



OpsWorld 运维世界大会·深圳站

# 基于日志数据运维/运营之道

简志：阿里云高级专家

微博：阿里云-简志

# 经历云计算研发过程



# 变化：节奏变了

App开发

卖方市场→买方市场

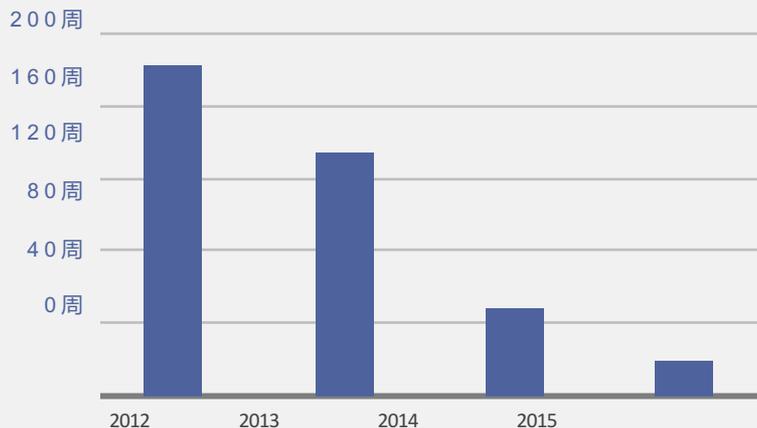
云对技术影响

运维转向运营  
上线周期缩短

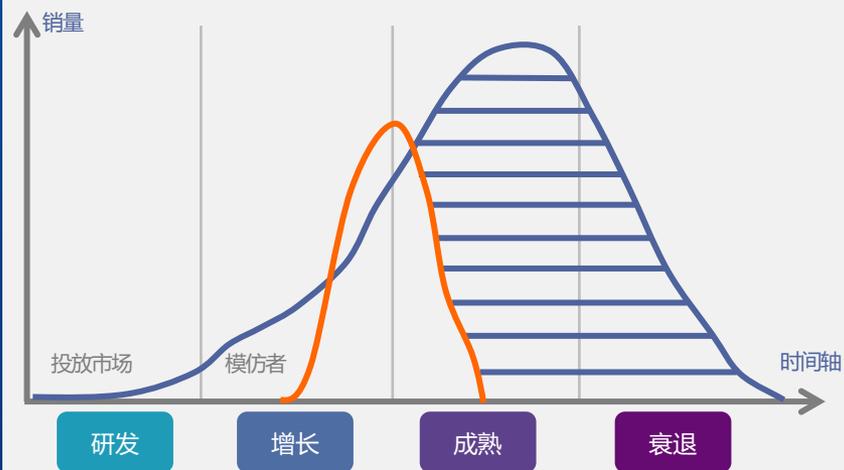
数据改变结果

收集百万用户行为  
实时提升游戏（投放、运营、活动）

成熟时间（从发布到饱和，累计下载）



生命周期



# 变化2：软件研发与架构设计方案

应用架构

单体系统



分层架构



微服务

资源单位

服务器



虚拟机



容器

资源管理

专有系统



软件定义



事件驱动

发布与部署

主版本发布与停机升级

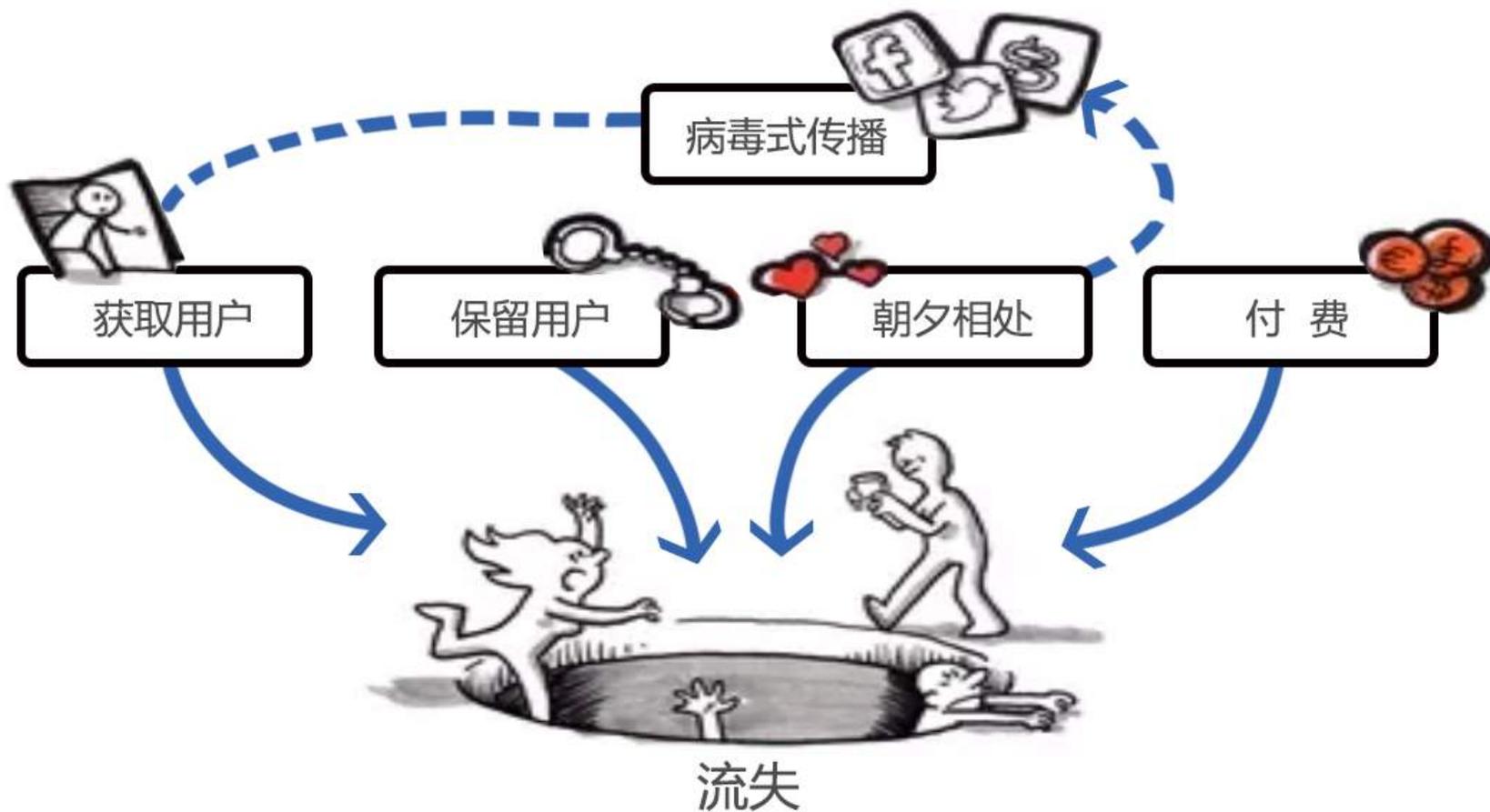


软件更新包与热升级



持续交付

# 不变：用户获取途径



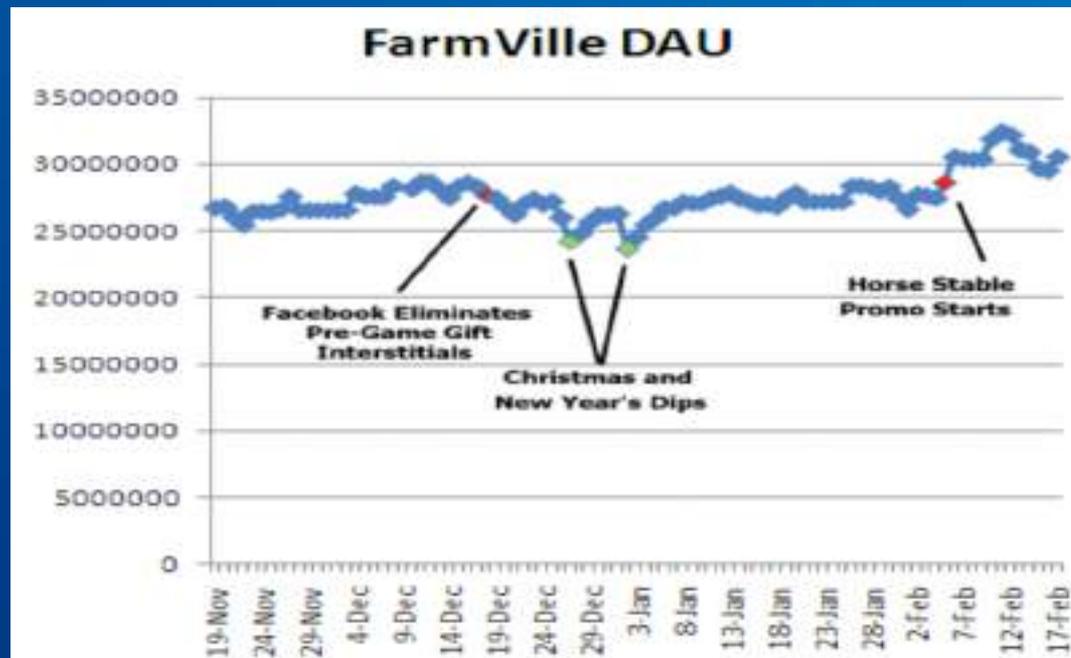
# PD视角

## 问题

- 多久玩一次?
- 什么时候退出?
- 能把用户捞回?
- 发生了什么?

## 来源与声音

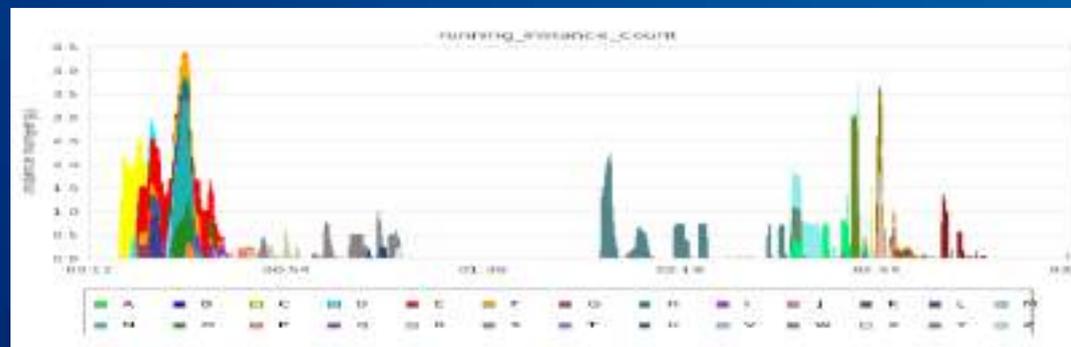
- 微博, 朋友圈评论
- 网页访问次数



# Ops视角

## 问题

- 是否需要扩容
- 带宽是否足够



# 运营视角

## Facebook API

- 用户访问游戏
- 用户信息

## 用户参与度

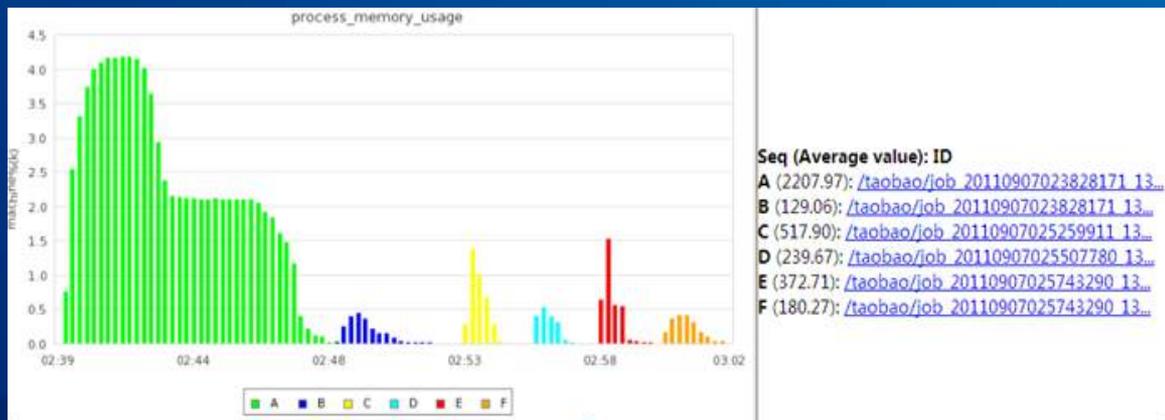
- 送礼物
- 收礼物



# OPS视角

## 谁的资源占用

- 优化资源配置
- 量化程序性能与资源占用



# 运营看什么

推广达成率



# Ops看什么



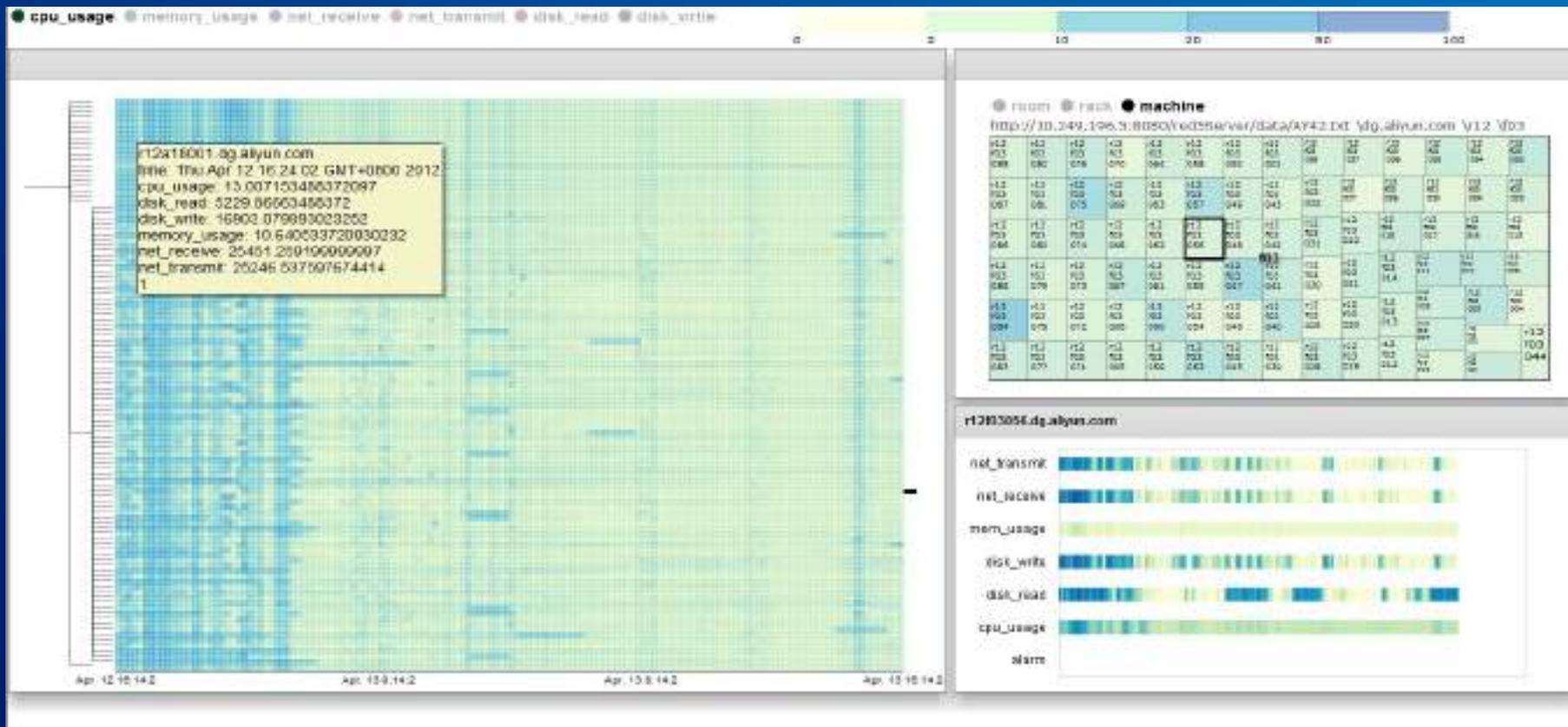
# Dev分析用户路径

Table 22.1 Player deaths by cause

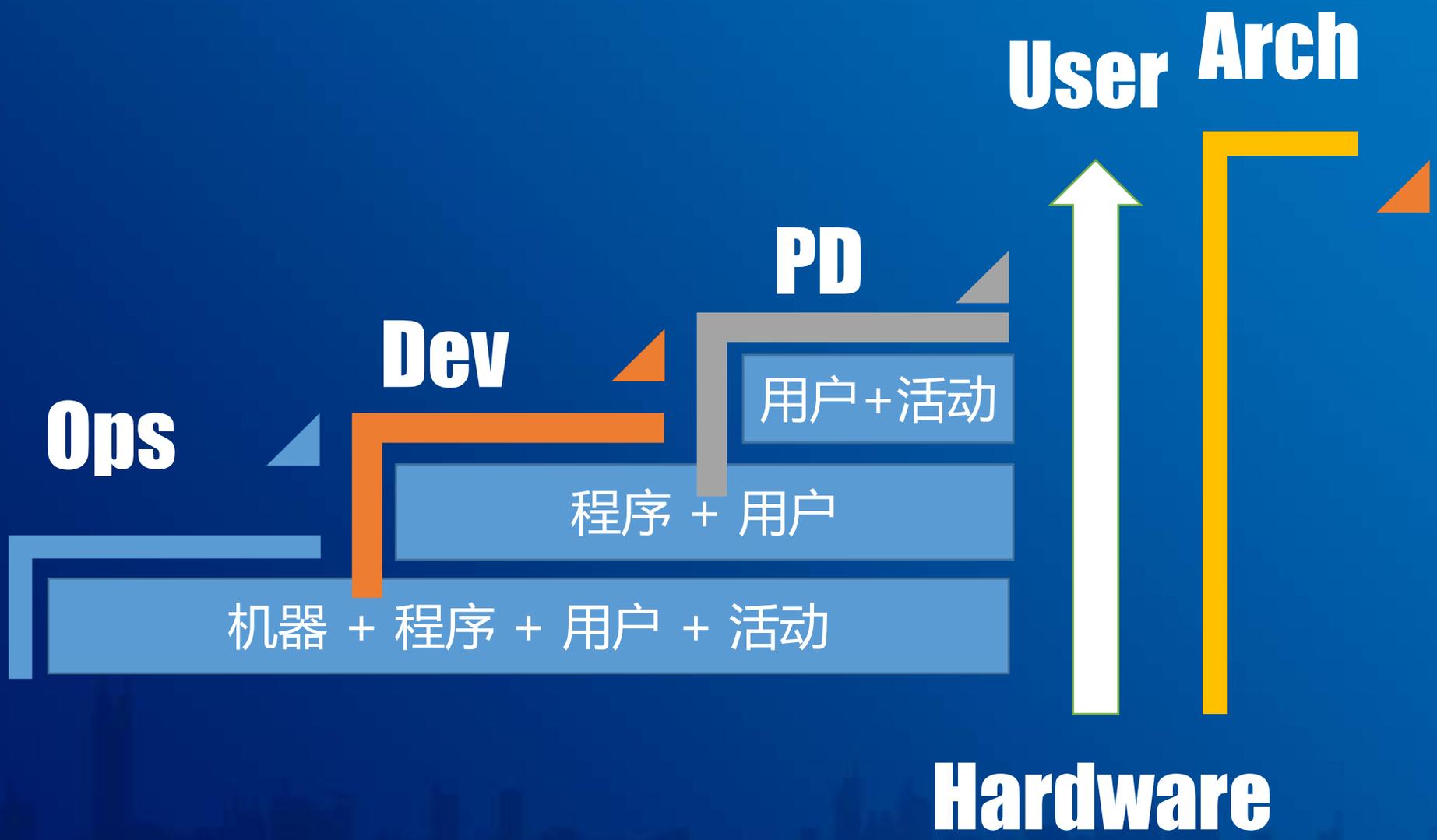
Name	PC death by cause						Grand total
	global_tag						
	@1Station1	@1Station2	@1Station3	@1Station4	@1Station6	@1Station7	
Other	19	3	8	15	4	20	69
Special-SniperRifle	4		1	14	3	2	24
Patrol-Gang	4		1	11	7	1	24
heli_fighter_02_w	3		2	7	1	8	21
brute_mirigan	1	0		3	3	3	10
TermpPickup-LuchadoreGren.	1	0	2	2	2	6	15
SMG-Cyberspace			4		4	2	10
SMG-Gang	3			5			8
srv_4dr_02_w			4	2	2		6
Shotgun-Gang			2	3		1	6
Rocketlauncher	2		2	1	1		6
sp_vio101_w			2		1	2	5
Patrol-Police	2			2	1		5
Avatar_sword					4		4
srv_4dr_luxury05_w	3						3
Shotgun-Police	2		1				3
sp_tank03_w			2				2
sp_gat01_w	1		1				2
SMG-Storm				2			2
Shotgun-STAG	2						2
Rocketlauncher_VB					2		2
Rifle-STAG	1		1				2
smg_gun	1						1
sp_vio102_w						1	1
sp_tank01_w2	1						1
sp_tank01_w			1				1
Rifle-Gang	1						1
<b>Grand total</b>	<b>51</b>	<b>7</b>	<b>32</b>	<b>67</b>	<b>37</b>	<b>48</b>	<b>242</b>



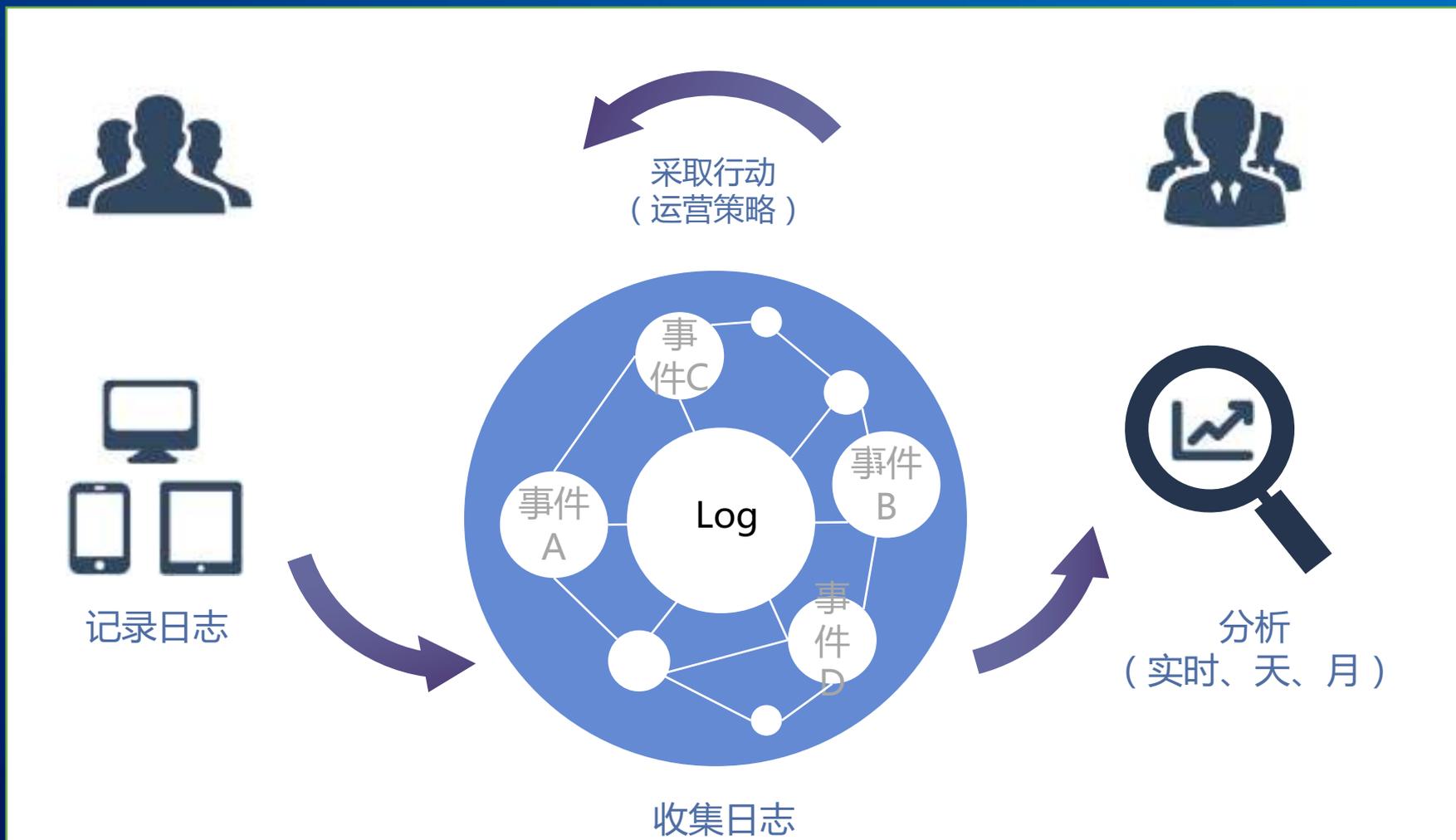
# OPS 分析硬件行为



# OPS 工作范畴与交集



# 数据驱动 → 决策过程



# 为什么是日志



# 如何来运维、运营日志数据

- 酒店业务运营
  - 住户：多种来源
  - 开发商：Dev
  - 管理：Ops
  - 销售：运营活动
- 目标：
  - 提升管理效率，提高用户体验
  - 将Ops做成品牌服务（万科物业）



# 第一步：数据采集

- 基础资源：电力、水利、网络
- 安全相关：摄像头，烟雾传感器
- 用户体验：电梯运行情况，停车场出入情况，刷卡情况
- 其他：用来从哪里来消费



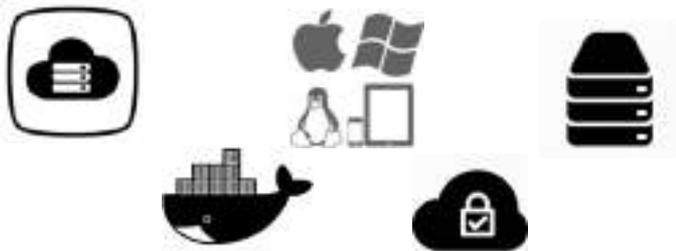
# 实时数据采集

用户行为散落在外部

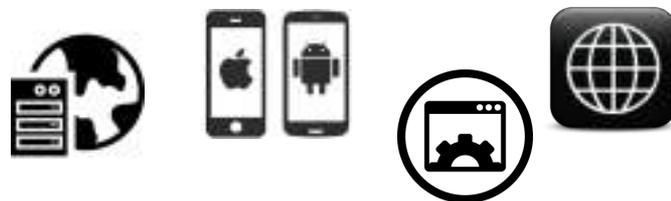
数据中心

不同地域

多渠道



IT 基础设施产生日志 →



多终端

异构网络

LogHub

Debug Log

User Action

Sys Operation

Order List

Server Log

Trace Info



# 有哪些方案

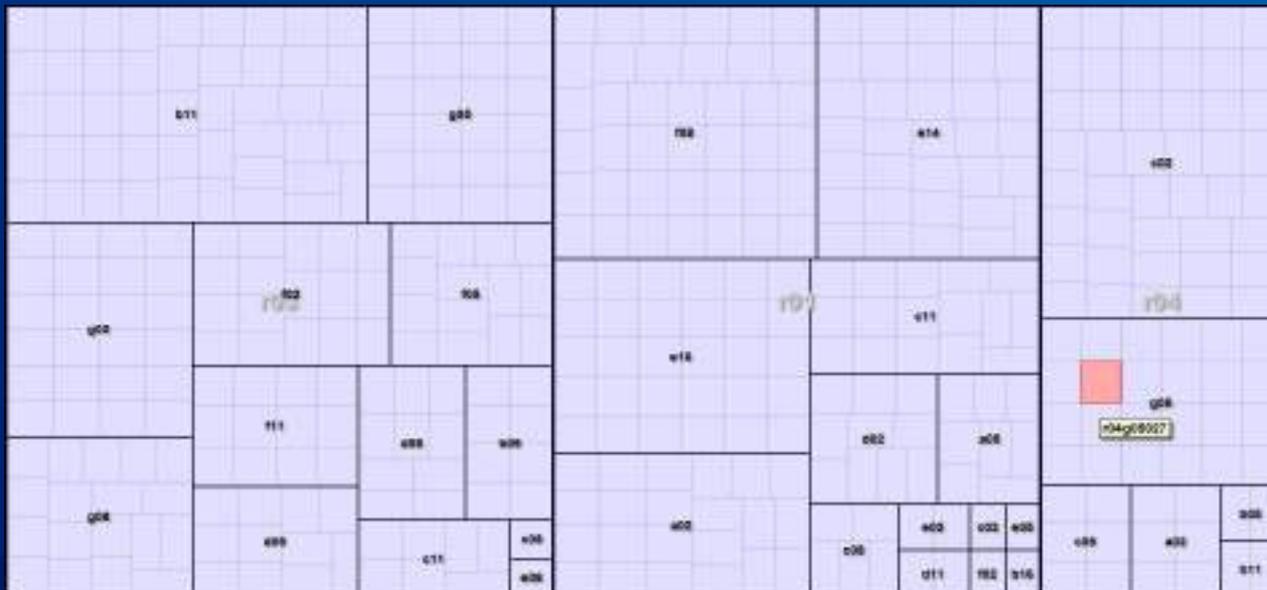


- wechat-server (存储微信服务器访问日志)
- wechat-app (存储温馨服务器应用日志)
- wechat-error (错误日志)
- alipay-server
- alipay-app
- deliver-app (送货员app状态)
- deliver-error (错误日志)
- web-click (H5页面点击)
- server-access (服务端Access-Log)
- server-app (应用)
- coupon (应用优惠券日志)
- pay (支付日志)
- order (订单日志)



## 第二步：实时监控、报警

- 电表、水表，网络带宽
- 视频监控程序：人流量，入住率
- 烟雾报警程序：是否有错误
- 实时查询分析



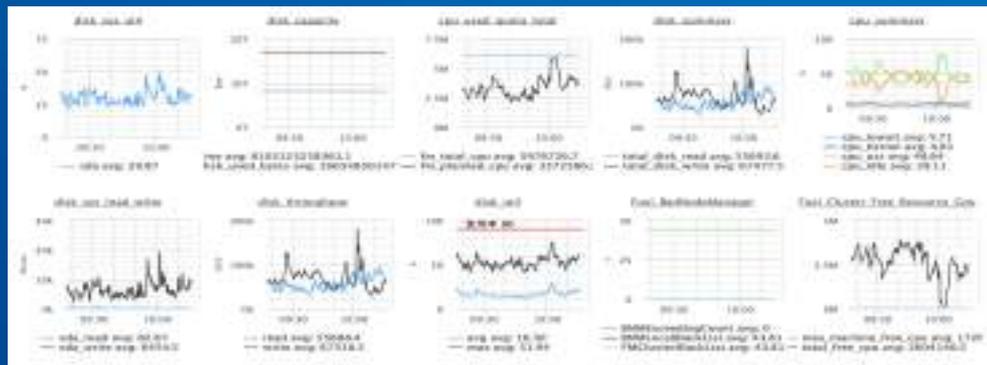
# 实时监控分析的技术

- 存储：Ganglia、OpenTSDB、InfluxDB
- 计算加存储
  - Storm + Hbase
  - SparkStreaming+ ES
  - Flink & Samza

# 案例：阿里云神农平台

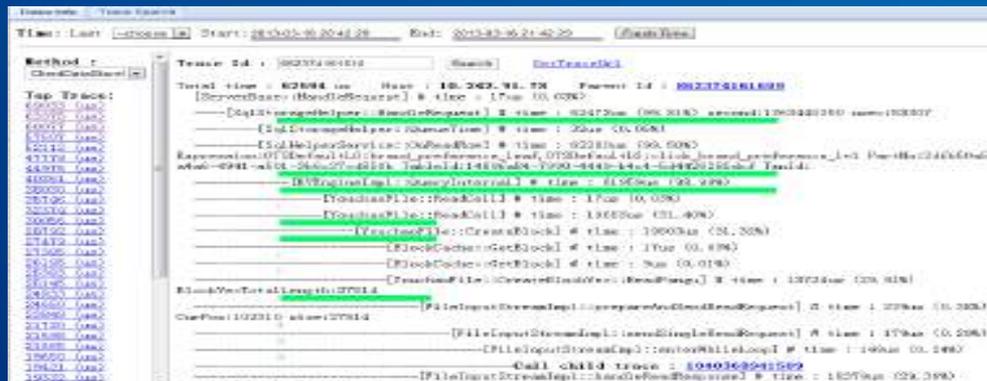
- 系统：

- Metric：5-10秒采样
- Tracer：按条件采样
- Logging：根据级别



- 应用：

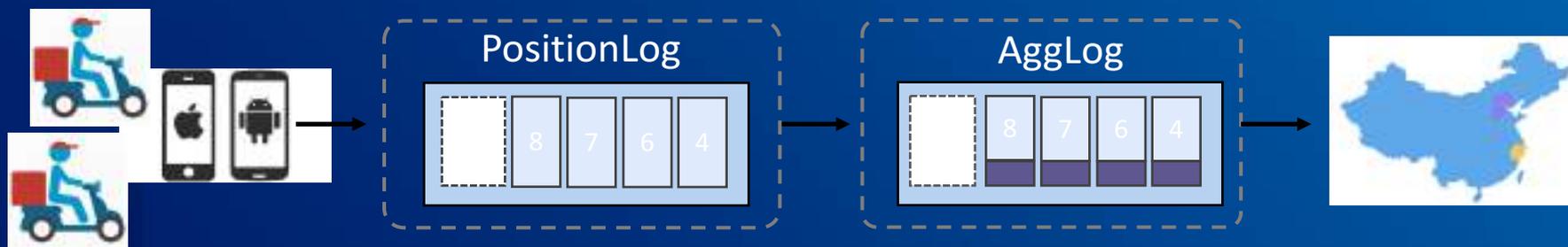
- StreamCompute
- 存储：NoSQL(Table Store)



# 案例：送货员位置

- 快递小哥哪个区域比较充足
- 繁忙程度如何？

```
2016-06-26 19:00:15 ID:10015 DeviceID:EXX12345678 Network:4G GPS-X:10.30.339 GPS-Y:17.38.224.5 St
```



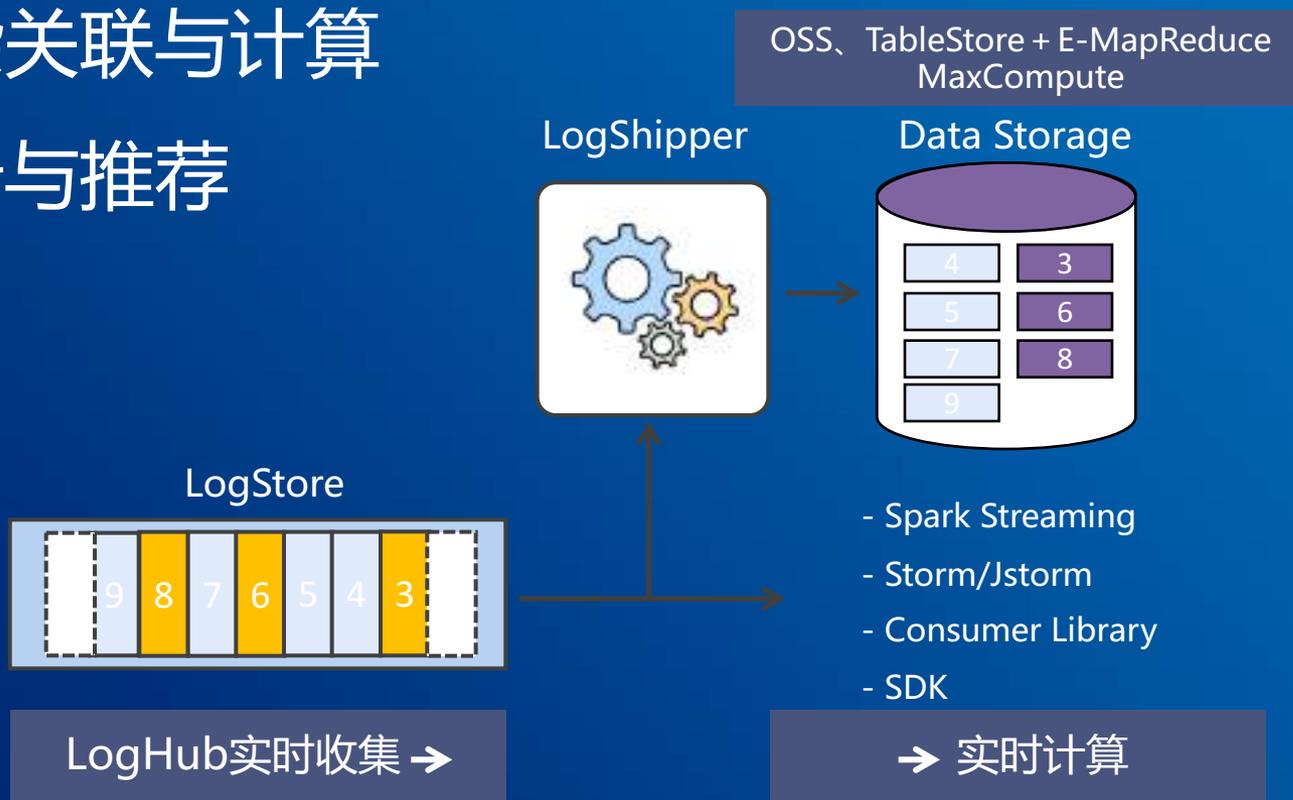
1. 日志收集

2. 地理坐标转换

3. 数据可视化

# 第三步：离线存储数据分析

- 较为复杂关联与计算
- 用户分析与推荐



# 案例：酒店推广

- 在若干渠道推广新用户
- 评估各渠道注册效果
- 评估各渠道注册数量

```
http://examplewebsite/login?source=10012&ref=kd4b
```

```
2016-06-20 19:00:00 e41234ab342ef034,102345,5k4d,467890
```

## 1. 设计用户注册URL

- time
- session
- source
- ref
- params

2. 生成二维码，各渠道传播
3. 服务端将URL参数写入日志
4. 数仓进行计算

渠道	用户数目	获取费用
传单-地铁站	700	2.0
传单-小区	900	1.5
网页注册	1130	0.7
推广渠道A	200	3

# 阿里巴巴日志处理架构

## LogHub

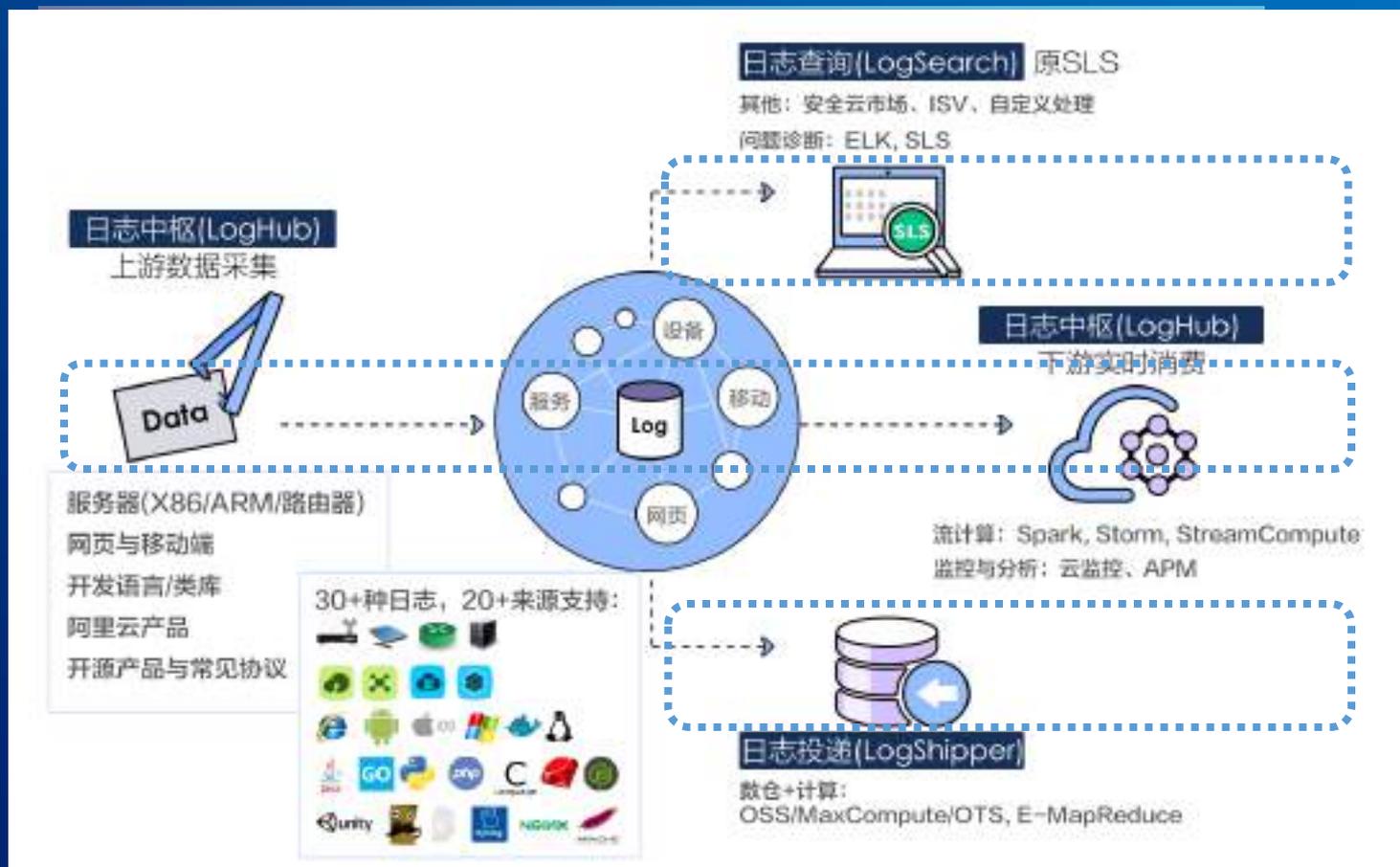
- 采集
- 消费

## LogSearch

- 索引
- 查询

## LogShipper

- 投递



# 现实中的挑战

- 流量波动
  - 活动、新服务开放、周末
- 数据采集、管道故障
  - 基础服务是否稳定
- 新增需求
  - 接入一组新数据：微信订房
  - 接入一个消费者：安全部门需要联网

# Docker场景中扩容案例

Config

← 日志路径

应用层



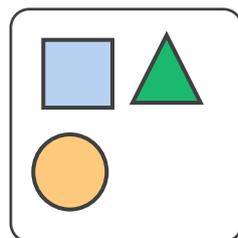
 新应用部署

机器组

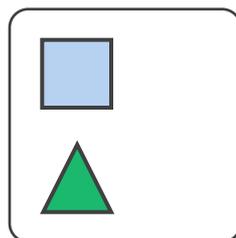
<- 应用部署

执行环境

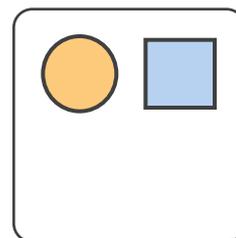
Server1



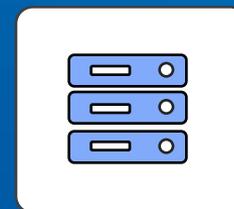
Server2



Server3



服务器扩容



Logstore

- LogHub
- LogSearch
- LogShipper

日志处理



增加消费端



# 选择阿里云日志平台原因



成熟：阿里集团同一套设施，久经考验

稳定：技术转换红利：高可靠 vs 极低成本



发挥空间大：完整支持开源、自研大数据方案

前沿：大量工程师与案例沉淀



# THANKS



日志服务微信群 ( 12/10 )



日志服务钉钉群