



SUMMIT  
SHENZHEN

物联网和移动技术分论坛

# 搭建无服务器架构的移动应用

黄卓斌

AWS 解决方案架构师

# 议程

什么是无服务器

快速搭建移动后台

利用 GraphQL 优化架构

移动应用开发工具

# 什么是无服务器

应用架构  
怎么应对服务器故障?

服务器被攻击?

怎么能够充分  
利用服务器资源

服务器应该运行  
哪种操作系统?

服务器还剩下多大的容量  
还能处理多大的流量?

怎么在服务器上实现动态配置变更?

什么时候应该扩容服务器?

预算有限  
我要选择哪种服务器?

怎么能保证操作系统  
及时打补丁?

应该部署哪个版本代码到服务器上?

# “服务器”

怎么控制服务器的访问权限?

应用怎么处理硬件故障?

有没更好的架构?

新版本代码怎么发布?

用户访问导致服务器压力过大?

谁能登录服务器?

怎么优化操作系统  
使得应用达到最优性能

我对性能要求  
多大的服务器合适?

我应该规划多少台服  
务器?

When should I decide to  
scale out my servers?

# 无服务器架构



无需维护  
服务器



灵活自动扩展



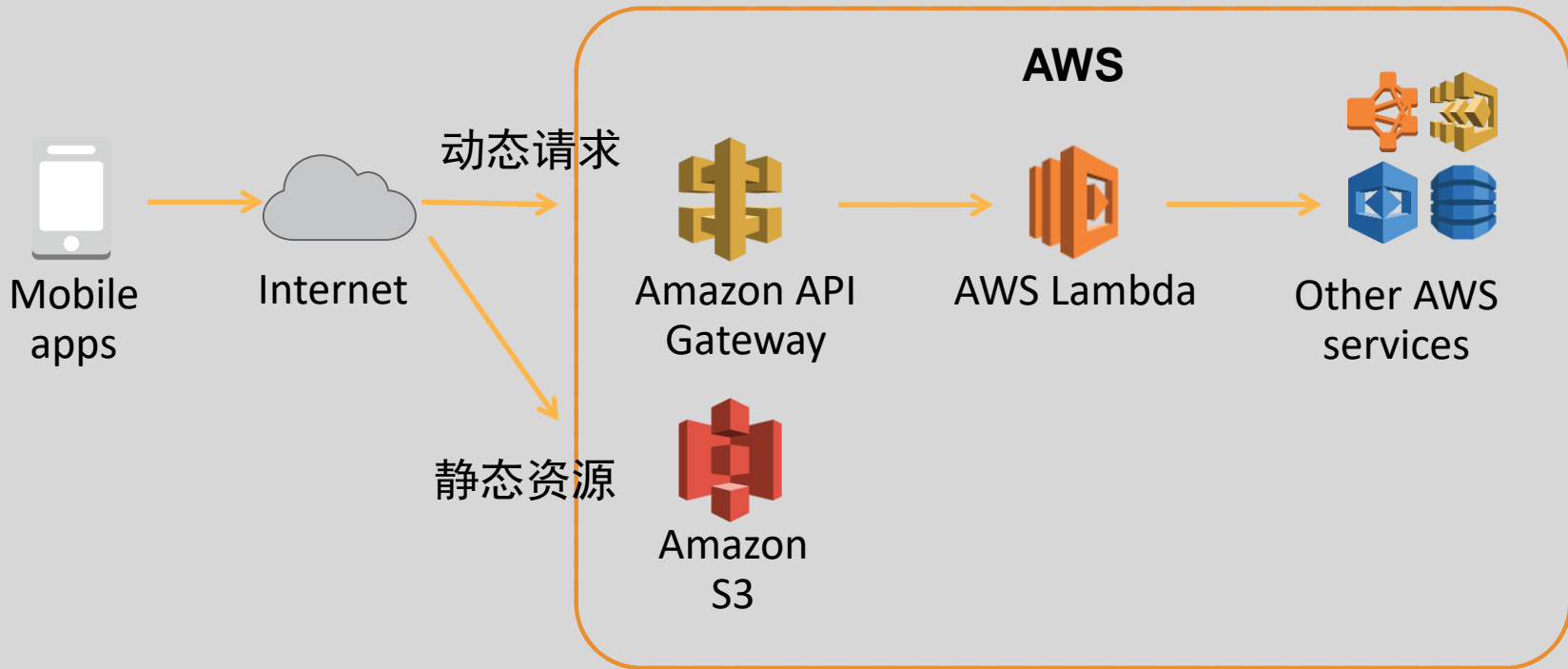
高可用



没有闲置的资源

# 快速搭建无服务器应用

# 无服务器移动后台架构 – 动静分离

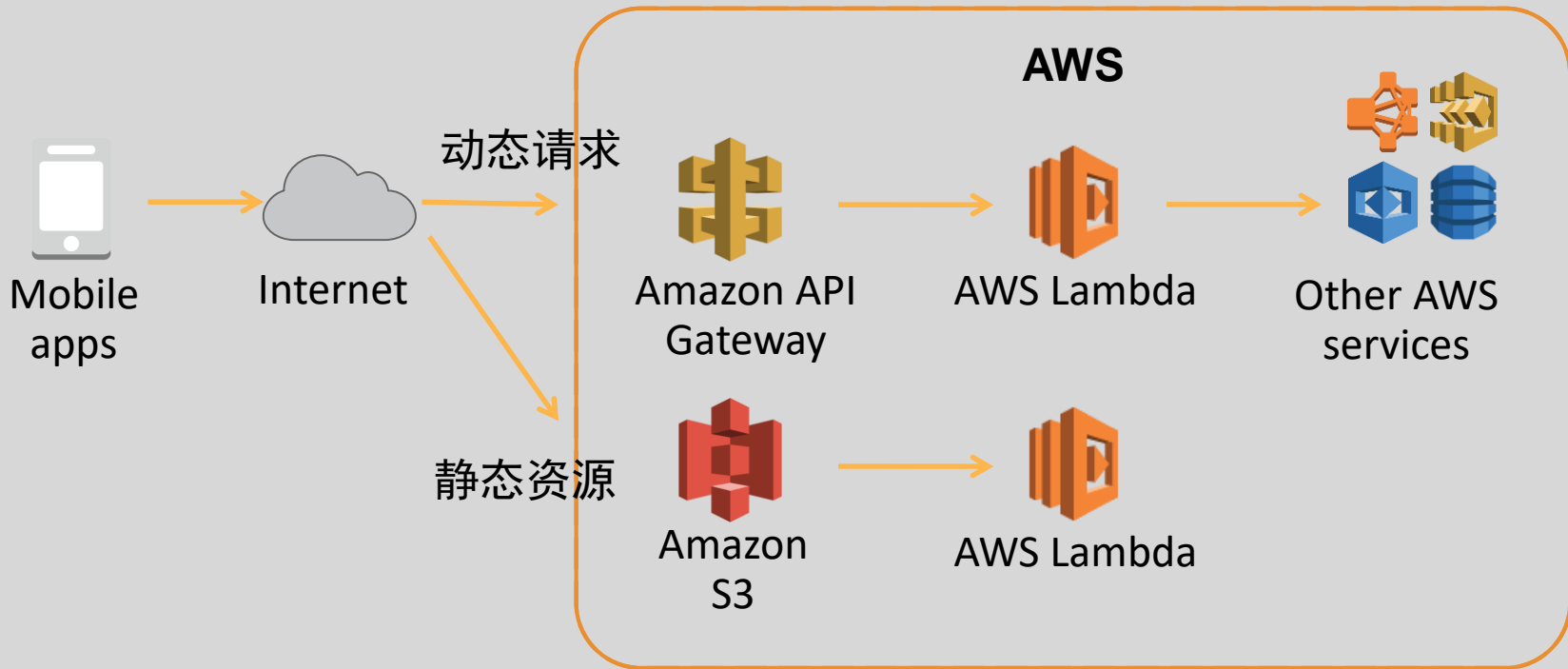


无需维护基础架构，只需要上传代码即可搭建无服务器Web应用

# 上传图片 想自动压缩?



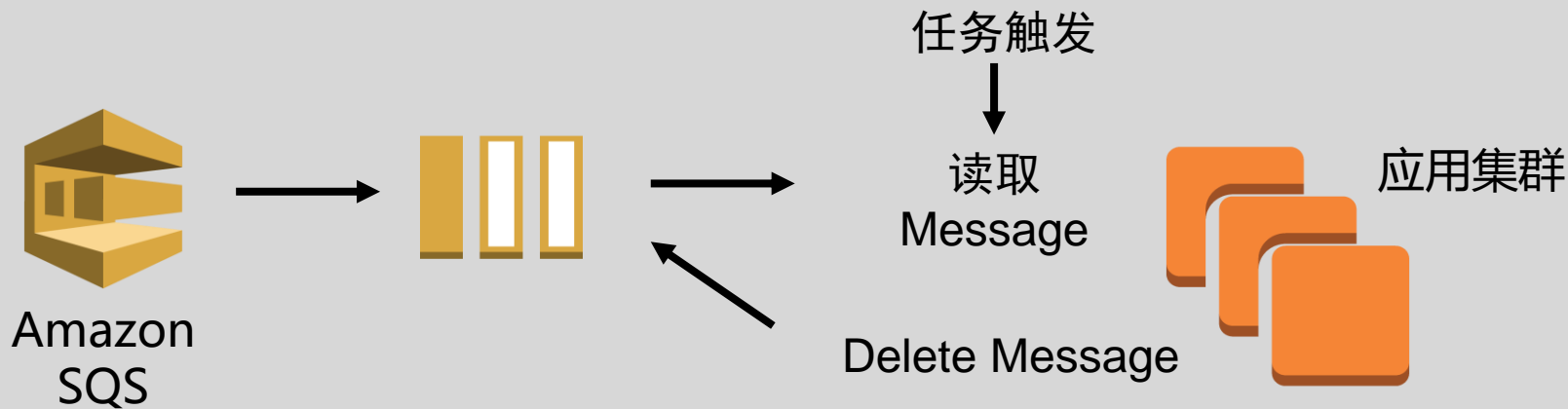
# 无服务器移动后台架构 – 自动触发



上传图片到 Amazon S3, 自动触发 AWS Lambda 完成图片压缩

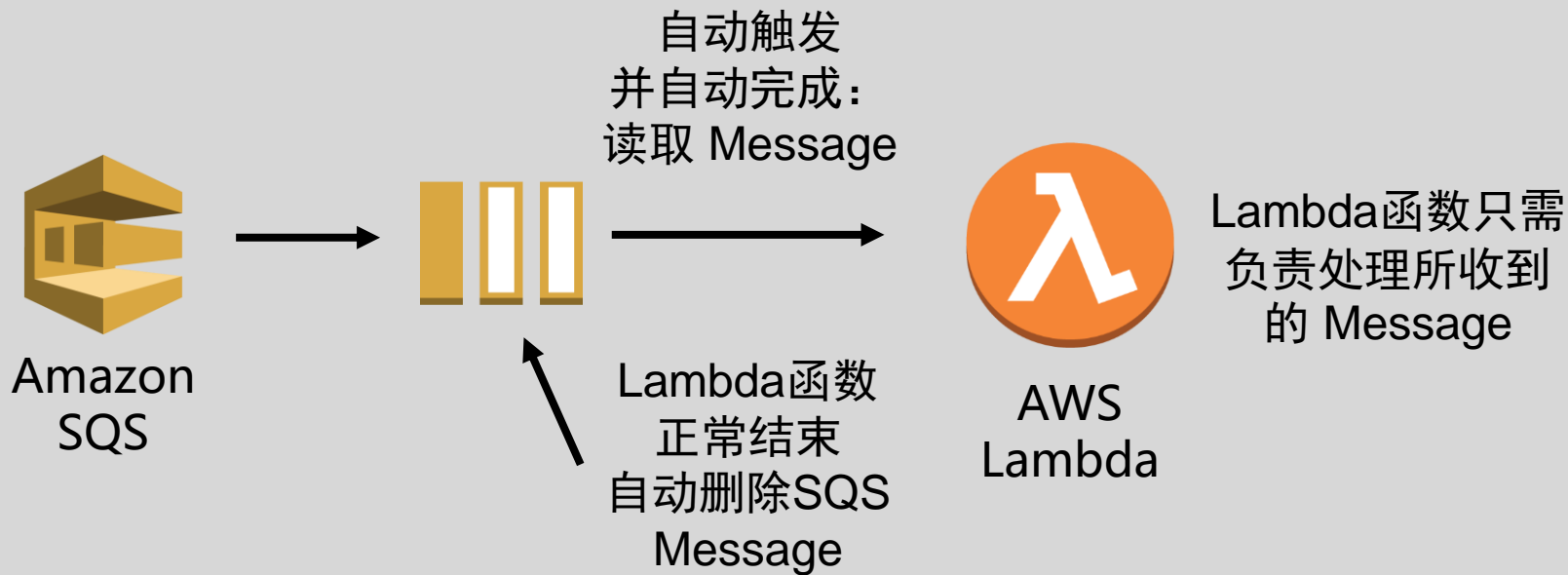
# 消息队列的消费？

# 处理消息队列的方式 – 传统模式



监控堆积的消息数量  
增减应用服务器数量

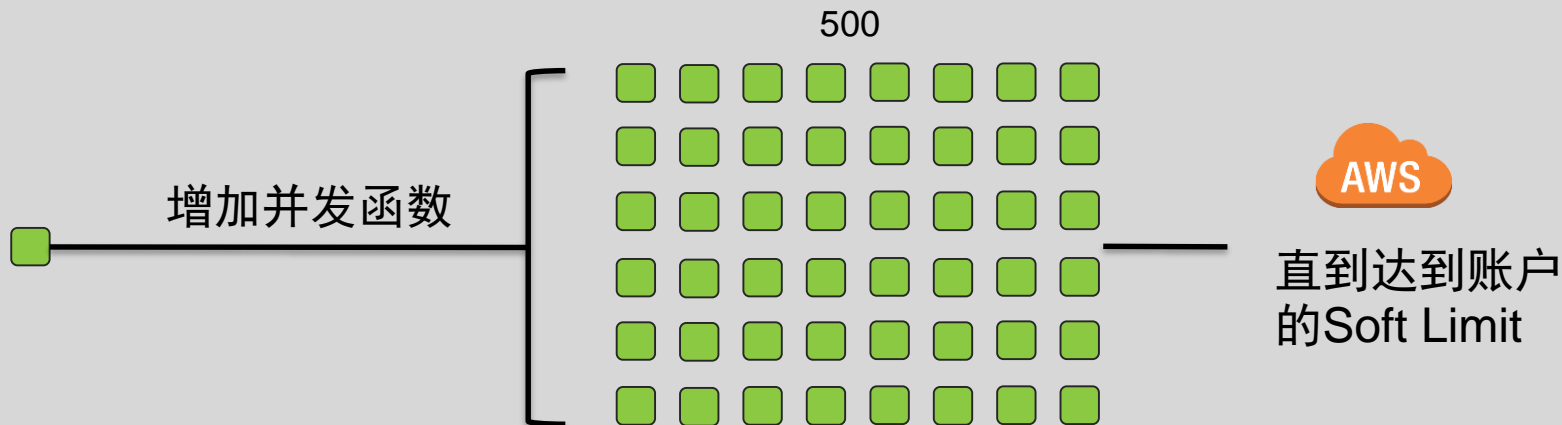
# 无服务器移动后台架构 – 消息队列消费 NEW



# AWS Lambda 并发?

# AWS Lambda 并发

AWS Lambda 根据增加的流量动态扩展容量  
突增流量，Lambda 将立即根据预定量增加您的并发执行函数。

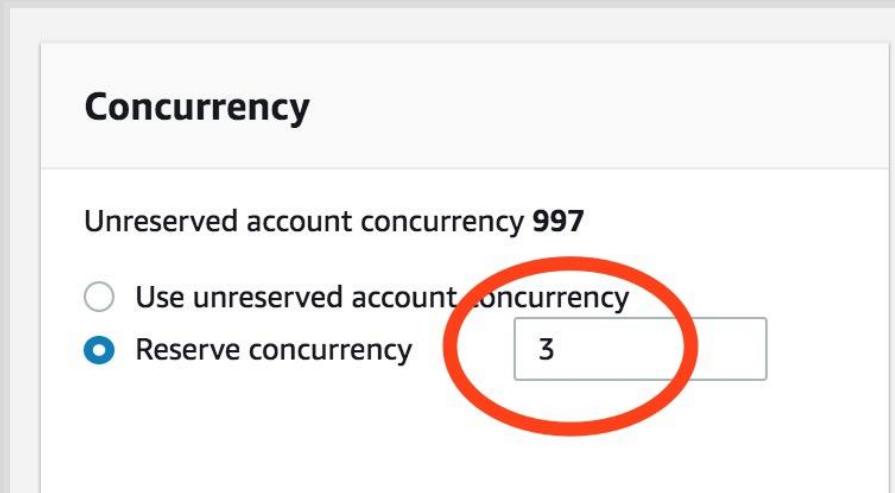


THINK 并发, NOT TPS

# AWS Lambda 并发控制设置

针对单个 AWS Lambda 函数

1. 预留
2. 封顶
3. 停用



The screenshot shows the 'Concurrency' settings for an AWS Lambda function. It displays 'Unreserved account concurrency 997'. There are two radio button options: 'Use unreserved account concurrency' (unselected) and 'Reserve concurrency' (selected). A red circle highlights the 'Reserve concurrency' option and the adjacent input field which contains the number '3'.

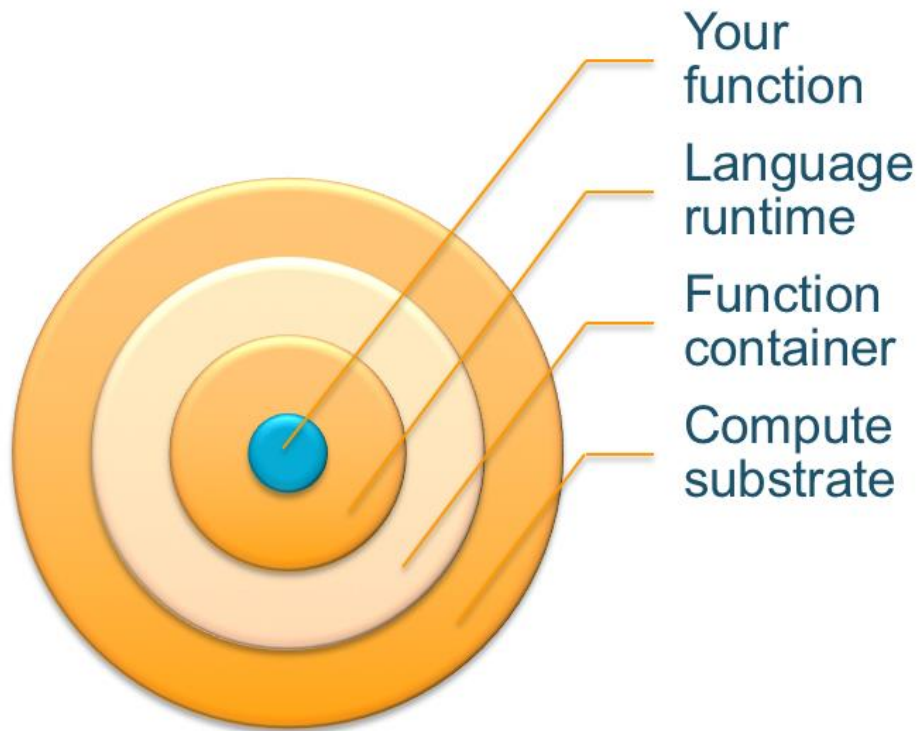
**Concurrency**

Unreserved account concurrency **997**

☐ Use unreserved account concurrency

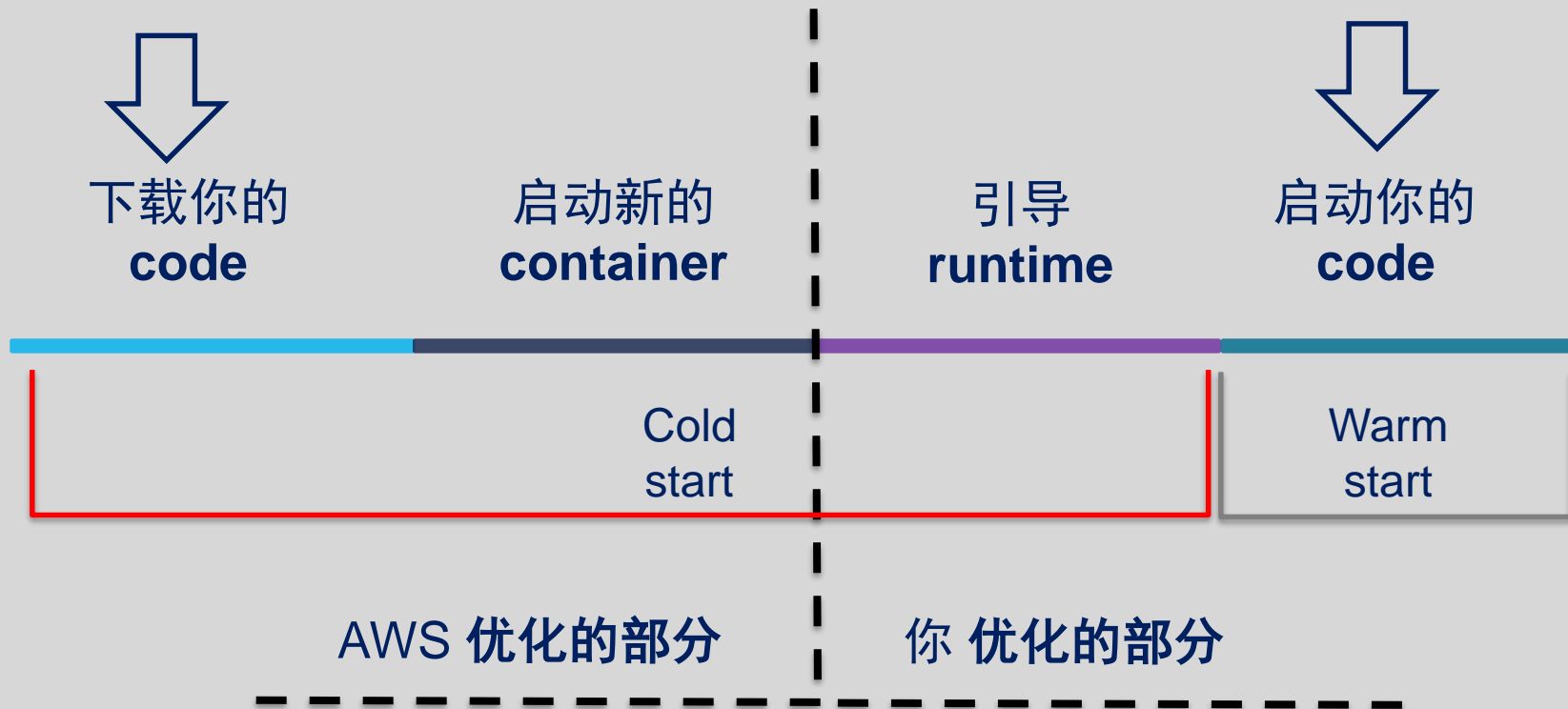
☒ Reserve concurrency

# AWS Lambda 原理





# AWS Lambda 的启动



# 利用 AWS Lambda 容器重用机制

```
var AWS = require('aws-sdk');  
var uuid = require('node-uuid');  
var db;
```

} 容器生成的时候执行  
在容器重用，即Lambda Function第  
二次调用时不再执行

Lambda  
Function  
函数被调  
用时执行

```
exports.myHandler = function(event, context, callback) {  
    if (!db || db.isConnected()) {  
        db = setupConnection();  
    }  
    // Rest of logic  
}
```

P99情况不会遇到冷启动

如果经测试，冷启动时间较长，影响用户体验，可以进行定时预热，特别是 Java 和 C# 的应用

可以利用 CloudWatch 定时触发预热的 Lambda，利用诸如 Node.js 的非阻塞机制进行并发预热容器

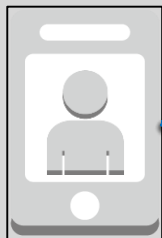
# 优化架构

# 设定一个简单场景

一个活动 Event 发布和评论 Comment 的应用

- 可发布新活动通知
- 用户可以进行评论
- 用户可关注某个活动，实时收到更新的评论

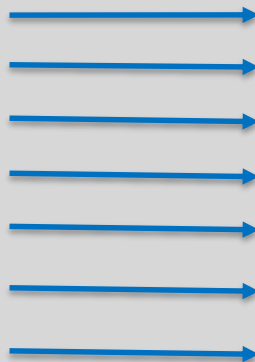
# 当前基于 REST API 的应用设计



- ✓ 碎片化的建链
- ✓ 标准 HTTP Calls

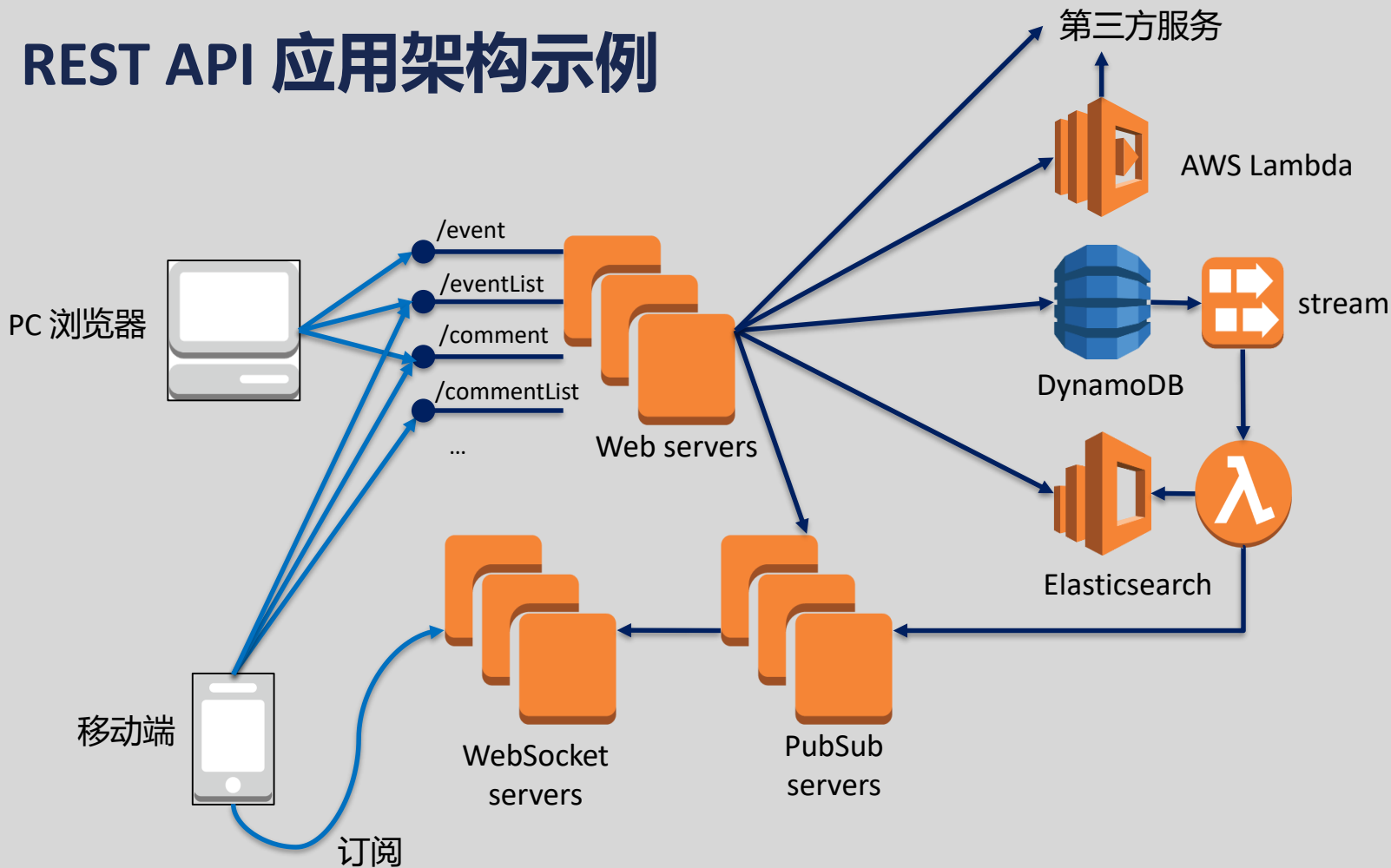
## REST Endpoints

POST /event  
POST /comment  
GET /eventList  
GET /eventDetail  
GET /commentList  
GET /comment  
POST /eventDelete

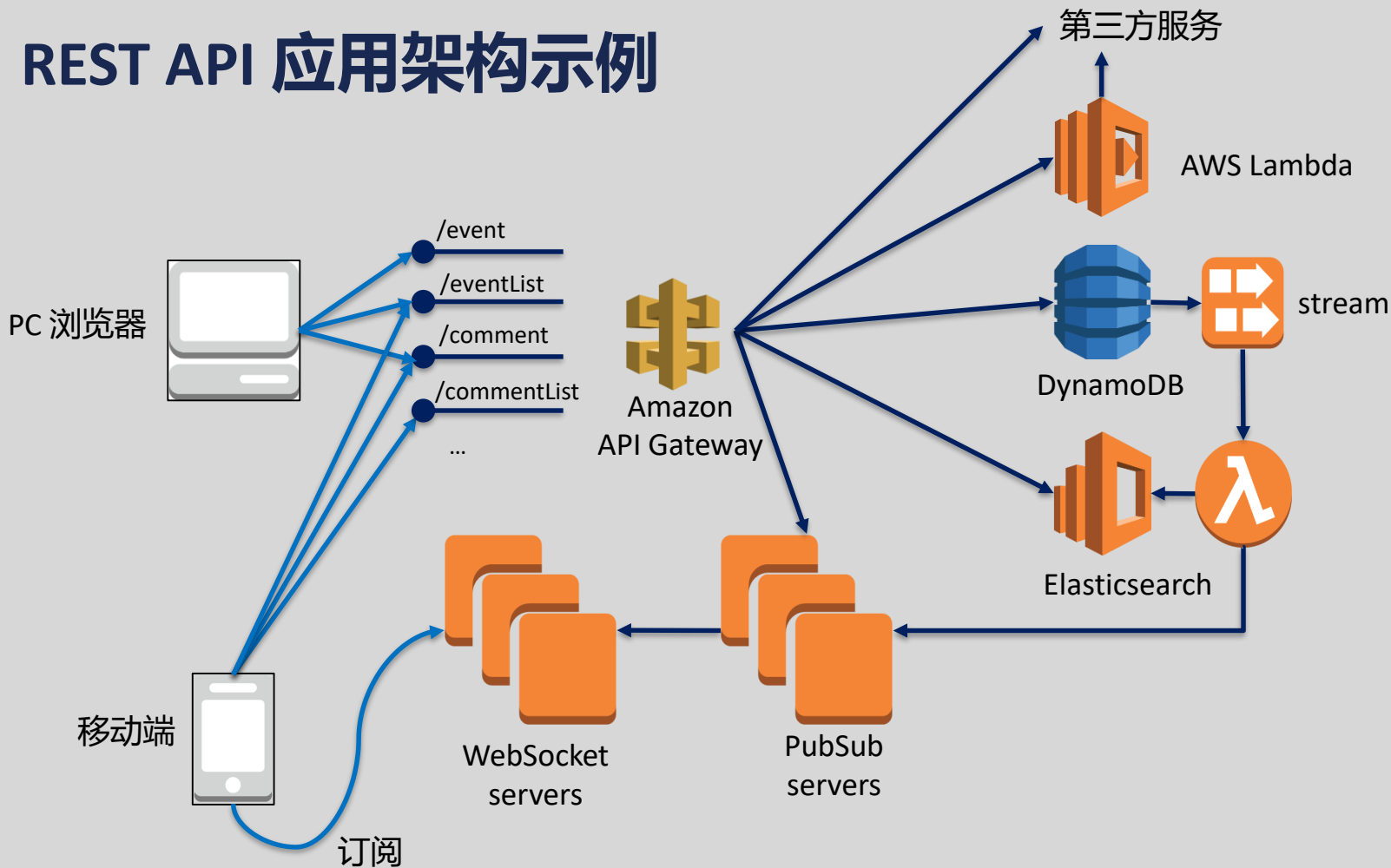


- ✗ API 关联关系
- ✗ 返回结果冗余信息
- ✗ 多数据源支持
- ✗ 排序和分页
- ✗ 订阅实时通知

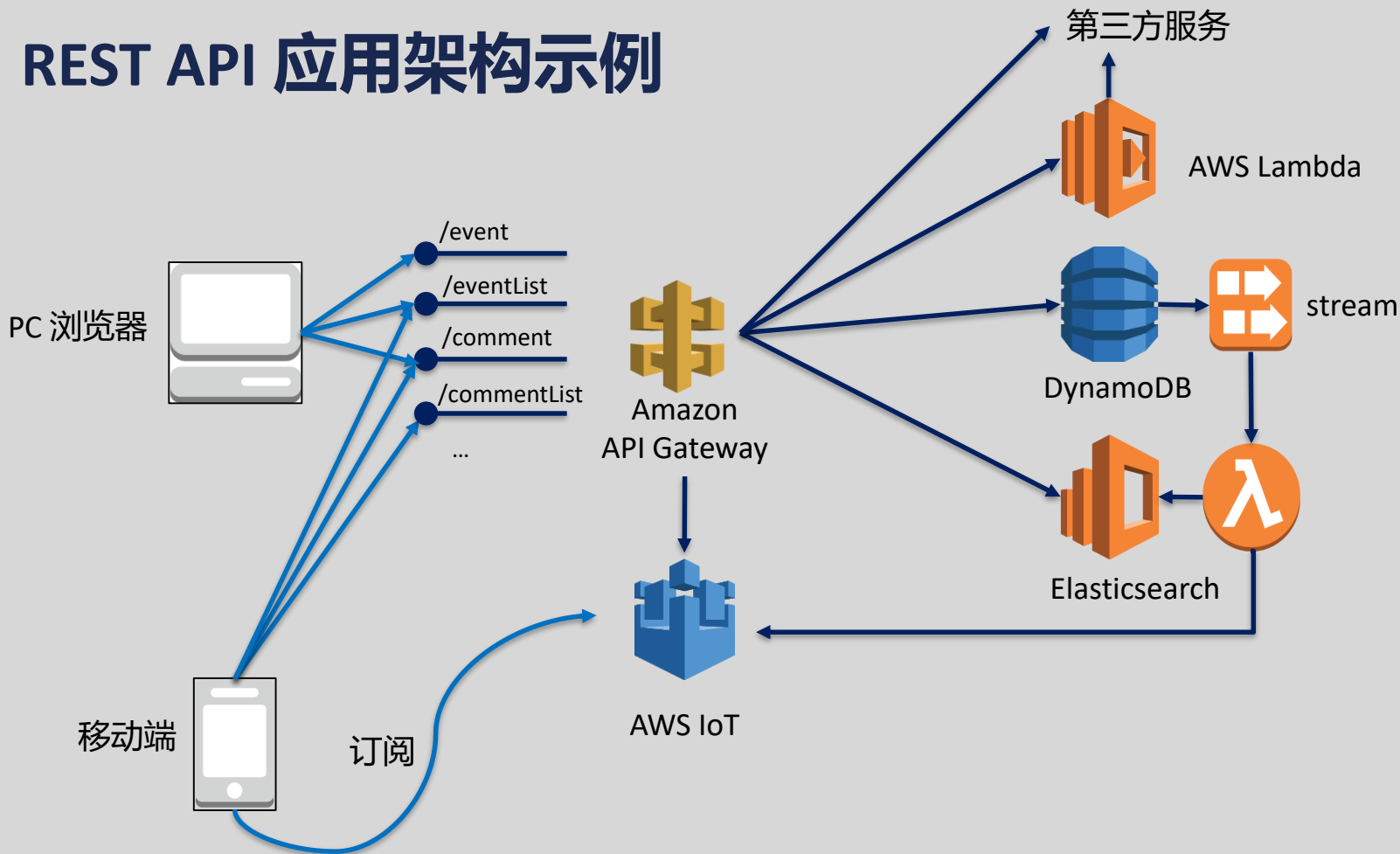
# REST API 应用架构示例



# REST API 应用架构示例

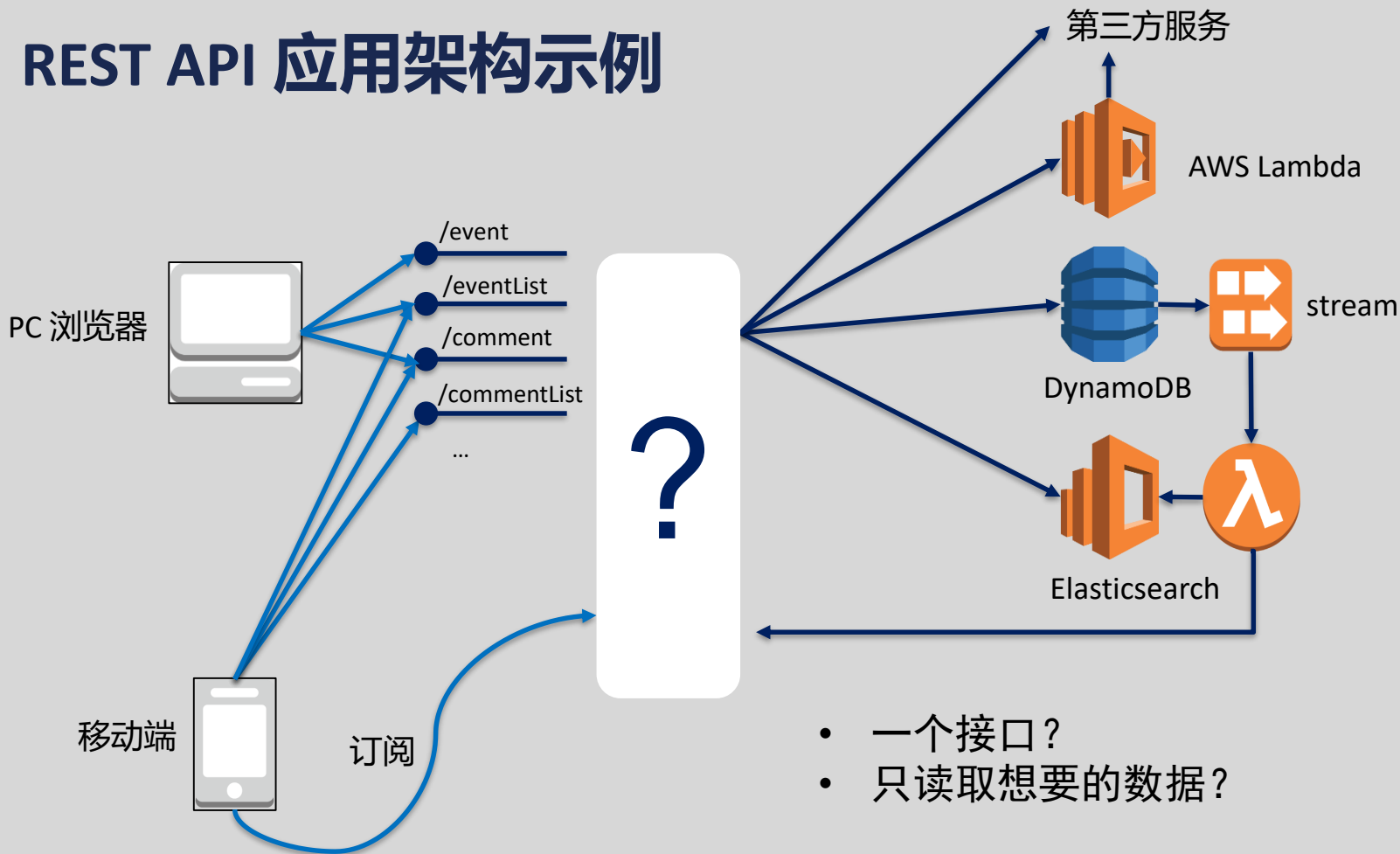


# REST API 应用架构示例





# REST API 应用架构示例



- 一个接口?
- 只读取想要的数据?

# GraphQL

## 为 API 而生的查询语言

**非常适合客户端 App 操作数据**

# 谁在使用 GraphQL 语言

Facebook 的移动应用从 2012 年就开始使用 GraphQL。  
GraphQL 规范于 2015 年开源，被各种体量的团队所使用。



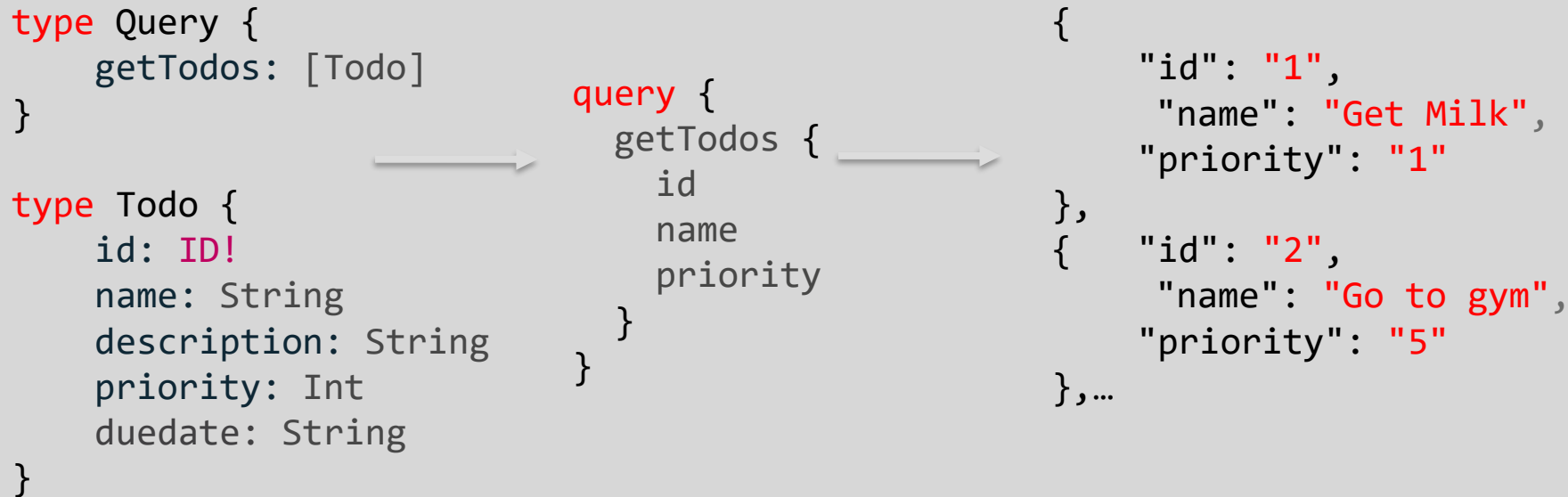
# GraphQL - 数据操作语言

Query 读数据

Mutation 写数据然后读数据

Subscription 订阅实时推送的数据

# GraphQL - 工作方式



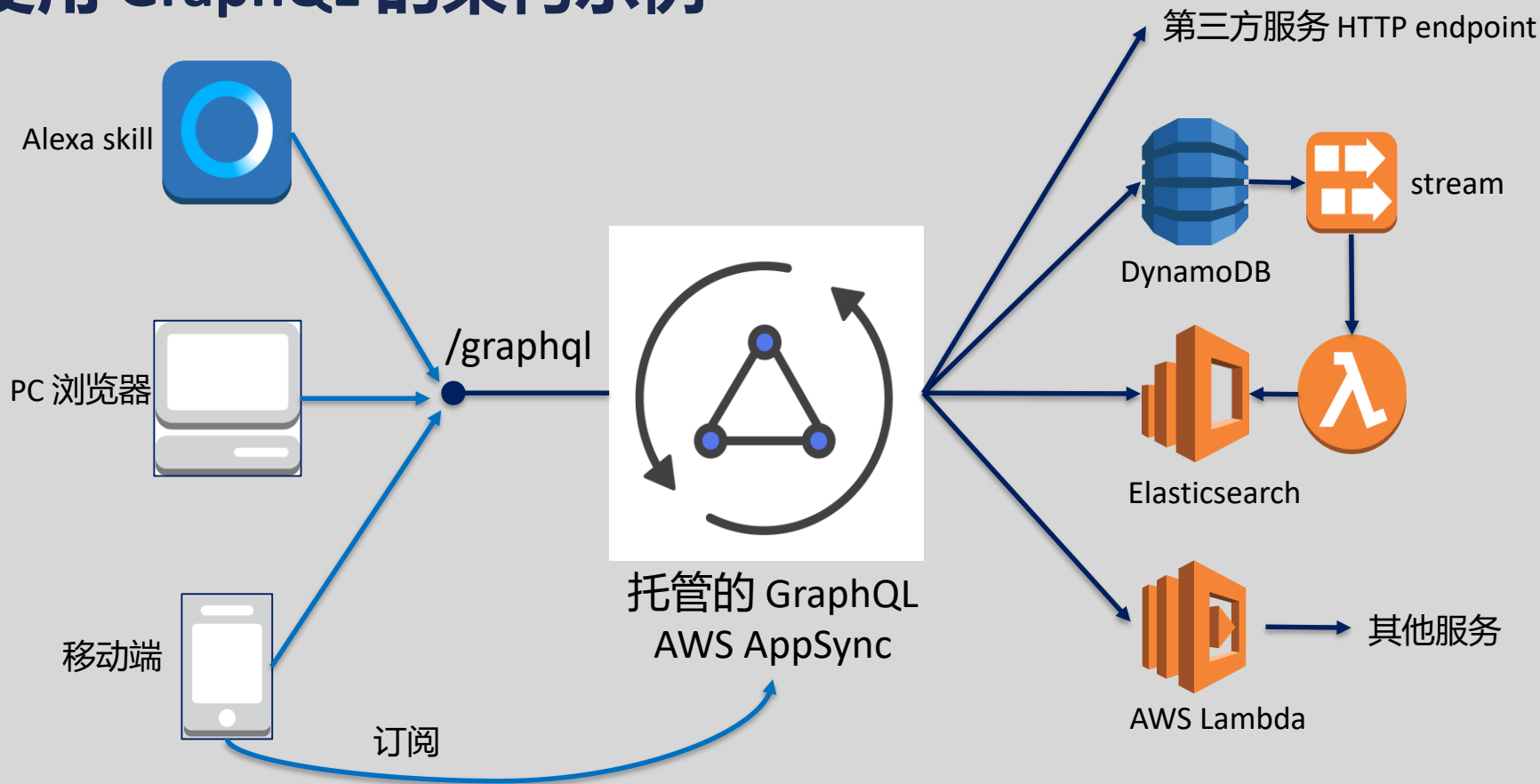
定义数据schema

查询query

获得数据data

- 一个 GraphQL 请求就能准确获得你想要的数据库，不多不少。
- 一个请求获取多个资源，轻松应对比较慢的移动网络
- API 演进无需划分版本

# 使用 GraphQL 的架构示例



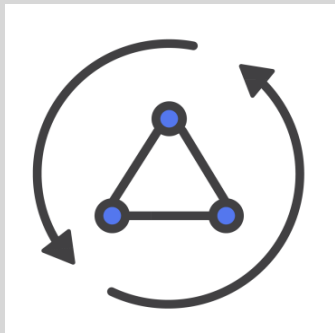
# AWS AppSync 构建强大的数据驱动的APPS



**实时, 协作  
Apps**



**离线编程模型  
同步机制**



**只收取你需要的  
数据**



**访问多个数据源**



**精细的权限控制**

# AWS AppSync 演示





Services ▾

Resource Groups ▾



VPC



EC2



Oregon ▾

Support ▾



SUMMIT  
SHENZHEN

AWS AppSync



APIs

Event Demo App

Schema

Queries

Data Sources

Settings

AWS AppSync > Event Demo App > Schema

## Schema

Design your schema using GraphQL SDL, attach resolvers, and quickly deploy AWS resources. [Info](#)

Create Resources

Undo Edits

Save Schema

### Schema

Export schema ▾

```
1 type Comment {  
2   # The id of the comment's parent event.  
3   eventId: ID!  
4   # A unique identifier for the comment.  
5   commentId: String!  
6   # The comment's content.  
7   content: String!  
8   # The comment timestamp. This field is indexed to enable  
9   createdAt: String!  
10 }  
11  
12 type CommentConnection {  
13   items: [Comment]
```

### Resolvers

Filter types...

### Comment

Field

Resolver

eventId: ID!

Attach

commentId: String!

Attach



Feedback



English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.

[Privacy Policy](#)

[Terms of Use](#)



Services ▾

Resource Groups ▾



VPC



EC2



Oregon ▾

Support ▾

AWS AppSync > Event Demo App > Queries

## Queries

Write, validate, and test GraphQL queries. [Info](#)



```

14   items{
15     id
16     name
17   }
18 }
19 }
20 |
21 query q1{
22   getEvent(id: "0a79f950-e807-45f9-a6d0-8
23     name
24     when
25     where
26   }
27 }
```

QUERY VARIABLES

```

{
  "data": {
    "getEvent": {
      "name": "helloevent",
      "when": "friday",
      "where": "office"
    }
  }
}
```

LOGS

### Documentation Explorer



Search Schema...

A GraphQL schema provides a root type for each kind of operation.

#### ROOT TYPES

query: **Query**

mutation: **Mutation**

subscription: **Subscription**


Chrome File Edit View History Bookmarks People Window Help

DynamoDB · AWS Console x AWS AppSync Console x React App x

localhost:3000

Apps Com GTech AWS Imported learn new&console Larchfield - Bushmi...

## AppSync Demo - 活动列表 -



Create new event

### 开发者聚会

May 25, 2018

2:00 PM

Chime

Presentators: XueFeng, HuangZhuobin

0 comments

Delete

r-HZB.pptx

Search in Presentation

Share

Convert to SmartArt Picture Shapes Text Box Arrange Quick Styles Shape Fill Shape Outline

aws | Webinars

Notes Comments 90%





Services

Resource Groups



huangzb @ huangzb

Oregon

Support



AWS AppSync > AppSyncDemo-GCRTrainingTalk > Queries

## Queries

Write, validate, and test GraphQL queries. [Info](#)

```
24 query quEvent{
25   getEvent(id: "6e9e9b2b-c46c-40c2-b664-44fd5c5a2a1f"
26     name
27     where
28   }
29 }
30
31
32 mutation comments{
33   commentOnEvent(
34     eventId: "6e9e9b2b-c46c-40c2-b664-44fd5c5a2a1f"
35     content: "I like it"
36     createdAt: "2018-07-10T14:30:00+08:00"
37   ){
38     commentId
39     eventId
40   }
41 }
```

**< Mutation**

**Comment**

No Description

FIELDS

**eventId: ID!**  
The id of the comment's parent event.

**commentId: String!**  
A unique identifier for the comment.

**content: String!**  
The comment's content.

**createdAt: String!**  
The comment timestamp. This field is indexed to enable sorted pagination.

# AWS AppSync 企业级安全性和精细访问控制

## AWS AppSync 认证方式

- API Key
- AWS IAM (Identity and Access Management) 使用角色来控制
- Amazon Cognito 用户池，集成了多种社交服务提供商 (Facebook、Google+ 和 Login with Amazon)，并支持使用 SAML 联合身份。
- OpenID Connect

# AWS AppSync 企业级安全性和精细访问控制 (续)

## 读数据鉴权的示例

ID	Data	PeopleCanAccess	GroupsCanAccess	Owner
123	{my: data,...}	[Mary, Joe]	[Admins, Editors]	Nadia

- ✓ 数据Owner可读
- ✓ 多用户可读
- ✓ 用户组可读
- ✓ 过滤结果子集
- ✓ 只允许特定访问

基于GraphQL Resolver 实现数据访问控制 (例子: Owner可读)

```
#if($context.result["Owner"] == $context.identity.username)
    $utils.toJson($context.result);
#else
    $utils.unauthorized()
#end
```

参考文档: <https://docs.aws.amazon.com/appsync/latest/devguide/security-authorization-use-cases.html>

# AWS AppSync 可以构建的 APP 示例



## 协作类APP

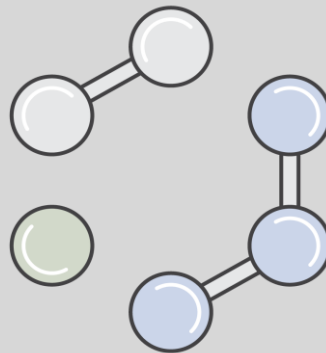
- 仪表盘
- 排行榜
- 白板
- ...



## 社交、聊天和 约会APP



## 基于地理位置的 APP



## 复杂的数据结构和 类型的APP

AWS AppSync 示例代码与文档资源 [amazonaws-china.com/appsyntax/resources/](https://amazonaws-china.com/appsyntax/resources/)

# 开发工具



# AWS Amplify 开发包



- 针对前端和移动端开发的 JavaScript Library
  - JavaScript (frameworks)
  - React + React Native
- 功能
  - Authentication
  - Analytics, APP用户行为分析
  - API, 支持 AWS Signature Version 4
  - GraphQL Client
  - Storage, 管理 Amazon S3 存储数据
  - Push Notifications, Amazon Pinpoint
  - Interactions, Amazon LEX
  - PubSub
  - Internationalization, 语言国际化方案
  - Cache, LRU(Least Recently Used) Cache

*<https://aws-amplify.github.io/>*

• AWS Amplify 开发包与 Apollo GraphQL Client 集成离线缓存 ([github.com/apollographql](https://github.com/apollographql))

# AWS Amplify 前端访问 GraphQL 示例代码

```
import Amplify, { API, graphqlOperation } from "aws-amplify";  
import aws_config from "./aws-exports";  
Amplify.configure(aws_config);
```

```
const ListEvents = 'query ListEvents {  
  listEvents {  
    items {  
      id  
      name  
    }  
  }  
}';
```

就这么简单就完成了

```
const allEvents = await API.graphql(graphqlOperation(ListEvents));
```

## AWS AppSync

APIs

My AppSync App

JS AppSync.js ×

Queries

Data S

Settings

```
1 export default {  
2   "graphqlEndpoint": "https://dufipijbuffcnc7vjrrh5qpb3i.appsync-api.us-west-2.amazonaws.com/graphql",  
3   "region": "us-west-2",  
4   "authenticationType": "API_KEY",  
5   "apiKey": "da2-dq7hb3vlezd6dnb4grxygkfsha"  
6 }
```

Endpoint端点和  
认证方式等配置



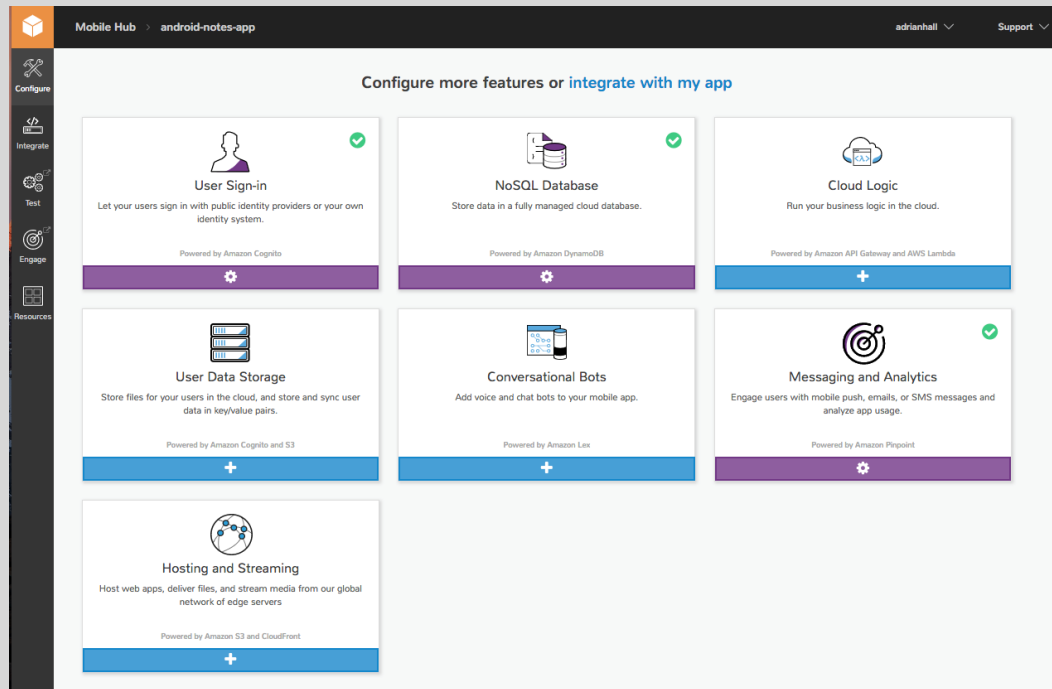
2. Download the AWS AppSync.js config file:

Download

3. Download the graphql schema:

Export schema ▾

# Mobile Hub



- 为一个移动项目构建多种服务
- 依照最佳实践进行通用配置
- 下载服务预配置常量 for iOS, Android & JavaScript
- 到处项目配置，以便多次重用部署
- 提供图形界面控制台 Console 和 Mobile Hub CLI 命令行

# AWS 移动服务官网

AWS 移动

概览 功能 定价 资源 常见问题 移动主题

## AWS 移动服务

构建可扩展应用程序的最快捷方法

入门

### 使用初学者应用程序入门

没有应用程序？没问题！使用我们的初学者应用程序来快速启动开发并学习如何使用 AWS 构建。

#### 宠物跟踪系统

使用 **React Native** 构建用户可以上传宠物照片的 **iOS** 和 **Android** 应用程序。

#### 餐厅订餐系统

**React.js PWA** 可让用户查看不同的餐厅菜单、选择菜品并下单。

#### 备注应用程序

演示如何在云中添加分析、用户登录和存储备注的备注应用程序教程。



<https://amazonaws-china.com/cn/mobile/>

# Trusted by category-defining apps

NETFLIX

tinder

yelp

airbnb

Periscope

Etsy

EASY TAXI

hike  
messenger

# 联系方式

黄卓斌

AWS 解决方案架构师

huangzb@amazon.com

# 谢谢!

扫码下载演讲资料

