

Serverless架构在机器 人行业应用案例与分 析

黄珏珅 CTO 上海思岚科技

Agenda

- ▶ 云计算架构的演进
- ▶ 机器人行业的需求
- ▶ Serverless架构在机器人行业的应用

云计算架构的演进

计算平台



软件架构

云计算架构的演进

架构



平台



1980s

2006

2011

2014

数据中心 / 单体架构

计算



独立服务器

运维



处理硬件故障
重启服务器

困难

难以实现HA

运维成本高

固定资产成本高

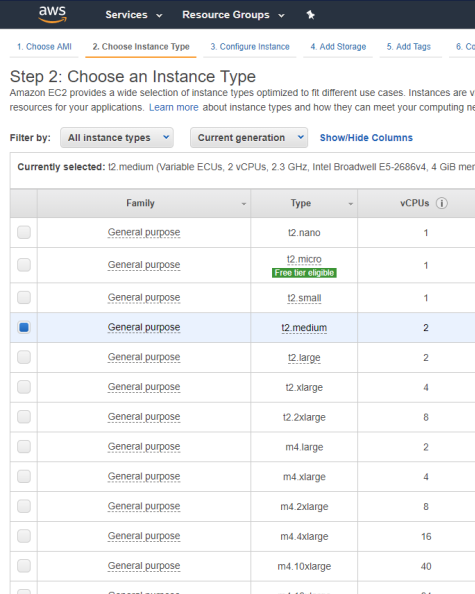
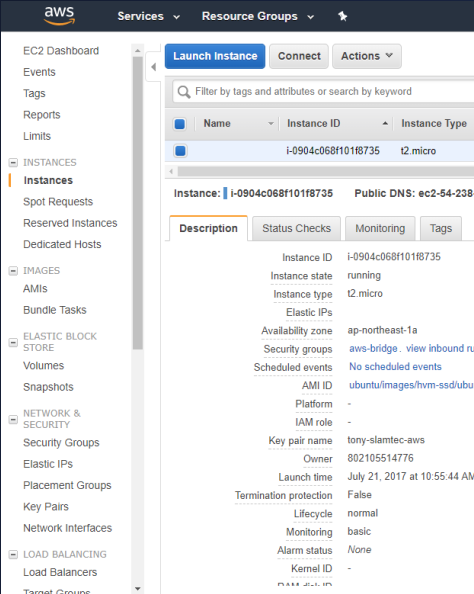
难以快速伸缩

laaS / 面向服务架构

计算

运维

困难



运维成本较高

难以快速伸缩

难以应对海量客户

虚拟服务器

购买虚拟服务器
远程重启服务器

PaaS / 微服务架构

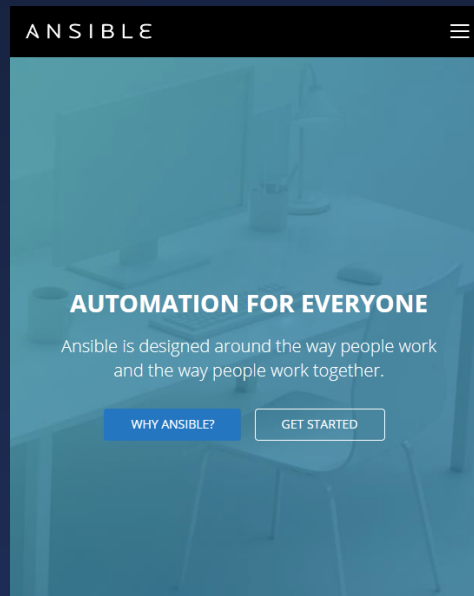
计算

运维

困难

```
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: tr-client-deployment
  namespace: test
spec:
  replicas: 3
  template:
    metadata:
      labels:
        app: tr-client
    spec:
      containers:
        - name: tr-client
          image: harbor.eng.slamtec.com/ton
          ports:
            - containerPort: 80
          resources:
            requests:
              cpu: 100m
              memory: 256Mi
            limits:
              cpu: 1000m
              memory: 512Mi
          imagePullSecrets:
            - name: regsecret
```

容器
虚拟服务



写程序自动买服务器
写程序自动重启服务器

DevOps很贵，而且招不到

维护一个容器编排集群成本高昂

难以实现秒级的伸缩

Serverless

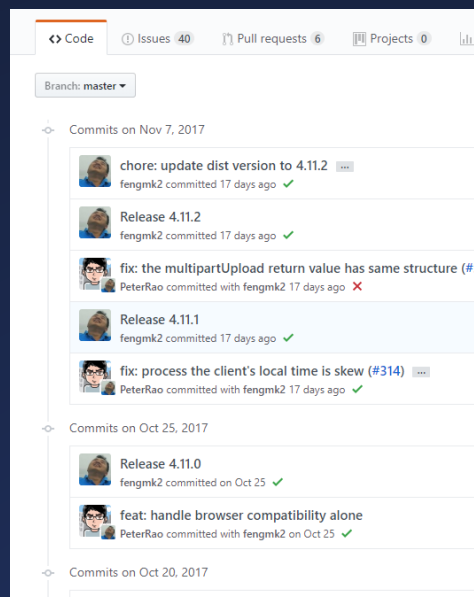
计算

运维

优势



FaaS



版本管理

无需运维

秒级伸缩

按需付费

常见家用机器人

扫地机器人



陪伴机器人



白家电智能化



常见机器人

商用机器人

迎宾机器人



导购机器人



送餐机器人



看护机器人

广告机器人

银行机器人

安防机器人

机器人需要具备的功能

机器人	特有功能	联网能力	自主移动	语音语义	图形UI	人脸识别
扫地机器人	清扫	✓	✓			
陪伴机器人	内容库	✓	✓	✓	✓	
智能白家电	家电功能	✓	✓			
迎宾机器人	身体动作	✓	✓	✓	✓	✓
导购机器人	地图内容管理	✓	✓	✓	✓	✓
送餐机器人	送餐托盘	✓	✓	✓	✓	
看护机器人	仪器互联	✓	✓	✓	✓	✓
广告机器人	大屏展示	✓	✓		✓	✓
银行机器人	银行设备	✓	✓	✓	✓	✓
安防机器人	摄影摄像	✓	✓			✓

机器人需要具备的功能



联网能力



语音语义



自主移动



图形UI



人脸识别

自主移动



我在哪里？

我要去哪儿？

我要怎么去？

自主移动

实时定位
自主建图



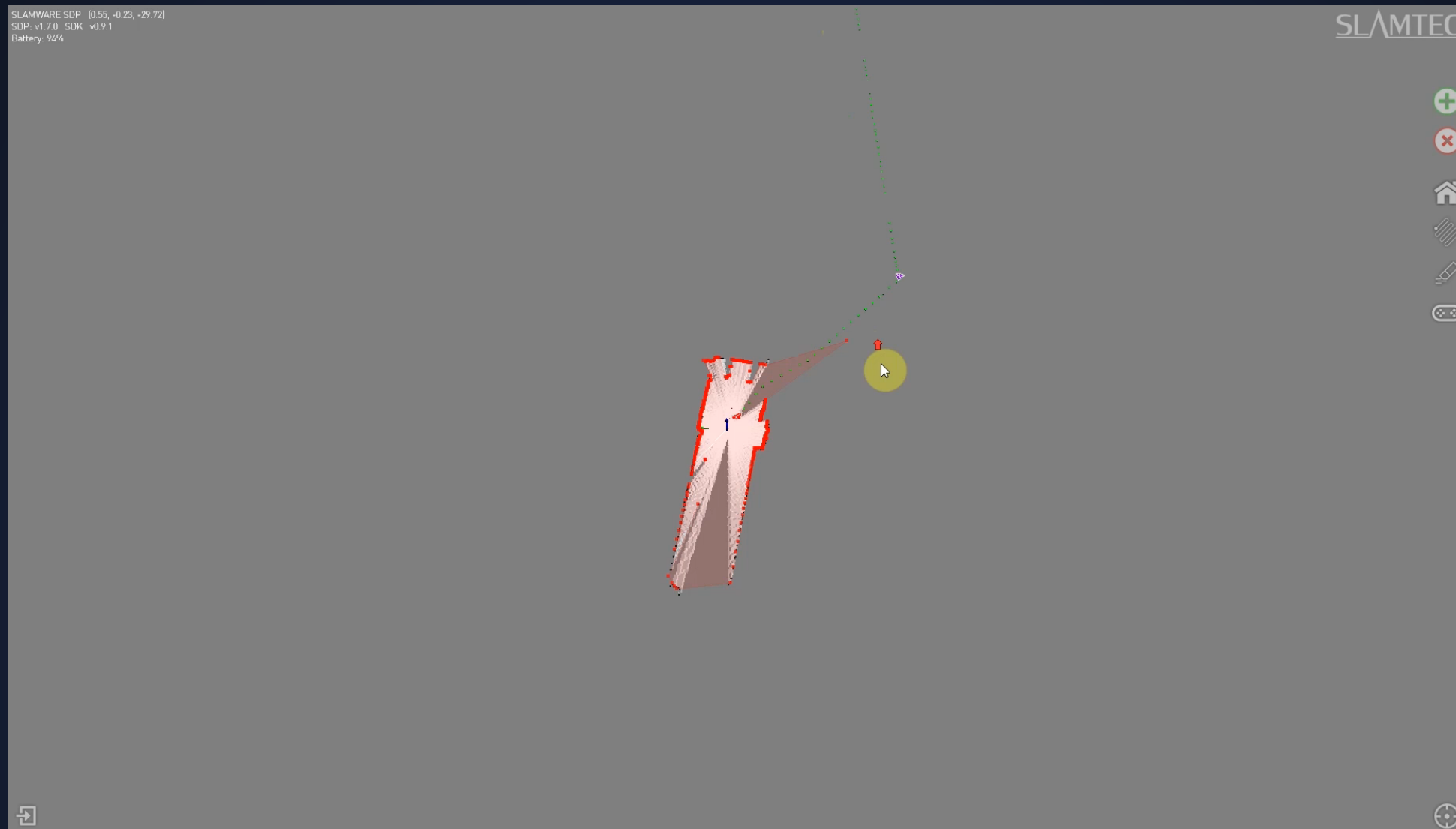
路径规划
运动执行

SLAM

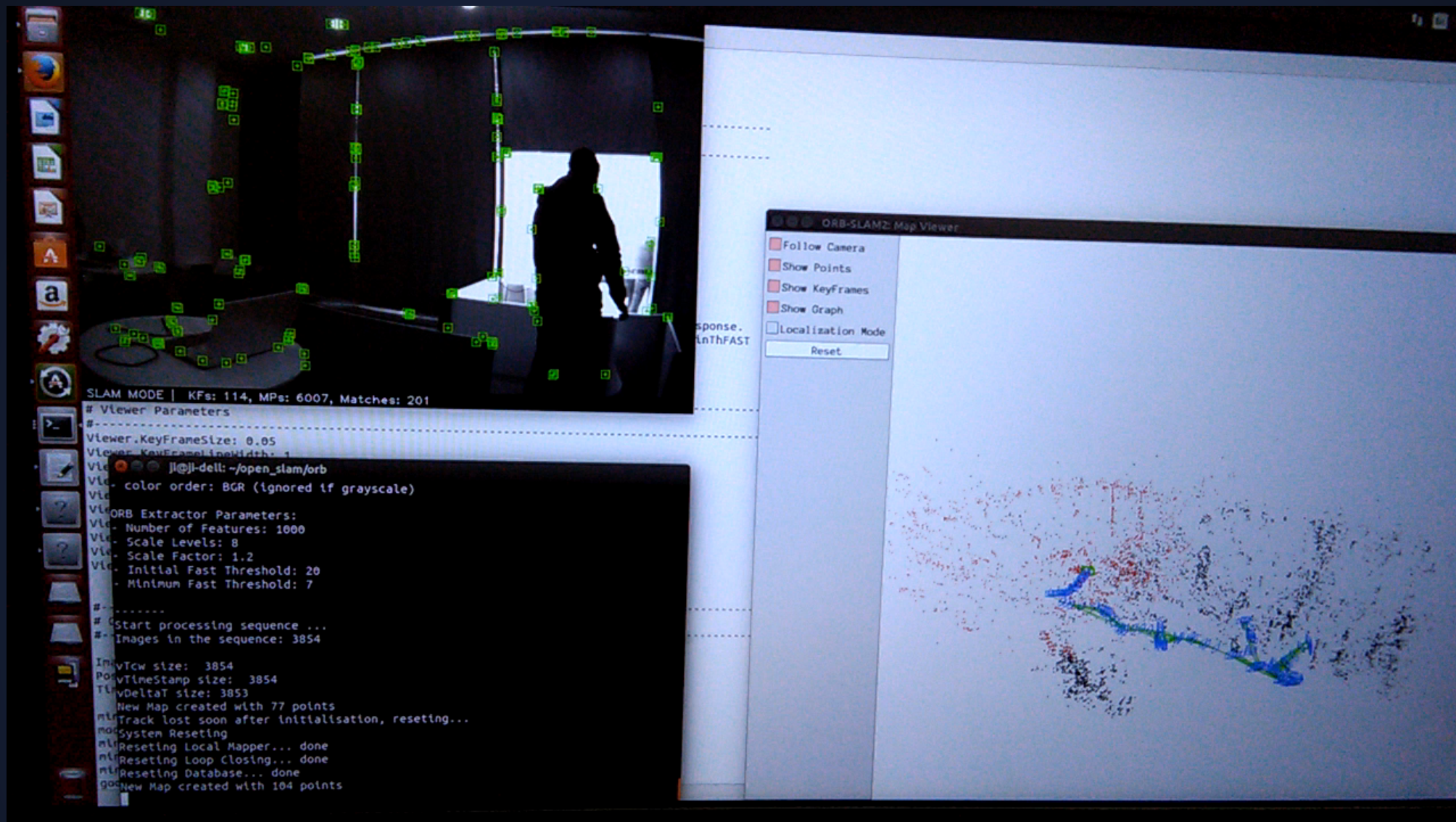
Simultaneous Localization And Mapping

同步定位与建图

LIDAR SLAM



vSLAM



路径规划和运动执行



路径规划

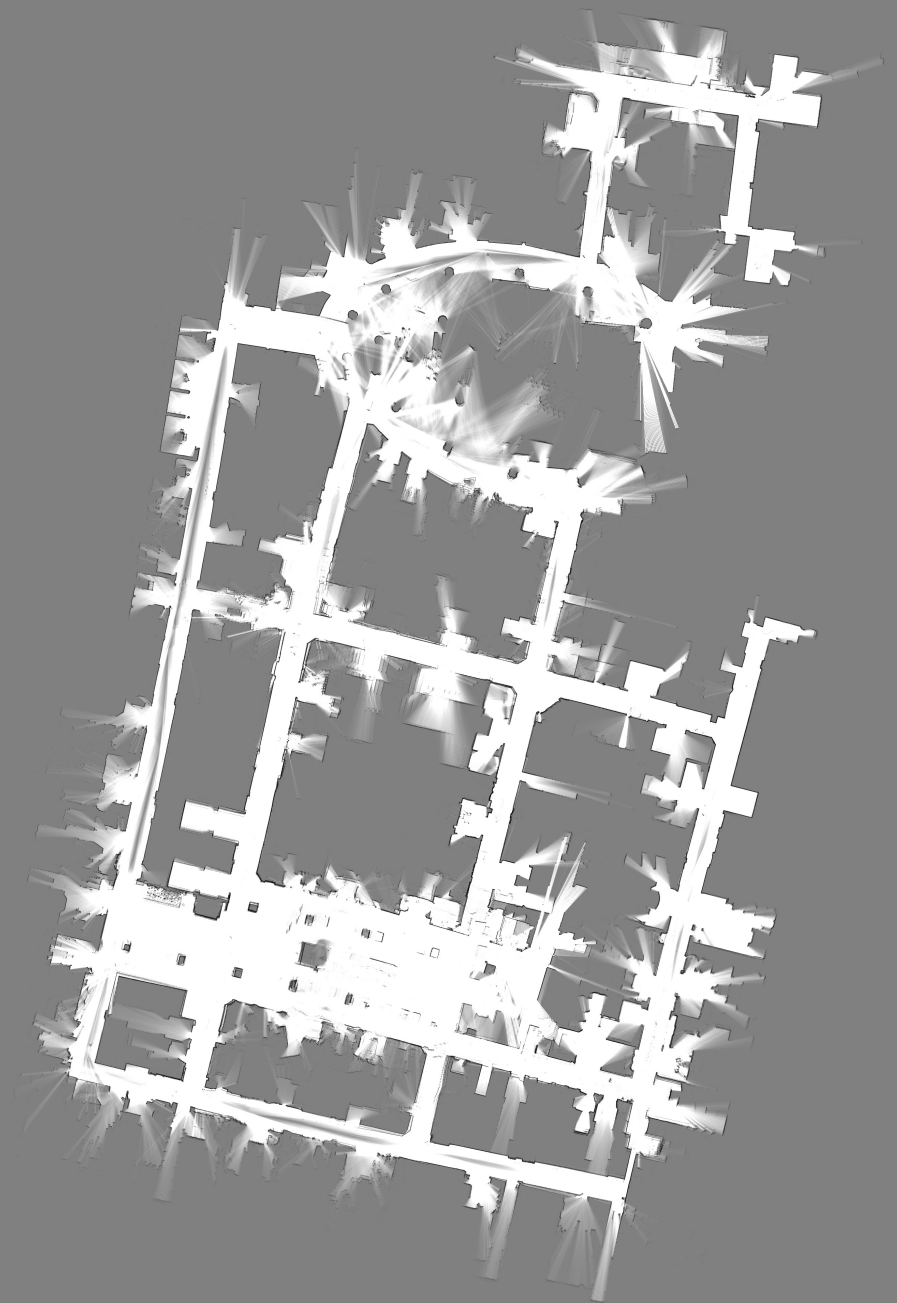


运动控制



多传感器融合





150m x 200m

云端图优化



STMS日志文件

二进制文件



数据预处理

C++程序



图优化程序

C++程序



地图与坐标

二进制和JSON文件

云端图优化

随机性

客户在现场实施时才会用到，无法预见流量

高负载

单次请求需要4 CPU Cores，2G以上的内存，且持续10分钟左右

无状态

单次请求跟前后请求都无关，为完全无状态的服务

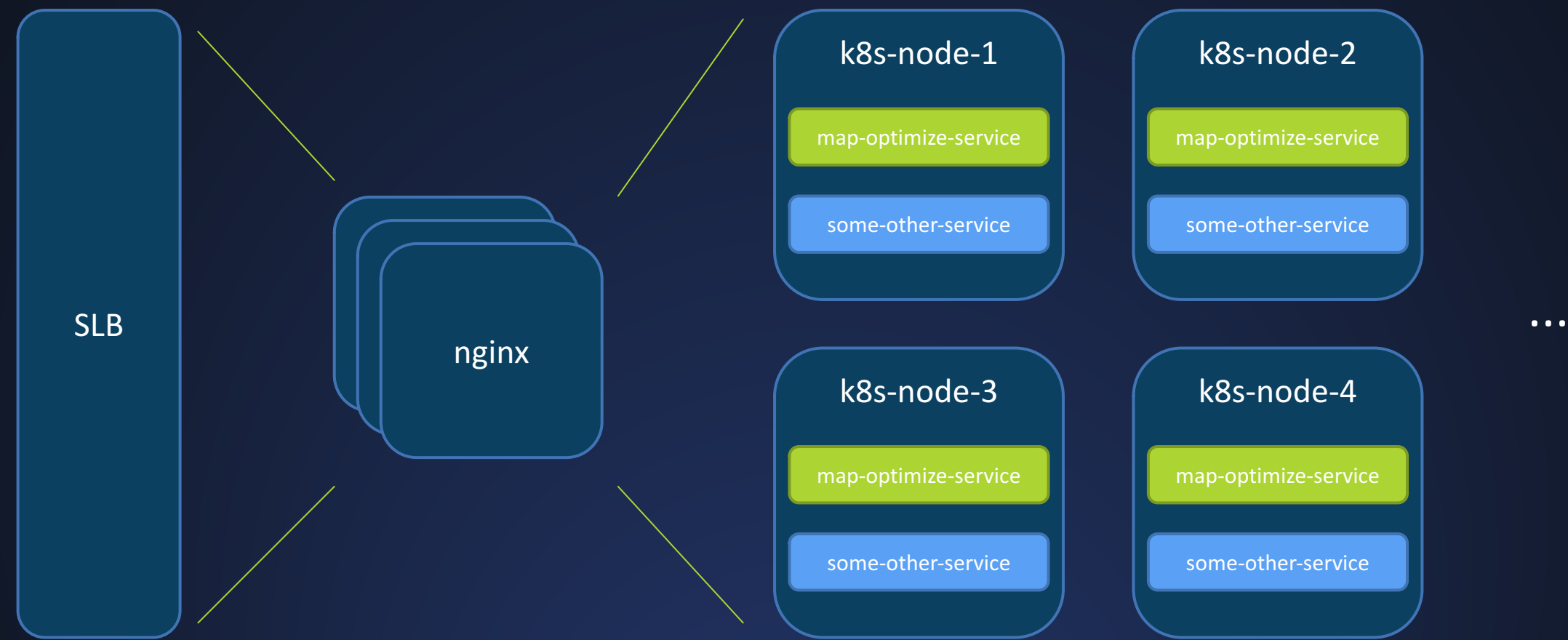
基于微服务架构的实现

```
from flask import Flask
import os

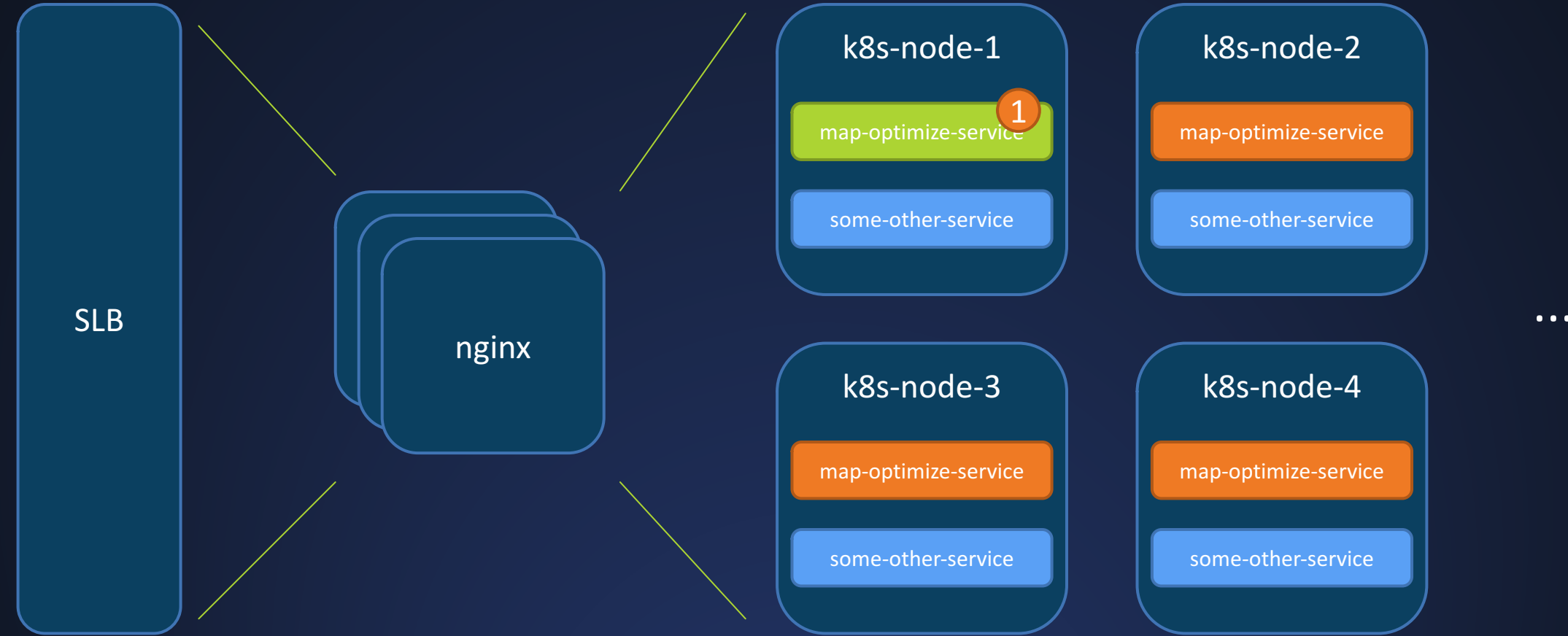
app = Flask('map-optimize-service')

@app.route('/jobs', methods=['POST'])
def create_job():
    filename = save_stms_log_to_temp()
    os.system('stms_ops preprocess "{}" -o "{}.processed"'.format(filename, filename))
    os.system('map_optimize -m output.map -p output.pose {}.processed'.format(filename))
    return result
```

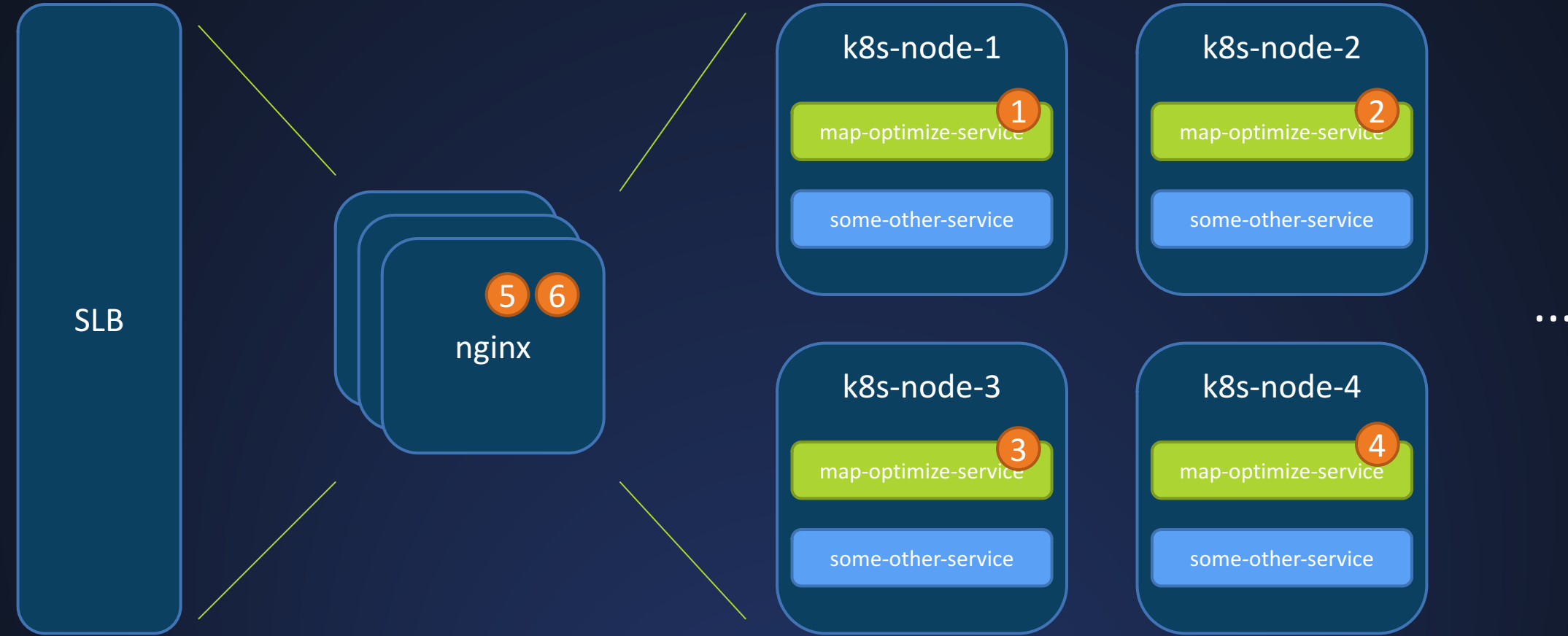
基于微服务架构的实现



基于微服务架构的实现



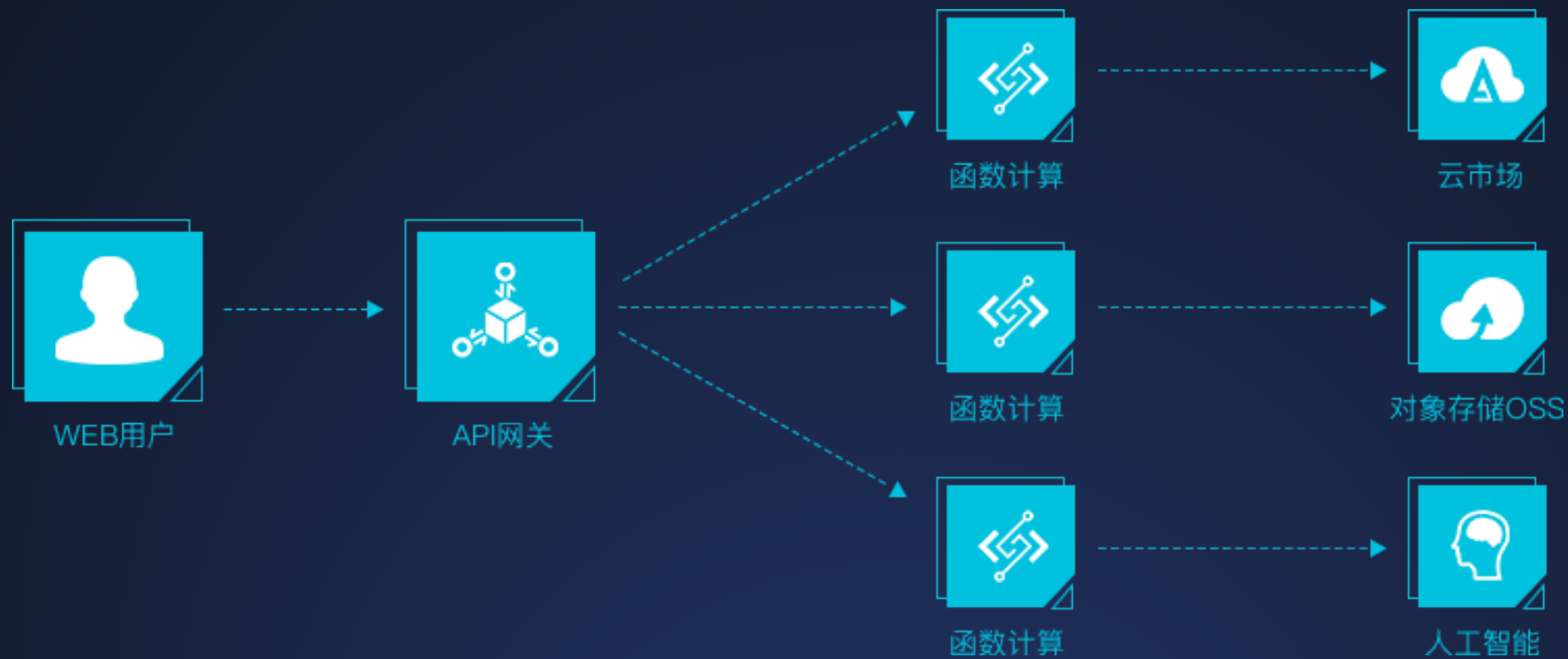
基于微服务架构的实现



Serverless/FaaS

- ▶ “**Serverless computing** is a cloud computing execution model in which the cloud provider dynamically manages the allocation of machine resources.”- Wikipedia
- ▶ “**Function as a Service (FaaS)** is a category of cloud computing services that provides a platform allowing customers to develop, run, and manage application functionalities without the complexity of building and maintaining the infrastructure. Building an application following this model is one way of achieving a "serverless" architecture, and is typically used when building microservices applications.” - Wikipedia

Serverless/FaaS 架构



事件驱动

- ▶ 客户调用了某个RESTful API
- ▶ RDS数据库中某一个表增加了一条记录
- ▶ OSS中有一个新的文件
- ▶ 弹性计算服务触发了一个告警

沙盒化运行

标准的可执行文件



沙盒环境

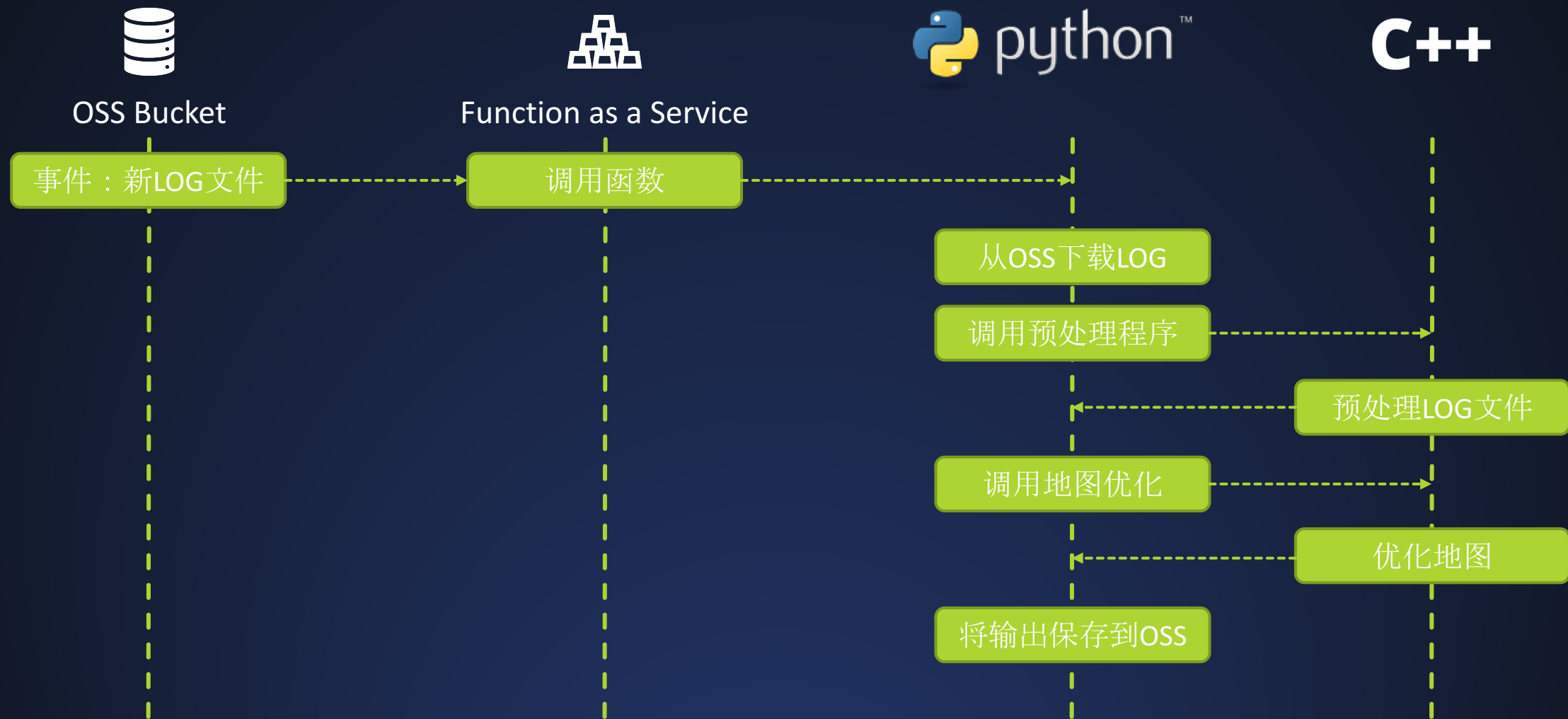
即用即弃

敏捷

秒级伸缩

按量计费

基于Serverless架构的实现



基于Serverless架构的实现

```
def handler(event, context):
    creds = context.credentials
    upevent = json.loads(event)
    objkey = upevent['events'][0]['oss']['object']['key']

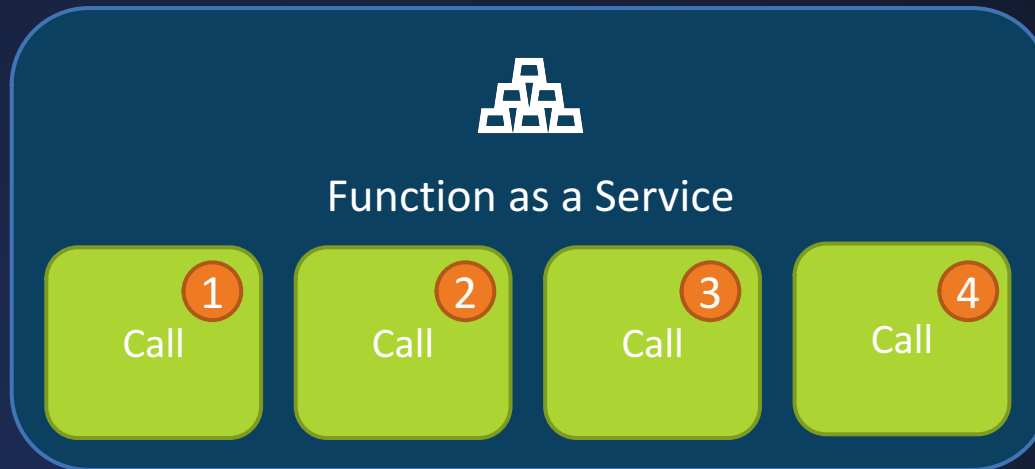
    auth = oss2.StsAuth(creds.accessKeyId, creds.accessKeySecret, creds.securityToken)
    bucket_in = oss2.Bucket(auth, endpoint, request_bucket)
    bucket_out = oss2.Bucket(auth, endpoint, response_bucket)

    bucket_in.get_object_to_file(objkey, save_path) # download
    os.system('stms_ops preprocess "{}" -o "{}.processed"'.format(save_path, save_path))
    os.system('map_optimize -m {}/output.map -p {}/output.pose {}.processed'.format(
        out_dir, out_dir, save_path))

    make_zip(out_dir, zip_file)
    bucket_out.put_object(outkey, zip_file) # upload result

    return objkey
```

基于Serverless架构的实现



更多的应用

- ▶ 基于云端的全局重定位
- ▶ 基于云端的大规模寻路

谢 谢