



BEIJING 2017

如何使用AWS构建跨地区的应用体验

张旻皓
绿洲游戏



促进软件开发领域知识与创新的传播



关注InfoQ官方信息
及时获取QCon软件开发者
大会演讲视频信息



扫码，获取限时优惠

ArchSummit
全球架构师峰会 2017 [深圳站]

2017年7月7-8日 深圳·华侨城洲际酒店

咨询热线：010-89880682

QCon

全球软件开发大会 [上海站]

2017年10月19-21日

咨询热线：010-64738142

背景介绍



中国顶尖的海外游戏发行公司



- 多年互联网、游戏、电信行业经验
- 敏捷导师
- packagist开源社区活跃贡献者



Messaging



Database



Analytics



3个级别的AWS使用程度

基础设施替换

- 云主机：EC2
- 私有云：VPC
- DNS：Route53

外层服务使用

- 负载均衡：ELB
- 安全策略：WAF
- 内容加速：CloudFront
- 通知：SNS

托管的第三方服务

- 数据库：RDS
- 缓存：ElastiCache
- 搜索：ElasticSearch
- 运维：OpsWorks

将AWS引入工作流

- S3
- DynamoDB
- SQS
- Redshift
- Lambda
- Kinesis
- Rekognition
-

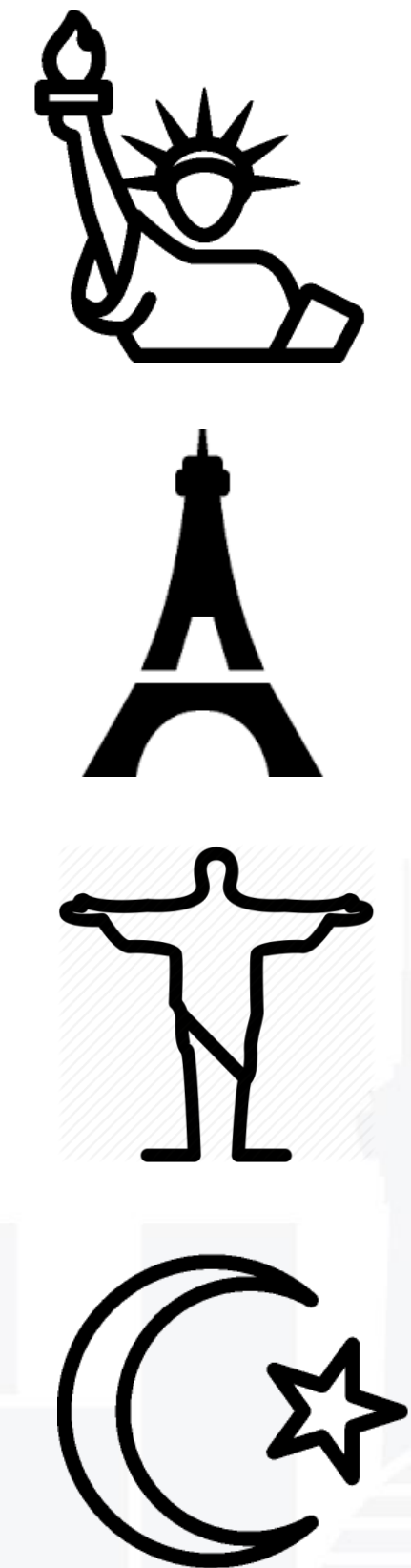
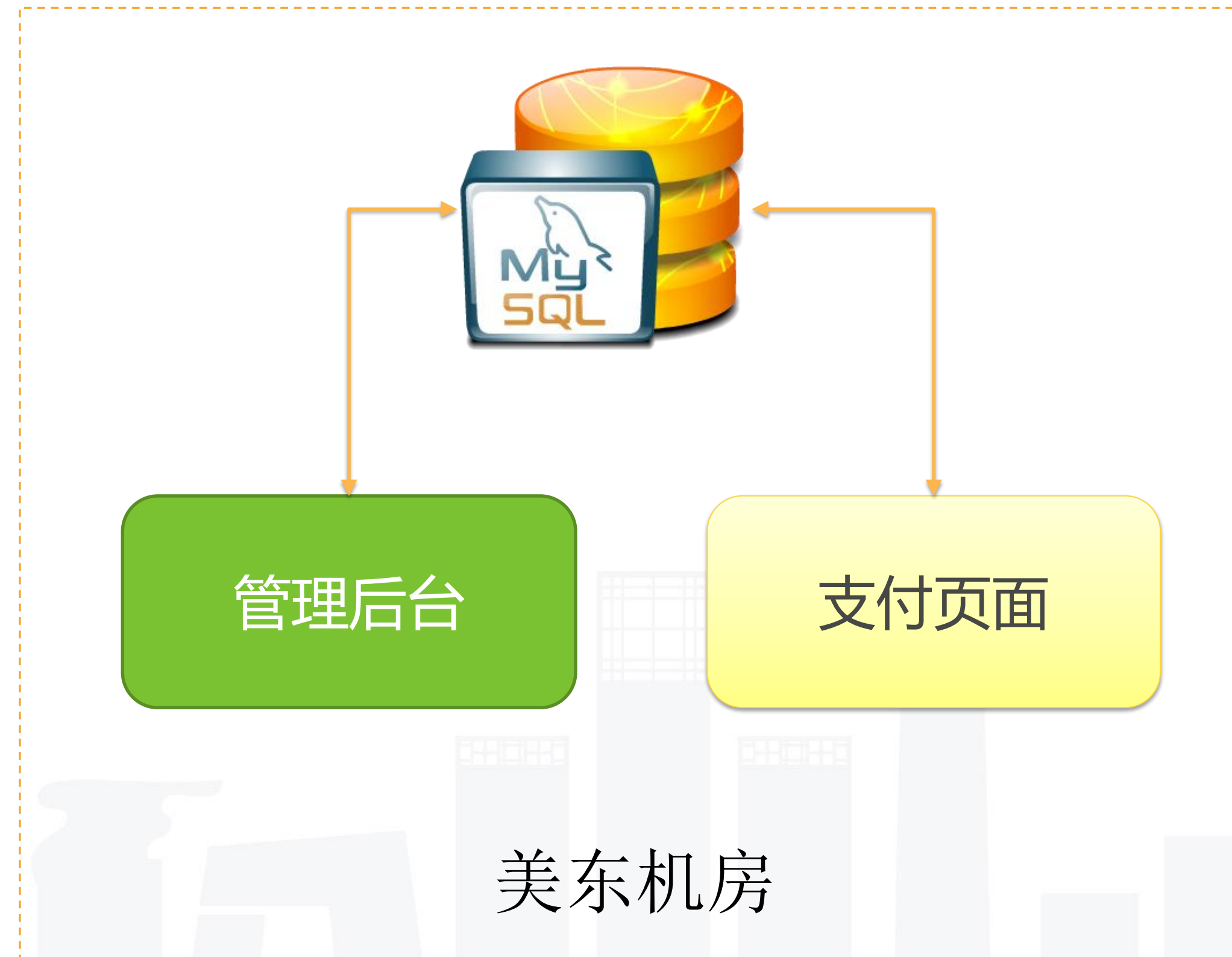


案例分析

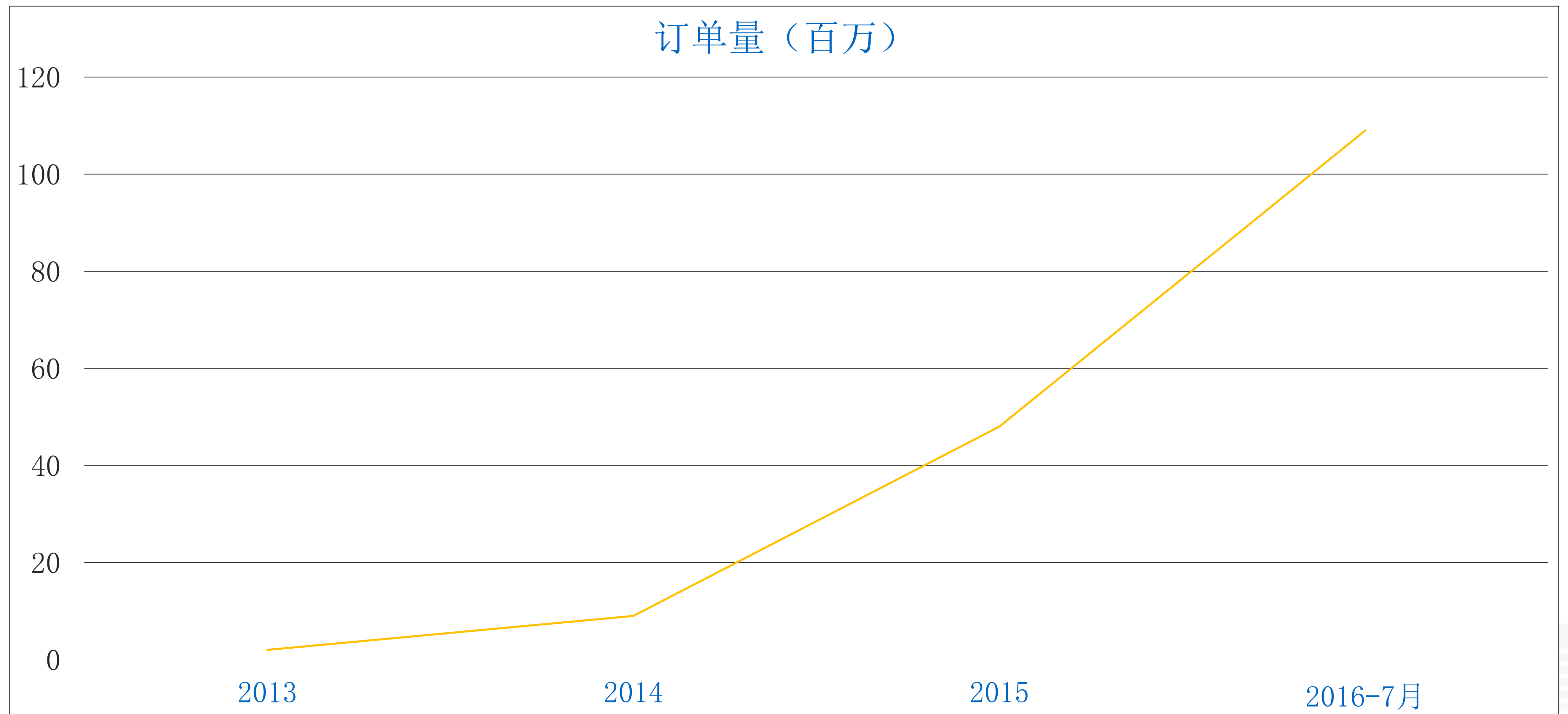
我们是如何通过AWS重构一款跨地区的支付应用



绿洲游戏支付平台 - v1.x (旧)



业务快速增长



旧系统问题

扩容决策

切片重构

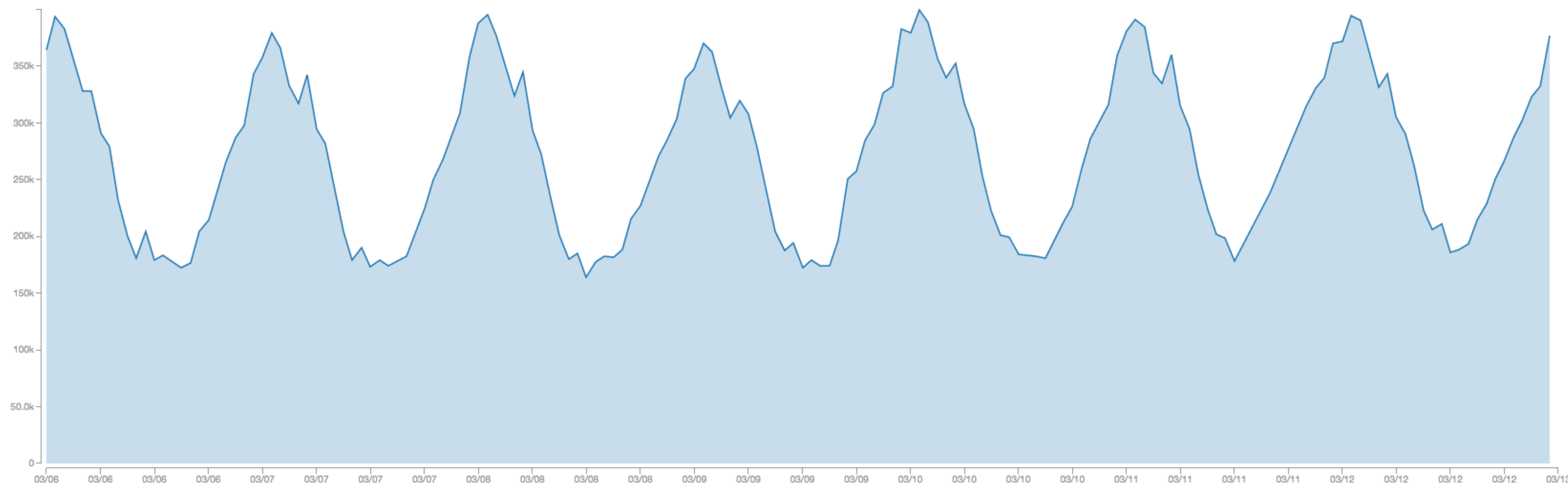
体验抱怨

数据备份



算力堆砌的浪费

预留算力



旧系统总结

优化方案（已尝试）：

- SQL优化
- MySQL参数调优
- 用RDS替换自部署
- 纵向扩容（升级）
- 切割冷热数据

待解决问题：

- 地理延迟（北京 - 美东，玩家 - 美东）
- 扩展性
- 算力浪费





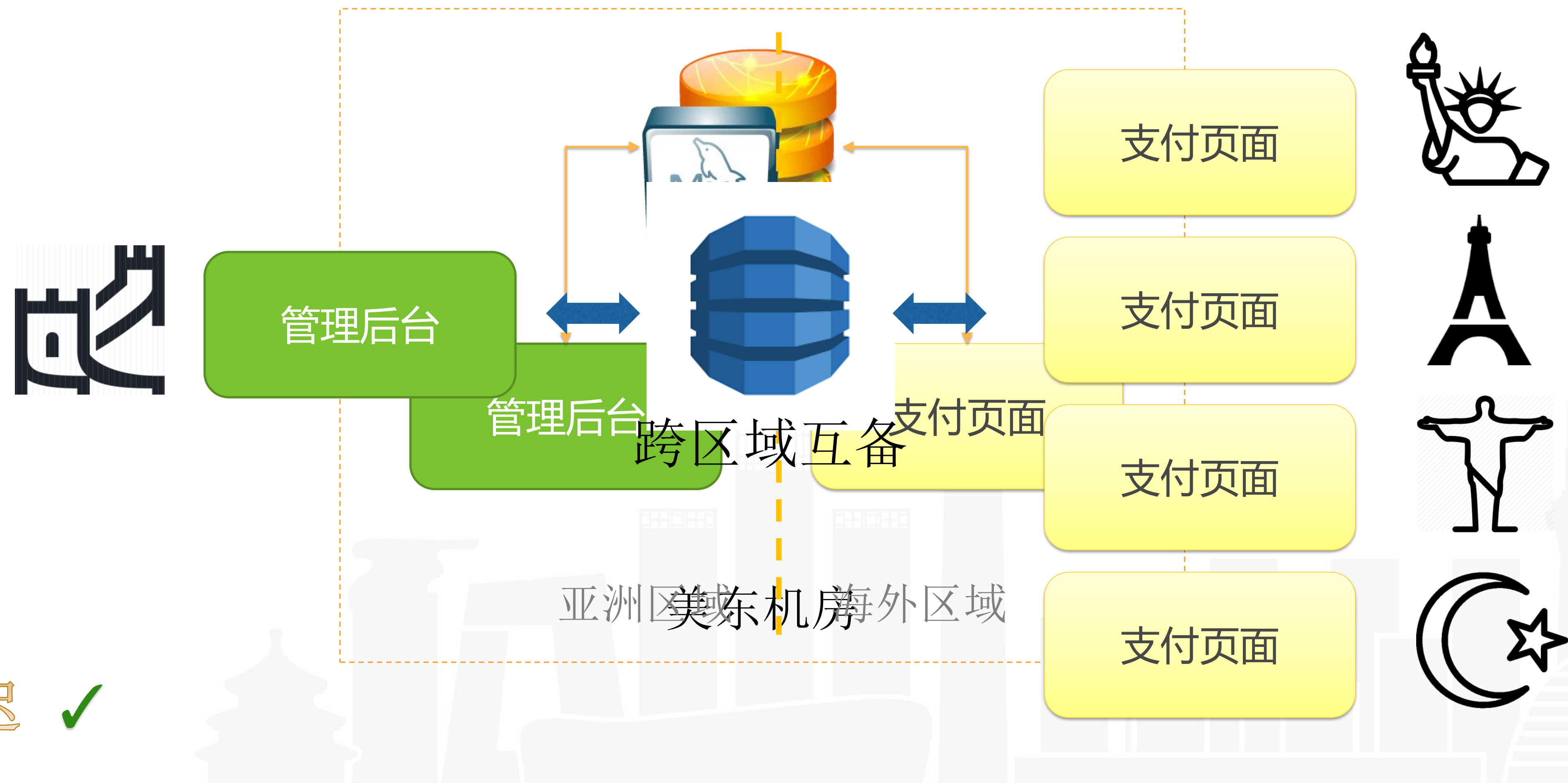
深入分析

第一步：从架构层解决地理延迟

问题：

- 地理延迟
- 扩展性
- 算力浪费

新系统架构推导





深入分析

MySQL的替代品

问题：

- ~~地理延迟~~
- 扩展性
- 算力浪费
- 跨区域互备

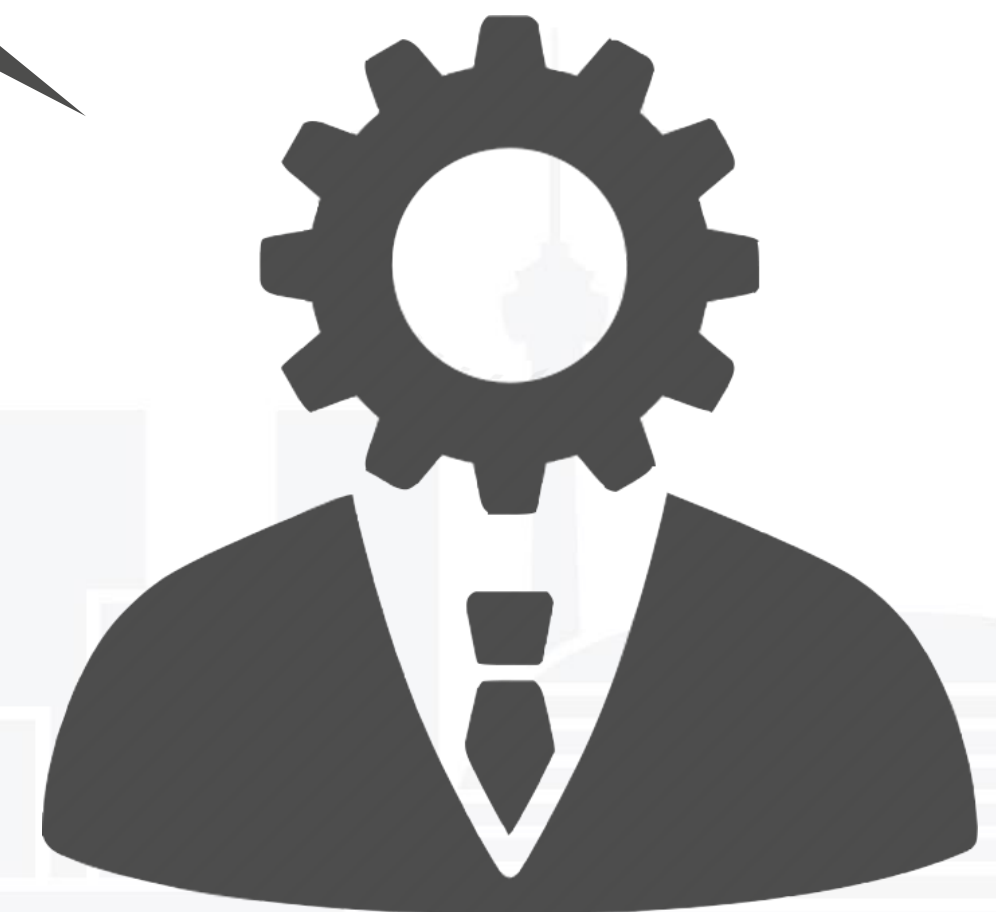
数据类型分析

	配置数据	用户生成数据（UGD）
量级：	小，几乎不增长	大，持续增长
类别：	一些	少量
北京节点：	读写	只读
海外节点：	只读	读写

RDBMS - 我们是否需要那个R

- 扩展性问题的根源：
 - 海外节点的读写请求
 - UGD，用户数据的增长
- 关系约束：
 - UGD：无
 - 配置：有、弱
- 事务保护：
 - 数据一致性要求强
- 分析查询：
 - SQL：需要
 - 关系：不需要

试一试
DynamoDB



简介DynamoDB

- Key-Value数据库（支持Document）
- 高灵活性：按照访问量阈值和数据量计费，实时更改需求
- 高可靠性：至少3份备份，跨越不同AZ
- 高扩展性：
 - 永远常量级别的访问速度
 - 可实时调整访问量阈值
 - 全自动横向扩容
- 易维护：全托管服务，集成CloudWatch提供全面监控报警

算力浪费 ✓

可扩展性 ✓

- 不可以使用SQL进行数据分析
- 不支持事务

重构时刻

决定：

- 迁移UGD到DynamoDB
- 将配置数据留在MySQL（RDS）
- 利用AWS RDS完成配置数据的跨区域复制

进一步行动点：

- 实现DynamoDB的跨区互备
- 实现数据一致性约束（事务处理）
- 检视分析查询场景





深入分析

DynamoDB的跨区域互备

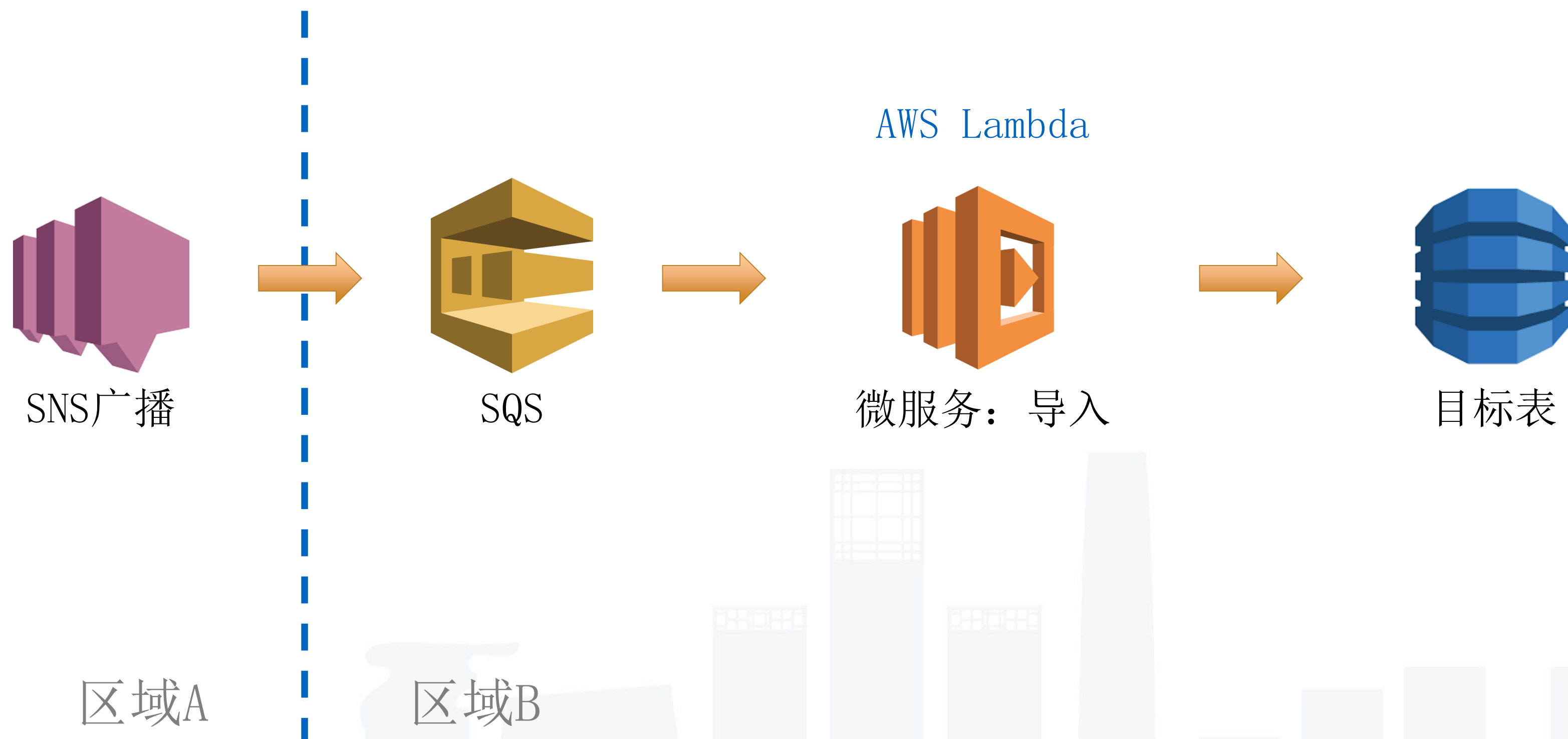
问题：

- ~~地理延迟~~
- ~~扩展性~~
- ~~算力浪费~~
- 跨区域互备
- 数据一致性
- 查询场景

实现DynamoDB的数据镜像 - 发现



实现DynamoDB的数据镜像 - 执行



跨区域复制的核心功能

SNS + SQS 自动实现

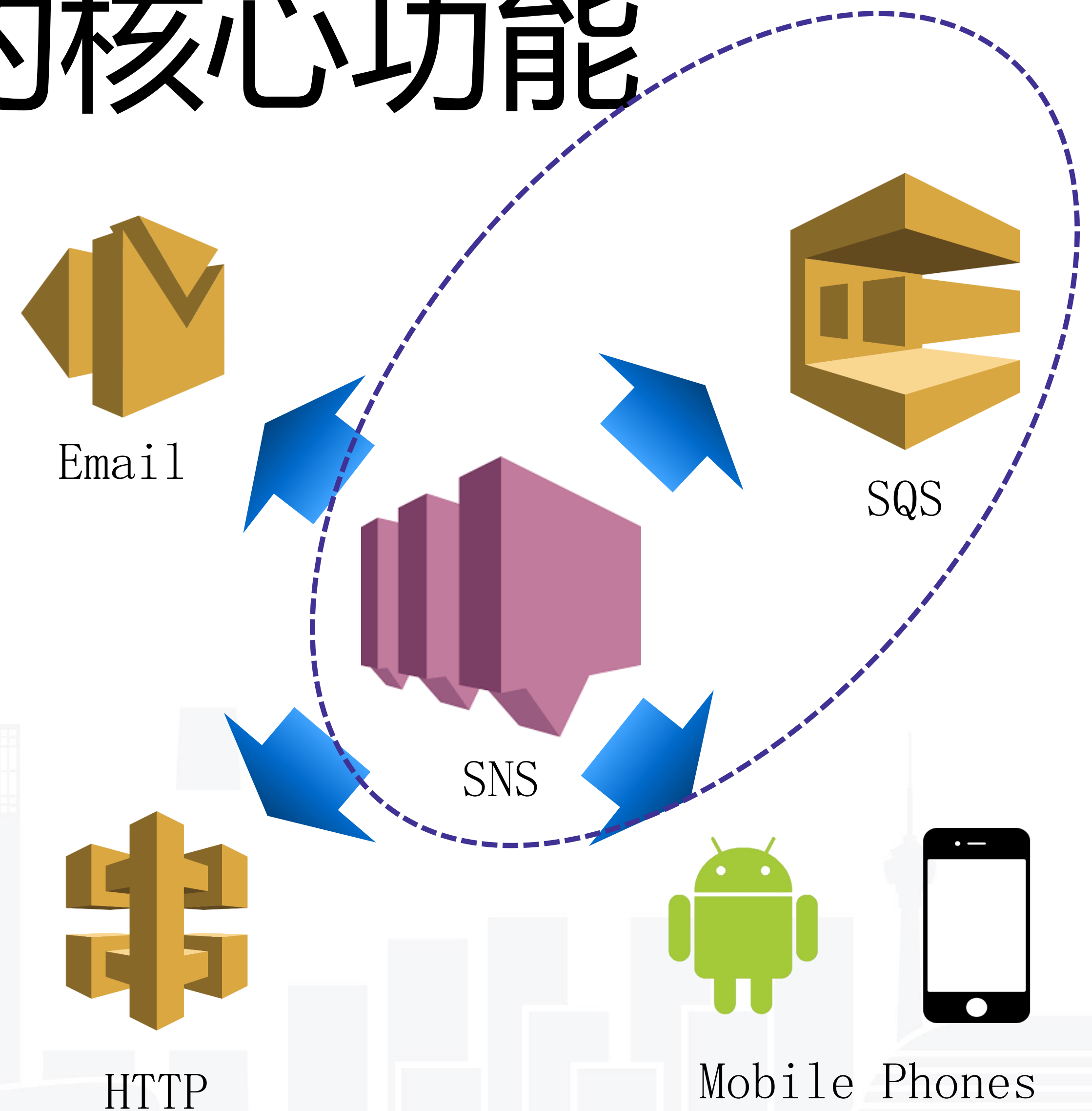
SNS：通知服务

- 自动扩展
- 至少一次送达的保证
- 支持HTTP、Email、SQS等多种方式

SQS：队列服务

- 自动扩展
- 每条消息至少被接收一次
- FIFO possible

跨区域互备 ✓





深入分析

数据一致性与查询场景

问题：

- ~~地理延迟~~
- ~~扩展性~~
- ~~算力浪费~~
- ~~跨区域互备~~
- 数据一致性
- 查询场景

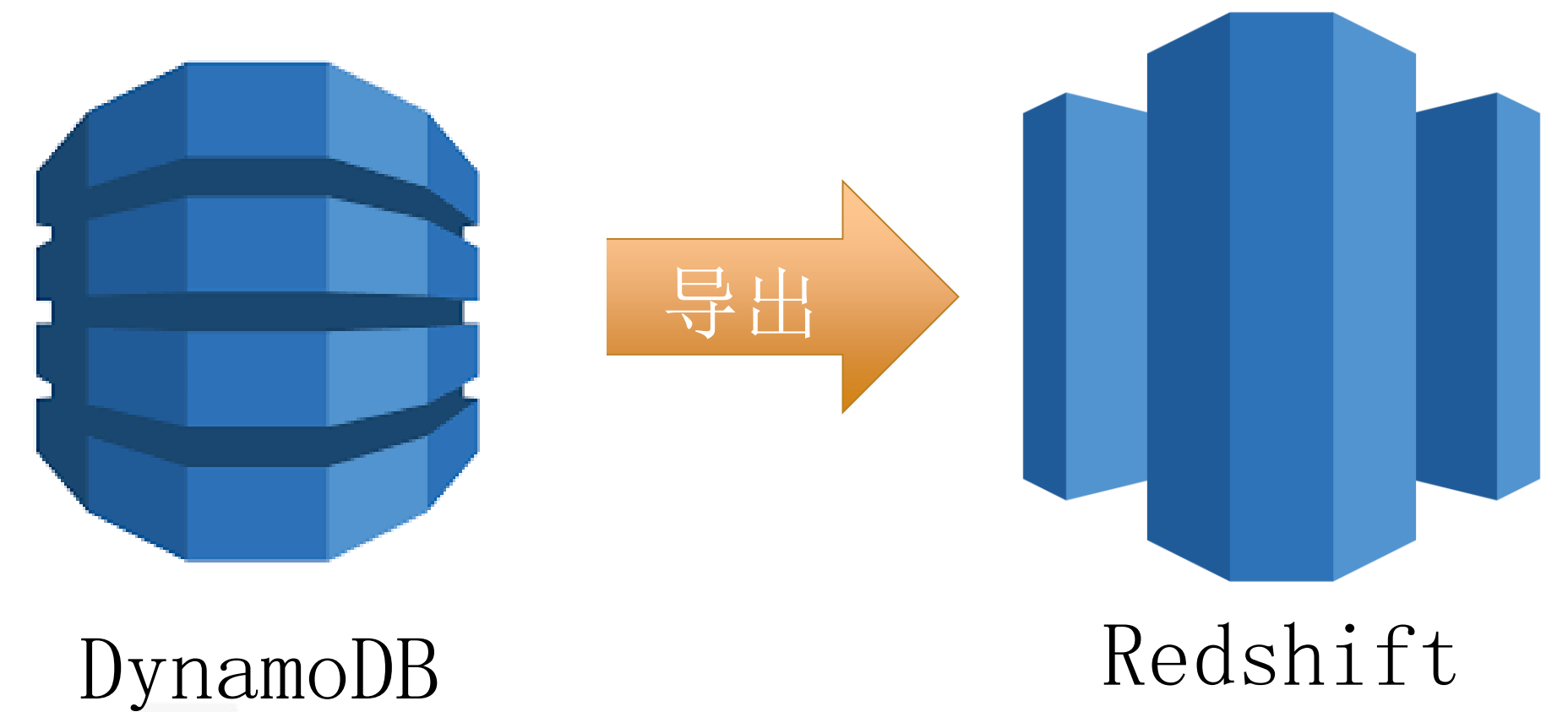
DynamoDB的数据一致性

- 读：
 - 使用Consistent-Read标志，保证读到最新的数据改动
- 写：
 - 增加“version”字段
 - 每次修改均使用条件表达式，在条件表达式中判断“version”

数据一致性 ✓

数据分析的选择：Redshift

- DynamoDB的缺点：
 - 可查询，但不支持SQL
 - 非常有限的索引
- 简介AWS Redshift：
 - 专为大数据（PB级）分析而生的列式数据库
 - 无需设计复杂索引
 - 高性能：
 - GB级：秒级
 - TB级：分钟级
 - PB级：分钟级（在增大配置情况下）
 - 兼容PostgreSQL
 - 无关系约束



查询场景 ✓



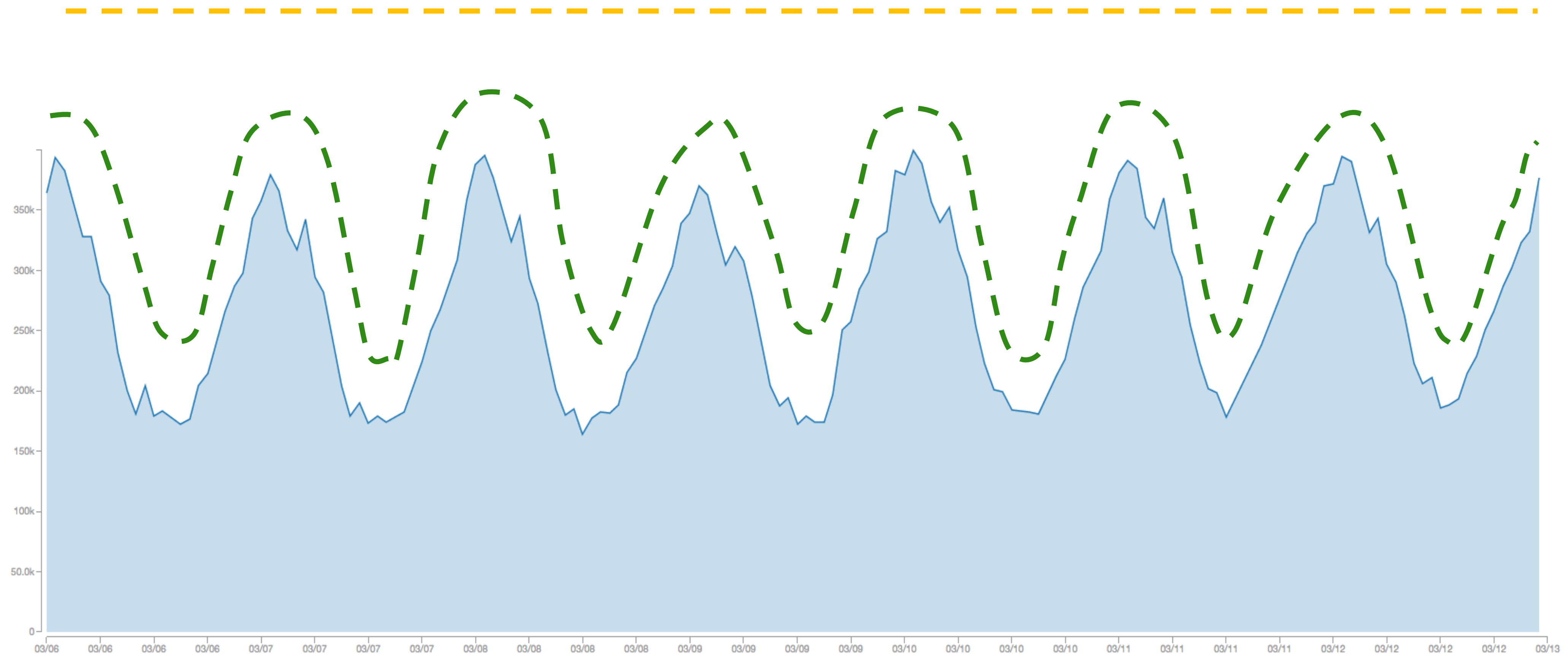
深入分析

使用AWS的成本影响

问题：

- ~~地理延迟~~
- ~~扩展性~~
- ~~算力浪费~~
- ~~跨区域互备~~
- ~~数据一致性~~
- ~~查询场景~~

算力浪费的解决



成本分析

增加开支:

- AWS EC2、RDS等的单价在市场没有较强竞争力

节约开支:

- 动态调整算力用量可至少节约30%-40%算力 (=成本)
- 针对长期使用的服务 (算力中的底线值) 购买预留实例可以有效降低成本 (40%)
- 全托管服务带来的运维成本节约
- 引入AWS服务重构你的软件, 在生产力上更可以获得极大提升



总结

- 使用AWS的3个级别
- 案例分析：用DynamoDB替代RDBMS
- DynamoDB跨区域互备：SNS + SQS
- 使用Redshift进行大数据分析
- 成本节约





关注QCon微信公众号，
获得更多干货！

Thanks!



主办方 **Geekbang** 极客邦科技 **InfoQ**

张旻皓

minhao.zhang@oasgames.com