并提供基础设施外包服务。
- PaaS： Platform as a Service
  - 构建在IaaS之上的一种平台服务，提供操作系统安装、监控和服务发现等功能。
  - 广泛使用的技术是docker。
  - 对软件更高的抽象层次，接触到应用程序运行环境的本身，可由开发者自定义，不必接触更底层的操作系统



Serverless（无服务器架构）指的是由开发者实现的服务端逻辑运行在无状态的计算容器中，它由事件触发，完全被第三方管理，其业务层面的状态则被开发者使用的数据库和存储资源所记录。



BaaS：后端即服务，一般是一个个的API调用后端或者别人已经实现好的程序逻辑。



FaaS：函数即服务，本质上是一种事件驱动的由消息触发的服务，FaaS供应商一般会集成各种同步和异步的事件源，通过订阅这些事件源，可以突发或者定期的触发函数运行。

- 两者都为我们的计算资源提供了保障。BaaS其实是服务外包，FaaS使我们更加关注业务逻辑。

- 两者都不需要我们关注应用程序所在的服务器，但是服务器依然可观存在。

FAAS

优势

- 降低人力成本
- 降低风险
- 减少资源开销
- 增加伸缩的灵活性
- 缩短创新周期

不足

- 不适合长时间运行的应用
- 状态管理
- 冷启动时间
- 缺乏调试和开发工具
- 构建复杂

应用场景

- 发送通知
- webhook
- 轻量级API
- 物联网
- 数据统计分析
- Trigger和定时任务
- 聊天机器人

Function-as-a-Service Landscape

Redpoint

Tools






Lambda cult

Libraries

stdlib

Python-λ

FDK

Security

snyk

PureSec

Frameworks

Claudia.js

Zappa

Lambdify

dawson

Lambda-uploader

Lambder

Shep

Serverless

Chalice

Lambdoku

λ Gordon

Lambda-restify

Lambada

SCAR

Lambda Forest

Kappa

DEEP

EFFE

Platforms





DISTRIBYTE

Open Lambda

FaaS全景图

syncano


spotinst

nuclio

OVH Functions

GALACTIC FOG

PubNub BLOCKS

Weblab.io

fn

OVRCLK

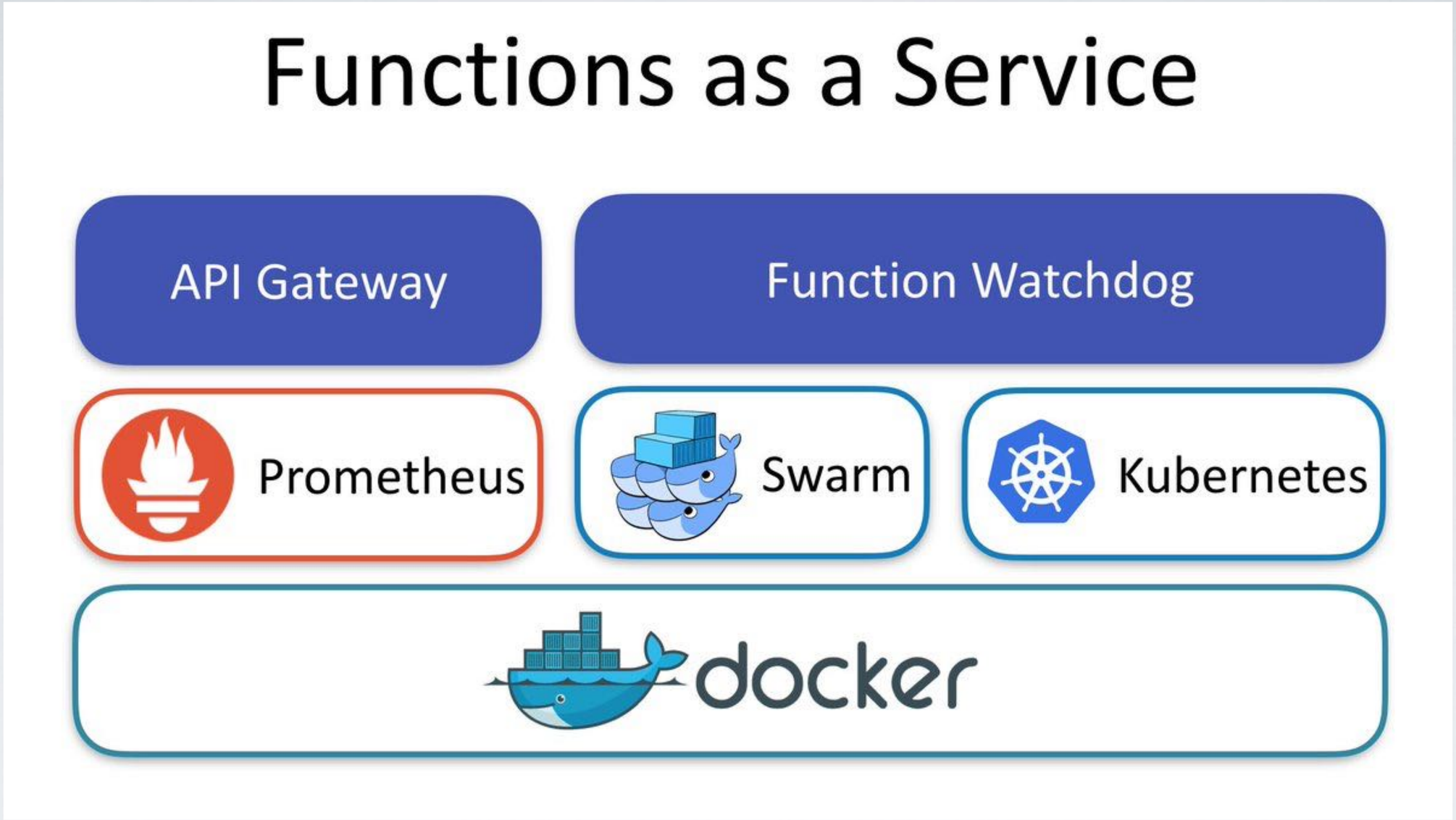
NANO-LAMBDA

BINARIS

Functions

- OpenFaaS - 生态丰富，开发者活跃
- Fn - oracle开源，还比较简陋
- OpenWhisk - Apache基金会，资料少
- Knative - Google和IBM开源，基于istio，刚起步

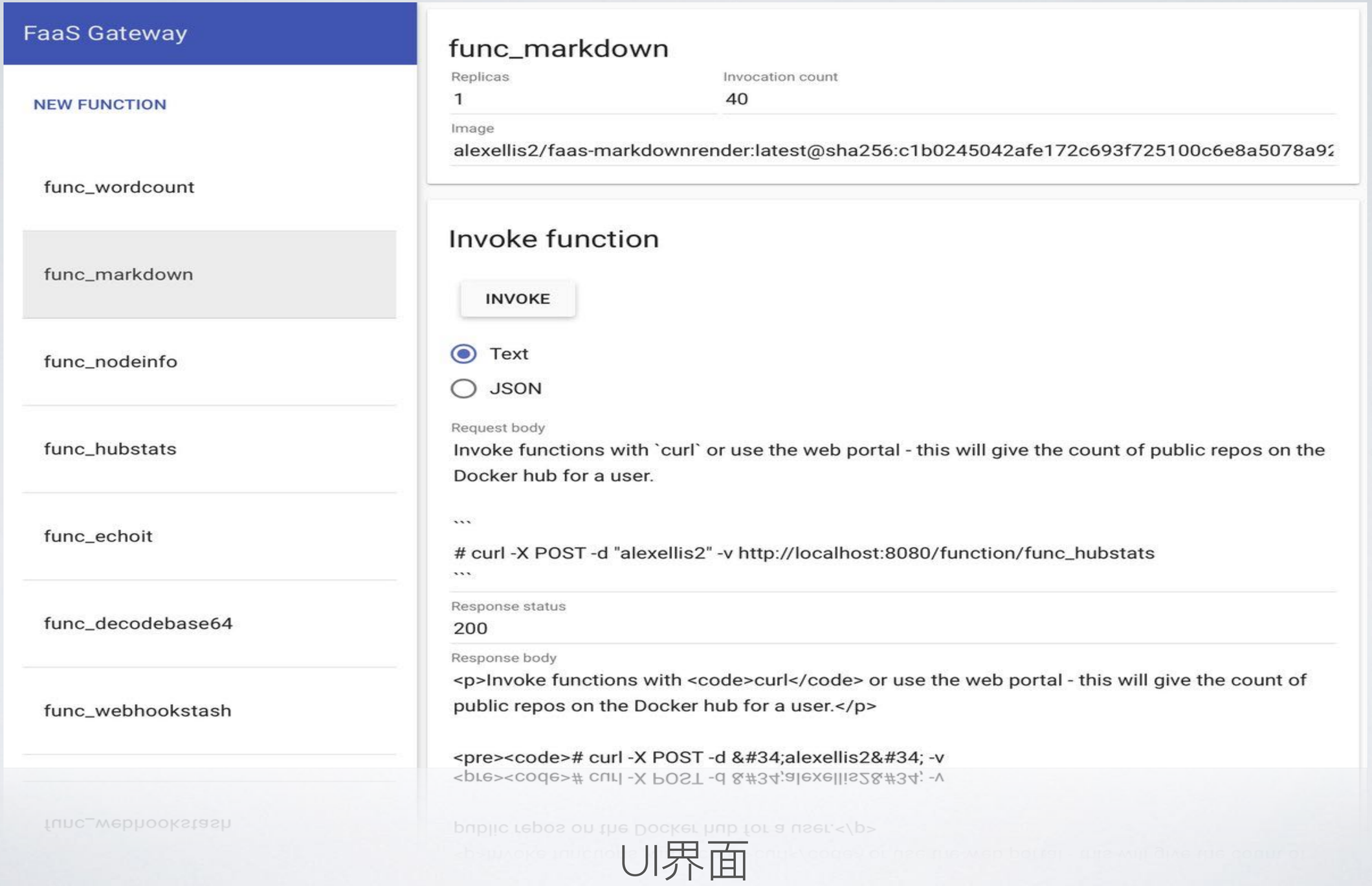
OPEN FAAS



OpenFaaS是一款高人气的开源faas框架，可以直接在Kubernetes上运行，也可以基于swarm运行。

HIGHLIGHTS

- UI支持
- 可以使用多种编程语言，也支持二进制文件打包的docker镜像
- 可在Kubernetes和swarm上运行
- 丰富的CLI命令
- YAML配置
- 根据需求自动伸缩
- 函数商店


FaaS Gateway
NEW FUNCTION
func_wordcount
func_markdown
func_nodeinfo
func_hubstats
func_echoit
func_decodebase64
func_webhookstash
func_markdown
Replicas
1
Invocation count
40
Image
alexellis2/faas-markdownrender:latest@sha256:c1b0245042afe172c693f725100c6e8a5078a92
Invoke function
INVOKE
Text
JSON
Request body
Invoke functions with `curl` or use the web portal - this will give the count of public repos on the Docker hub for a user.
```
# curl -X POST -d "alexellis2" -v http://localhost:8080/function/func_hubstats
```
Response status
200
Response body
Invoke functions with <code>curl</code> or use the web portal - this will give the count of public repos on the Docker hub for a user.
<pre><code># curl -X POST -d "alexellis2" -v

UI界面

- Python example:

```python
import requests

def handle(req):
    r =  requests.get(req, timeout = 1)
    print(req +" => " + str(r.status_code))
```

handler.py

- Node.js example:

```javascript
"use strict"

module.exports = (callback, context) => {
    callback(null, {"message": "You said: " + context})
}
```

handler.js

函数示例

FaaS

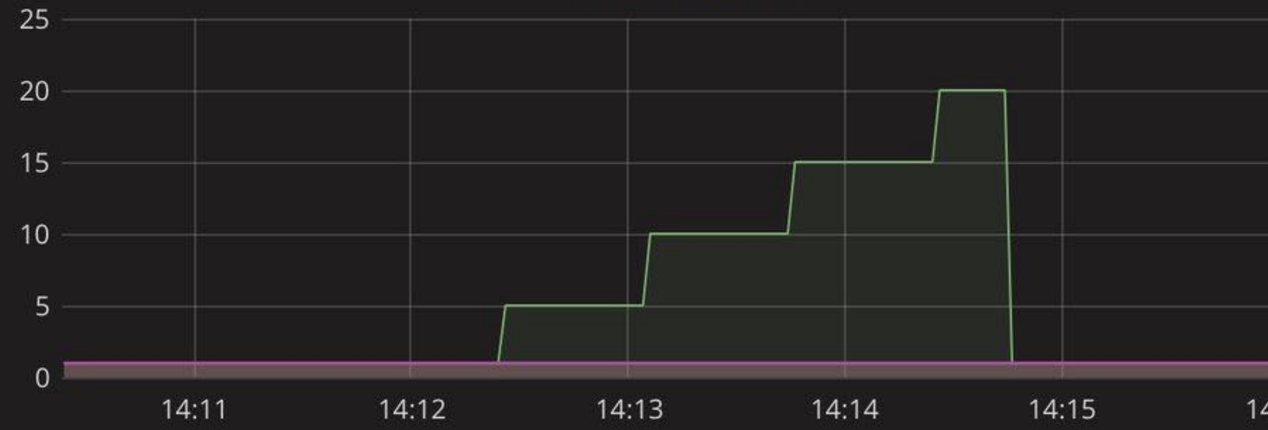
Zoom Out

Apr 14, 2017 14:10:23 to Apr 14, 2017 14:16:17

Function rate

unc_echoit 200 func_fanclub 200 func_getavatar 200 func_markdown 200
unc_mobymingle 200 func_myassistant 200 func_payroll 200 func_echoit 500
unc_fanclub 500 func_getavatar 500 func_markdown 500 func_mobymingle 500
unc_myassistant 500 func_payroll 500

Replica scaling

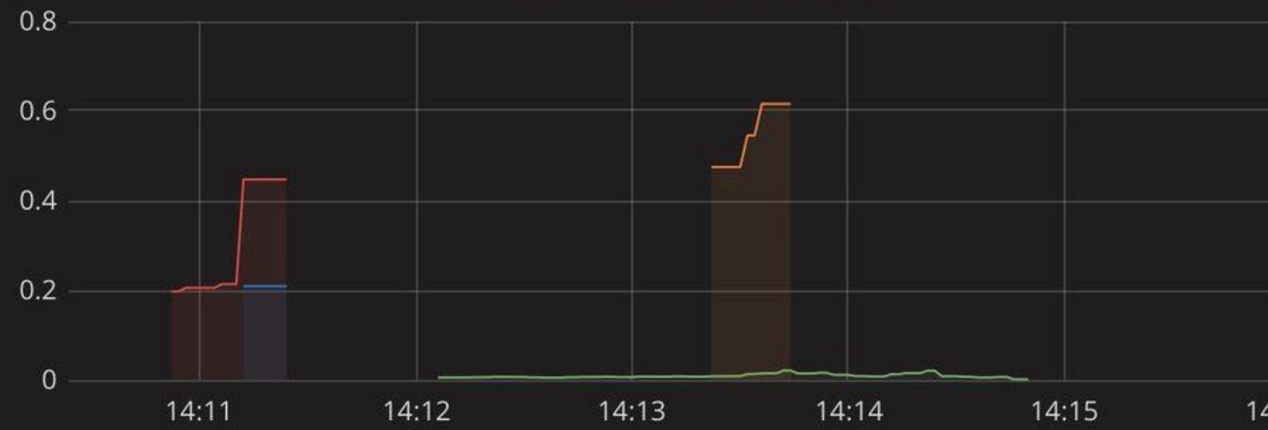


func_echoit func_fanclub func_getavatar func_markdown func_mobymingle
func_myassistant func_payroll

Total requests - 200 OK

5053

Execution duration (s)



func_echoit func_fanclub func_getavatar func_mobymingle func_myassistant
func_payroll

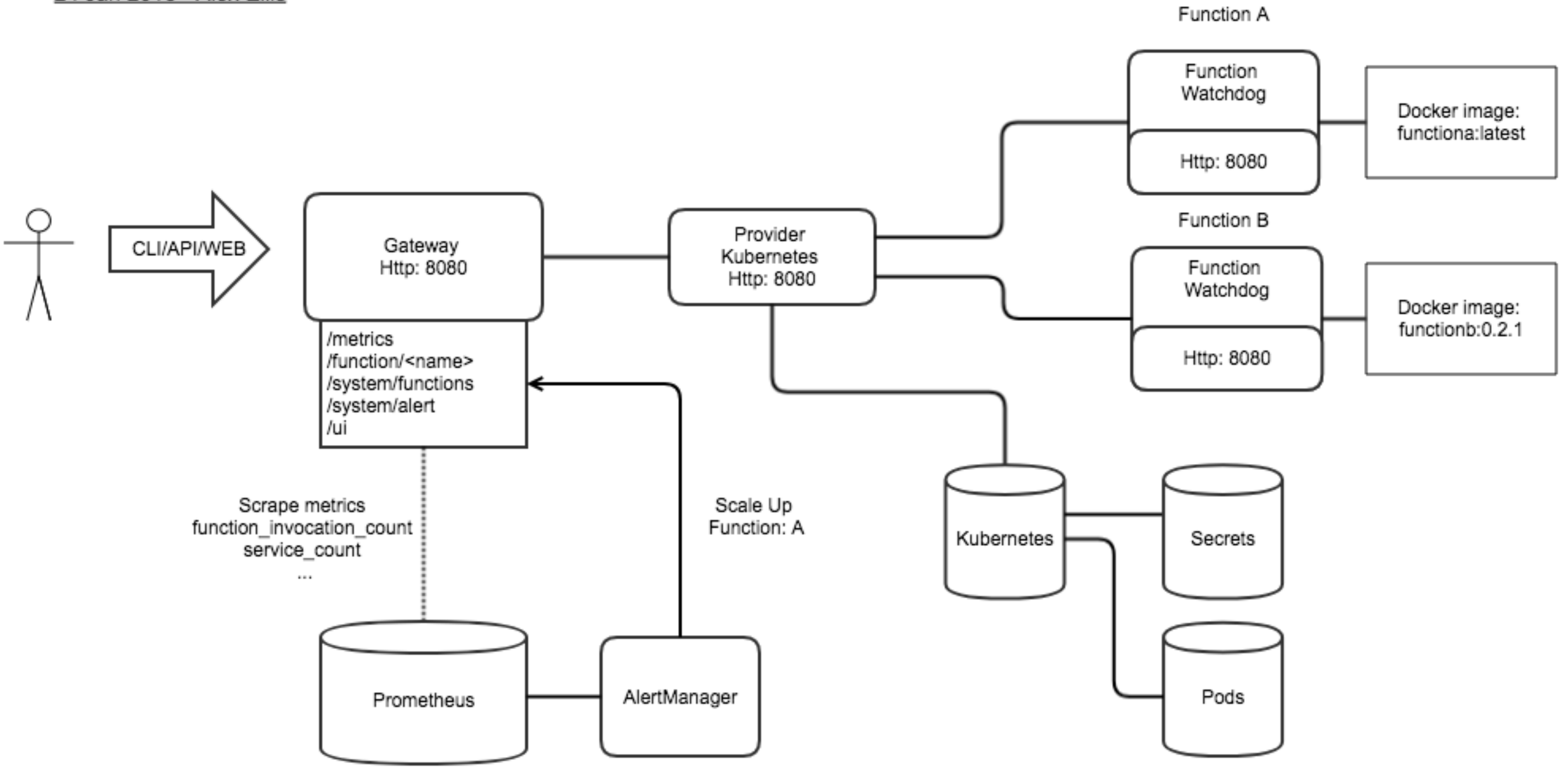
基础组件

- Gateway
- Provider
- queue-worker
- watchdog

GATEWAY

- 为函数调用提供一个路由，起到一个代理转发的作用
- 内置UI界面，可访问函数商店
- Prometheus收集监控指标
- 接收AlertManager通知，自动伸缩

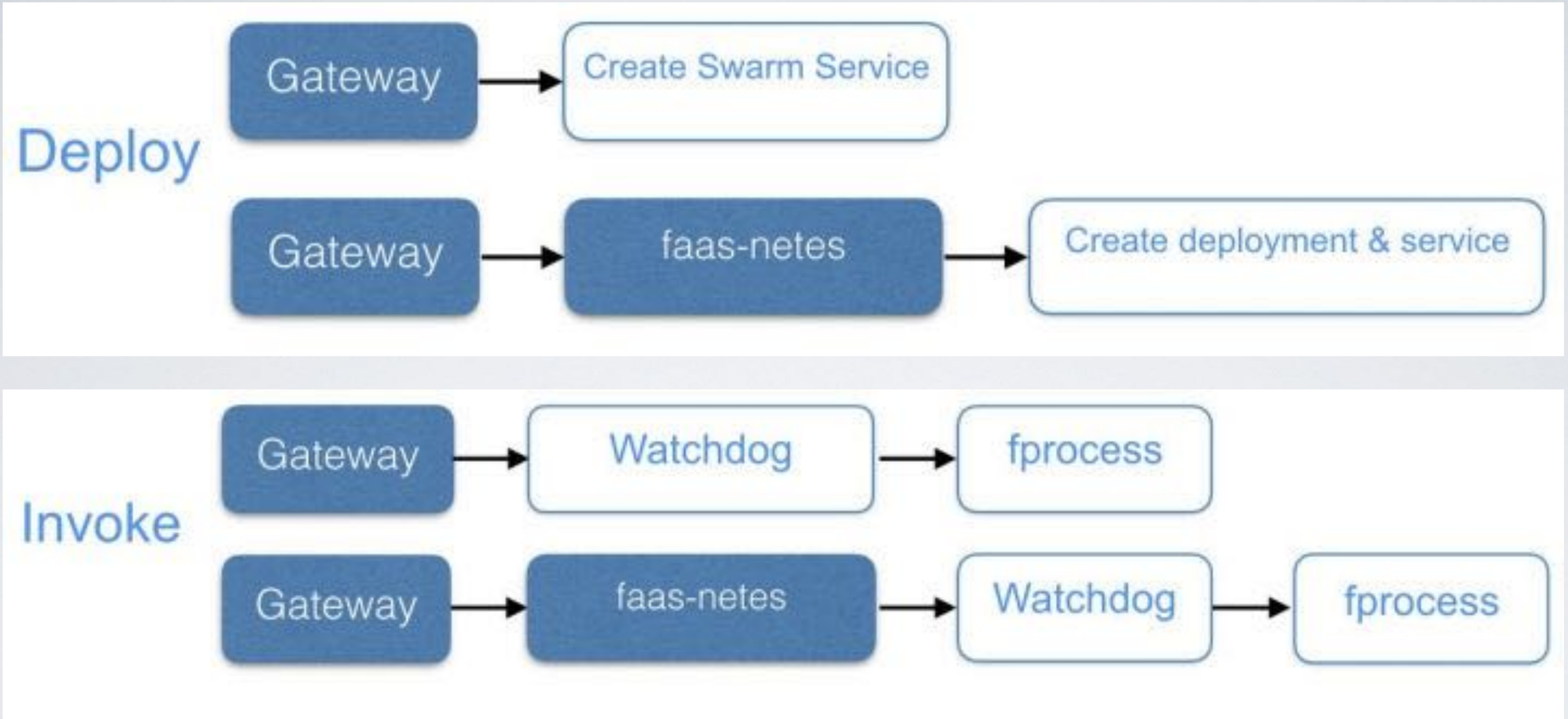
OpenFaaS conceptual diagram
24 Jan 2018 - Alex Ellis



Multiple replicas may exist for the Gateway, the provider and each function

PROVIDER

- 本质上是操作Kubernetes的API
- List / Create / Delete 一个函数
- 获取函数
- 缩放函数
- 调用函数


Deploy
Gateway
Create Swarm Service
Gateway
faas-netes
Create deployment & service
Invoke
Gateway
Watchdog
fprocess
Gateway
faas-netes
Watchdog
fprocess

部署和调用函数的过程

自动伸缩函数

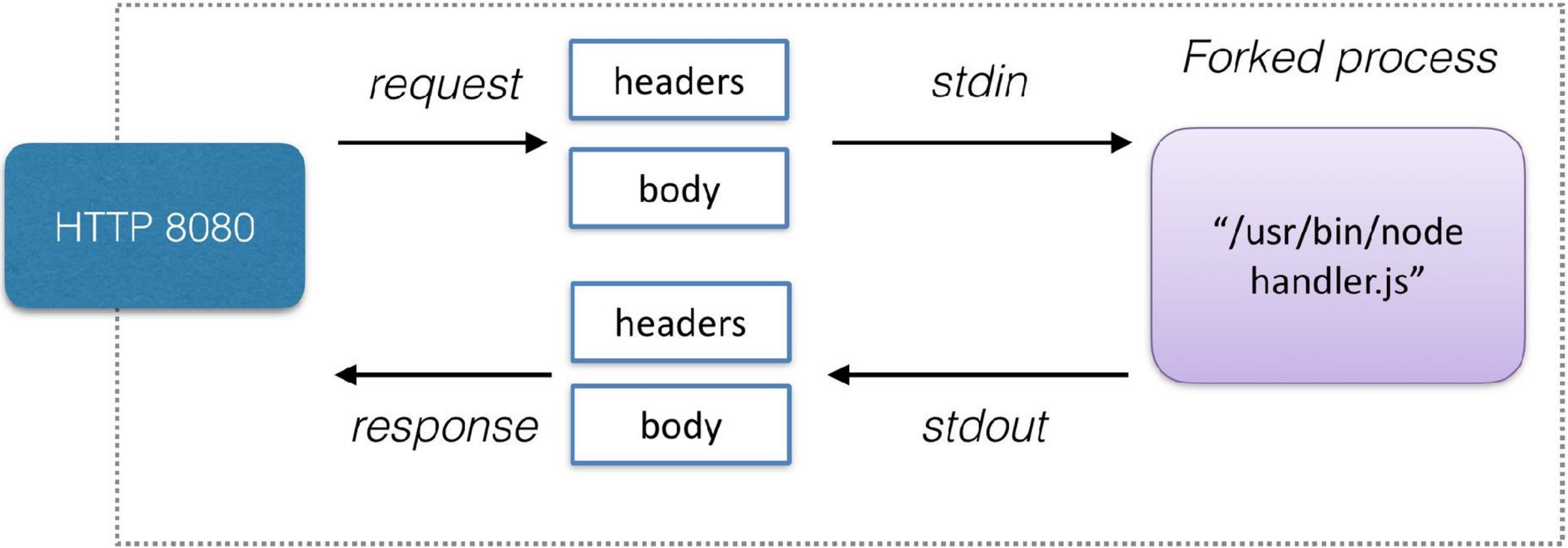
- 根据每秒请求数
- 内存和CPU使用量
- 最小或最大副本数
  - com.openfaas.sacle.min 最小
  - com.openfaas.scale.max 最大
  - com.openfaas.sacle.factor 步长

- 手动调用API伸缩
- AlertManager触发
  - 获取现在副本数
  - 计算新副本数： min+(max/100)*factor
  - 调用Kubernetes API设置新副本数

WATCHDOG

- 职责是调用函数，healthcheck和超时
- 任何二进制都会被watchdog变成一个函数
- 小型的http server

Watchdog


HTTP 8080
request
headers
body
stdin
Forked process
"/usr/bin/node handler.js"
response
headers
body
stdout

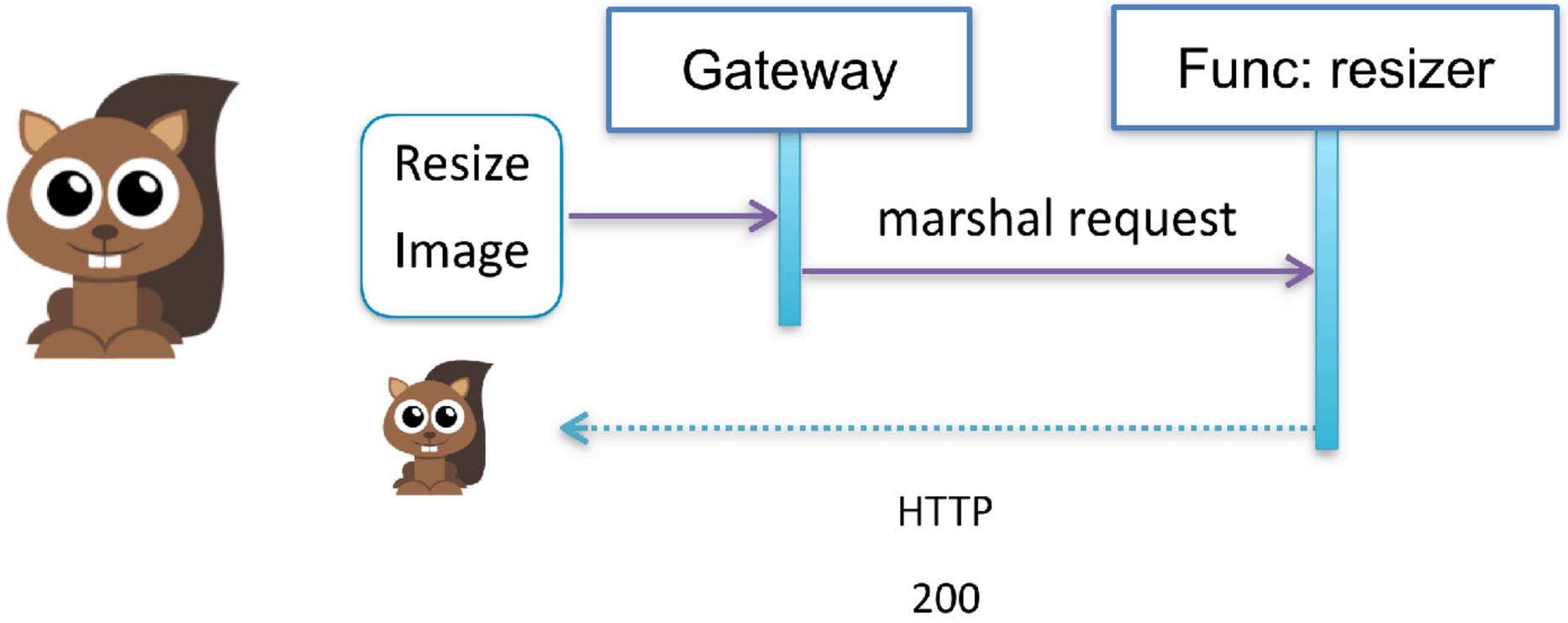
QUEUE-WORKER

- 同步函数
  - 路由是：/function/:name
  - 等待
  - 结束时得到结果
  - 明确知道是成功还是失败
- 异步函数
  - 路由是：/async-function/:name
  - http的状态码为202——即时响应吗
  - 从queue-worker中调用函数
  - 默认情况下，函数的执行结果是被丢弃的

- 依赖于NATS和NATS Streaming
- Gateway是一个发布者
- queue-worker是一个订阅者

Synchronous invocation

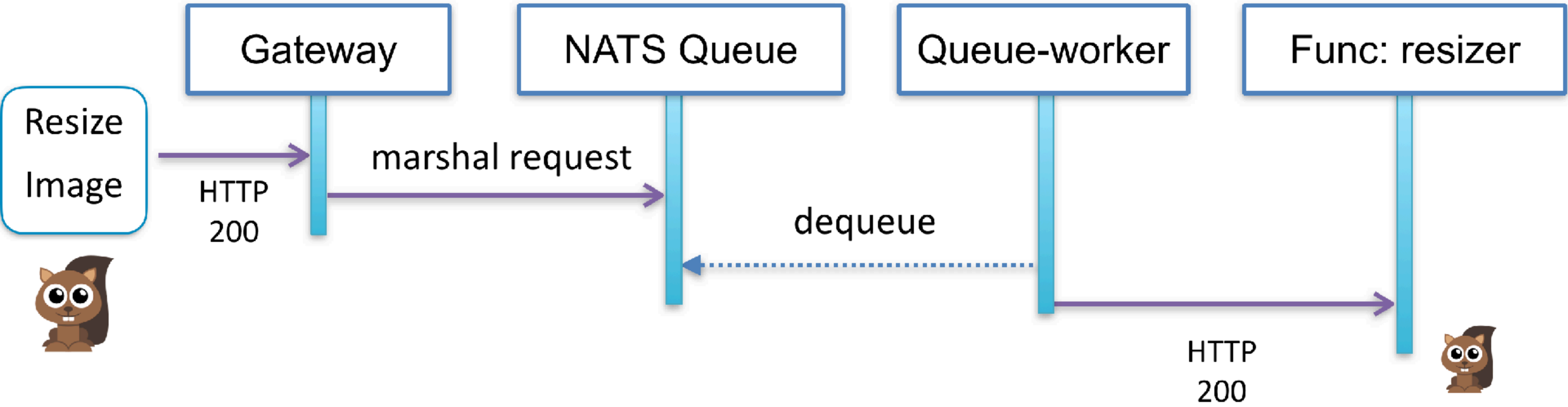
IN STORE RETURNS



OPENFAAS

Asynchronous invocation

RETURN A PRODUCT BY MAIL



OPENFAAS

定制

- faas-cli
  - 修改生成的yml模板
- templates
  - 剪裁template，自定义支持语言
  - 修改dockerfile

总结

- 云改变了我们对操作系统的认知，原来一个系统的计算资源、存储和网络是可以分离配置，而且还可以弹性扩展，但是长久以来，我们在开发应用的时候始终没有摆脱服务器的束缚，应用必须运行在不论是实体还是虚拟的服务器上，必须经过部署、配置和初始化才可以运行，还需要对服务器进行监控和管理，还需要保证数据的安全性。

- 当我们将应用程序迁移到容器和虚拟机中时，其实对于应用程序本身的体系结构并没有多少改变，只不过有些流程和规定需要遵守，比如12因素应用守则，但是serverlss对应用程序的体系结构来说就是一次颠覆了，通常我们需要考虑事件驱动模型，更加细化的不熟形式，以及在FaaS组件之外保持状态的需求。

学习资料

- 官方文档：https://docs.openfaas.com/
- workshop：https://github.com/openfaas/workshop
- 理解Serverless和FaaS：https://jimmysong.io/posts/what-is-serverless/
- Serverless架构应用开发指南：https://serverless.ink/
- Kubernetes handbook：https://jimmysong.io/kubernetes-handbook/