



基于七牛云大数据平台构建 APM 产品实践

黄德恒

纲要

- APM 简介
- 系统演进，遇到的问题
- 展望

APM

Application Performance Management

- 数字化体验监控 — Digital Experience Monitoring
- 应用程序发现、跟踪和诊断 — Application discovery, tracing and diagnostics
- 应用程序分析 — Application Analytics

不断颠覆和创新是七牛成功的基石

市场创新的速度远超于人脑的预判



智能

发展深度学习技术，
从基础设施云转向
智能云服务



大数据

发展大数据基础技术
将重心从数据的存储
和处理转向数据的价
值挖掘



计算

用容器化计算
服务来将之前
的服务通用化，
并推进基于微
服务架构的
DevOps理念



富媒体

从图片管理扩
充为支持视频
点播和直播的
富媒体云



海量存储

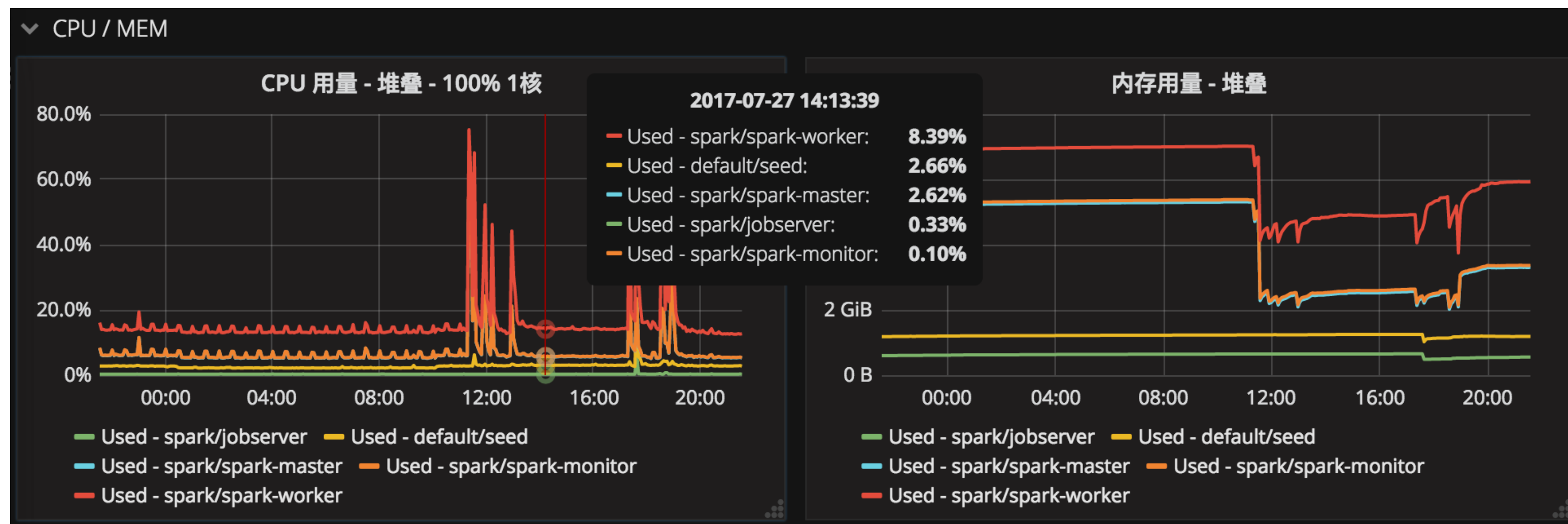
以数据处理
和内容分发的
存储PaaS切入
云计算市场

初衷

- 提升直播推流/播放 SDK 性能
- 优化直播节点质量
- 一些统计需求
- 日志查询需求

时序数据

基于时间的一序列数据点，在同一数据列中的各个数据必须是同口径的，要求具有可比性



萌芽版

- 内存里做数据聚合
- 打点到时序数据库，通过 Grafana 进行展示
- 导出全量日志到对象存储

萌芽版

数据源

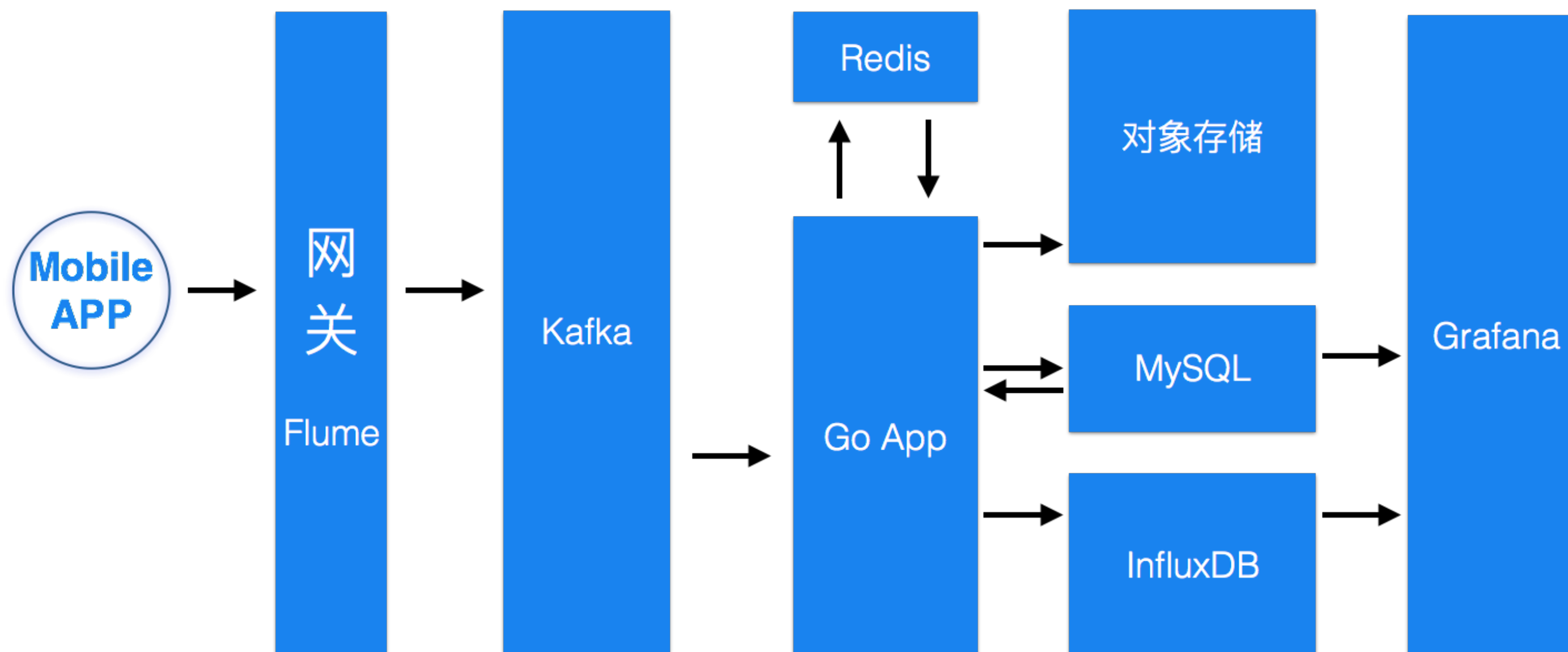
数据汇聚

异步队列

清洗

持久化

展示

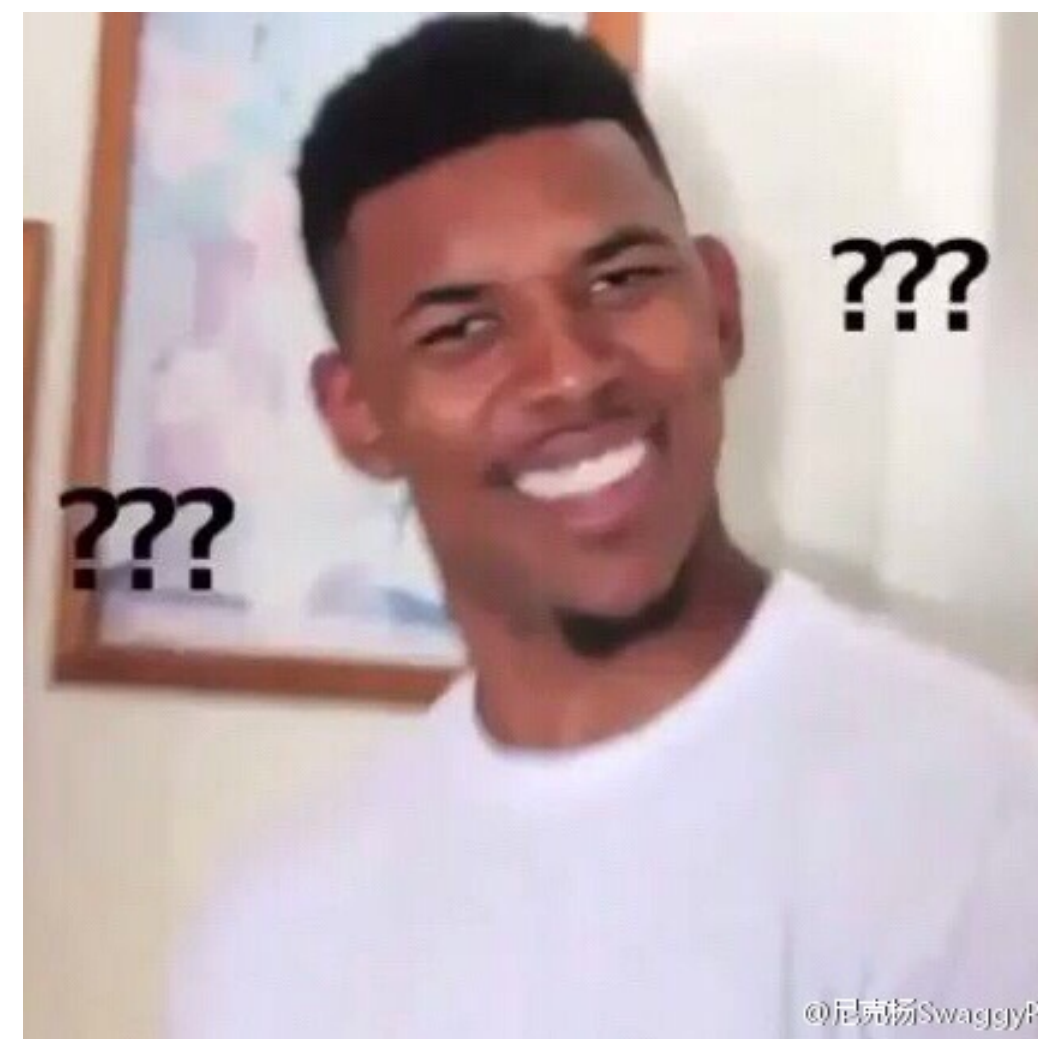


思路拓展

- 满足自身的统计、分析需求
- 做对七牛直播云用户有益的事
- 做对非七牛直播云用户有益的事

直播行业问题

- 视频打开速度慢 — 首开时间感人
- 视频经常缓冲 — 好似 PPT
- 用户出错排障慢 — 基本靠猜
 - 不知道谁的锅
 - 不知道何时开始出现问题



APM 目标

视频 APM

- 多维度 (CDN, 省份, 服务器 IP, ISP) 联合分析
- 实时监控数据
- 快速查询定位
- 快速统计单个主播终端在线数等简单运营统计

移动 APM

- 崩溃报告
- 网络性能监测
- App 响应时间监测
- 对性能指标的多维度联合分析
- 自定义统计数据分析

遇到了问题

- 功能扩展麻烦
 - 增加一个统计维度要修改源代码
 - 常见聚合函数需自己实现（如 percentile）
 - 新增一个统计点可能造成代码复用率降低
- 水平扩展困难
 - 遇到计算资源瓶颈时难以拆分统计逻辑

规划版

数据源

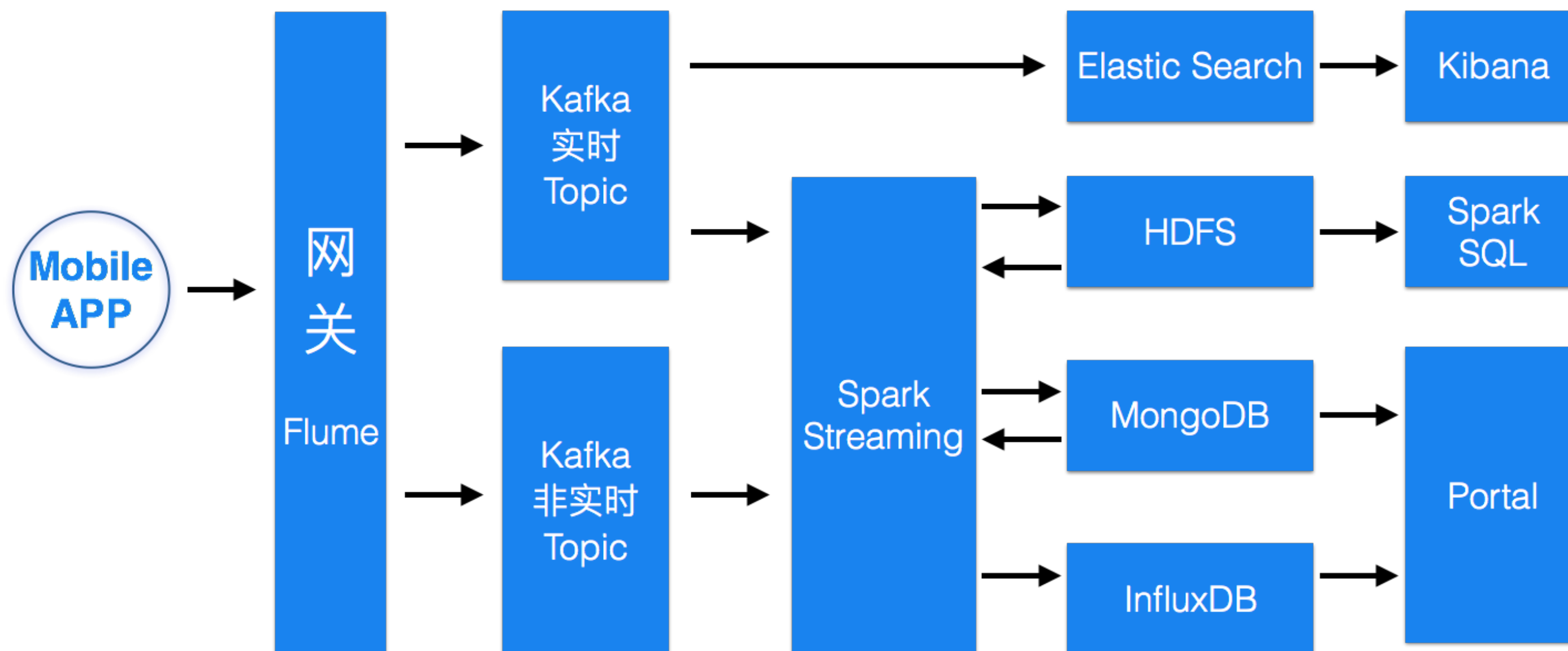
数据汇聚

异步队列

清洗

持久化

展示

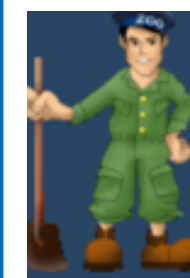


架构的纠结

数据可视化



批量/实时计算



Zook
eeper

集群调度



Yarn

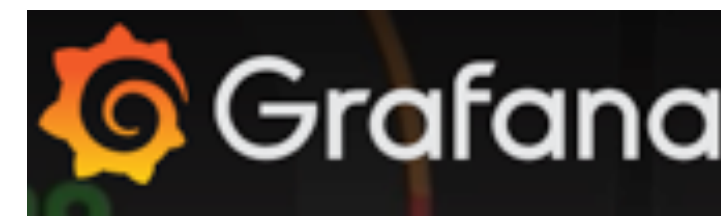
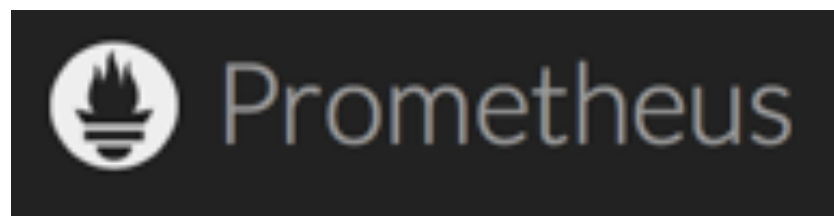


存储



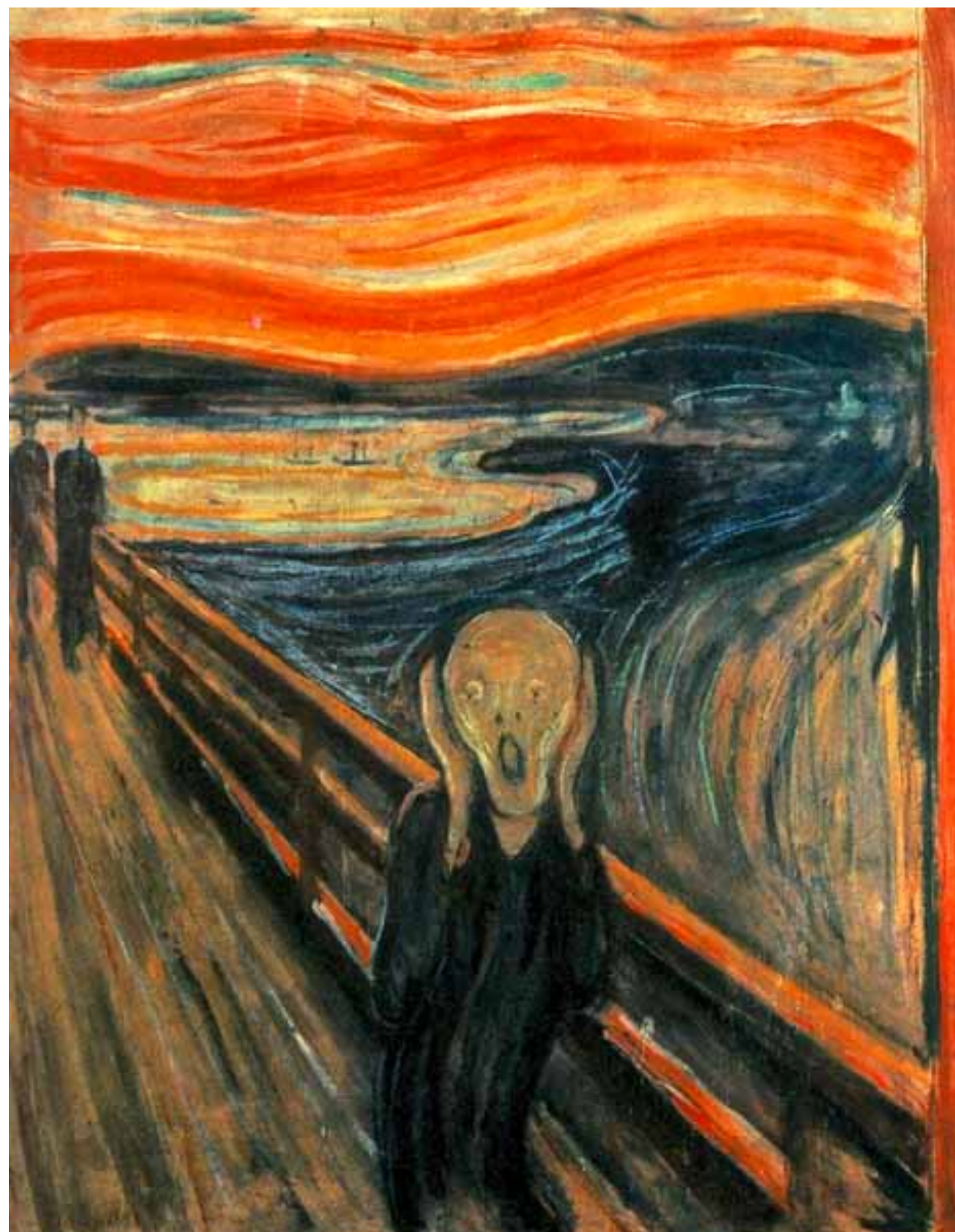
HDFS

监控



高可用

架构的纠结

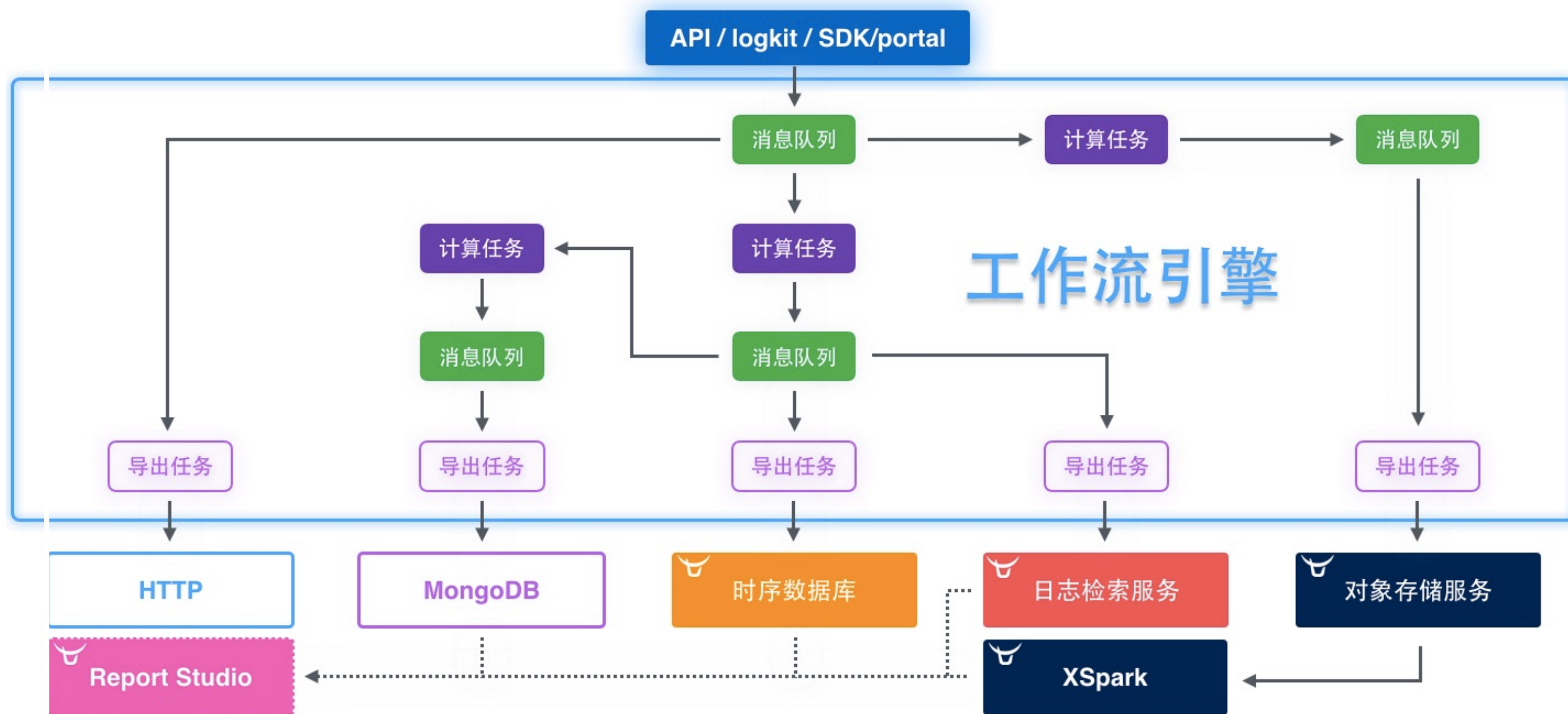


七牛大数据平台 — Pandora

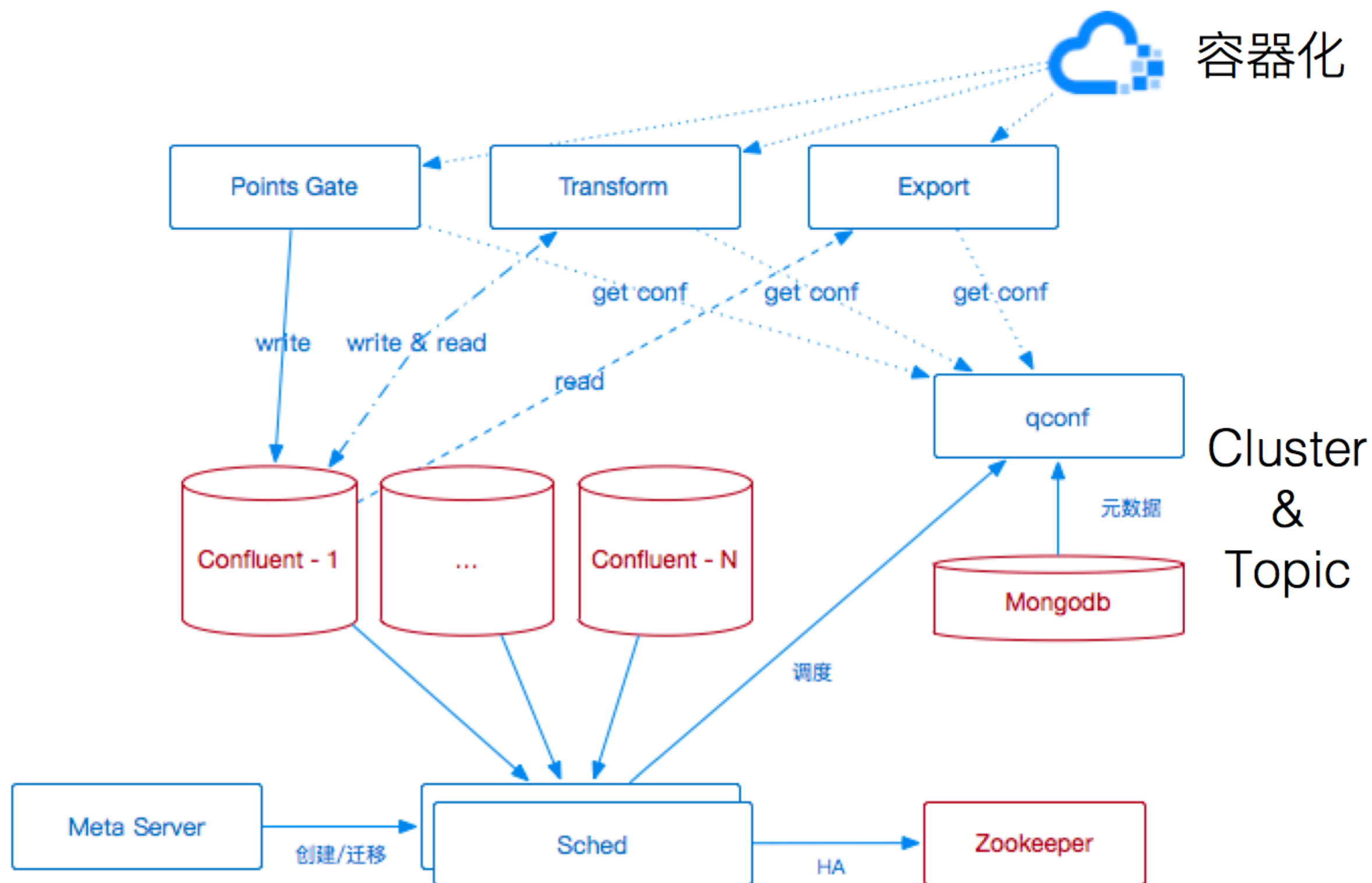
- 从存储到数据可视化，全栈的大数据分析产品
- 用户使用 Workflow 管理自己的数据流，无需大数据背景
- Workflow 实现可视化数据流监控，降低运维成本
- 集成优秀社区组件，优化并做到更好

Pandora

Pandora架构图



Pandora Kafka Clusters



实施版

数据源

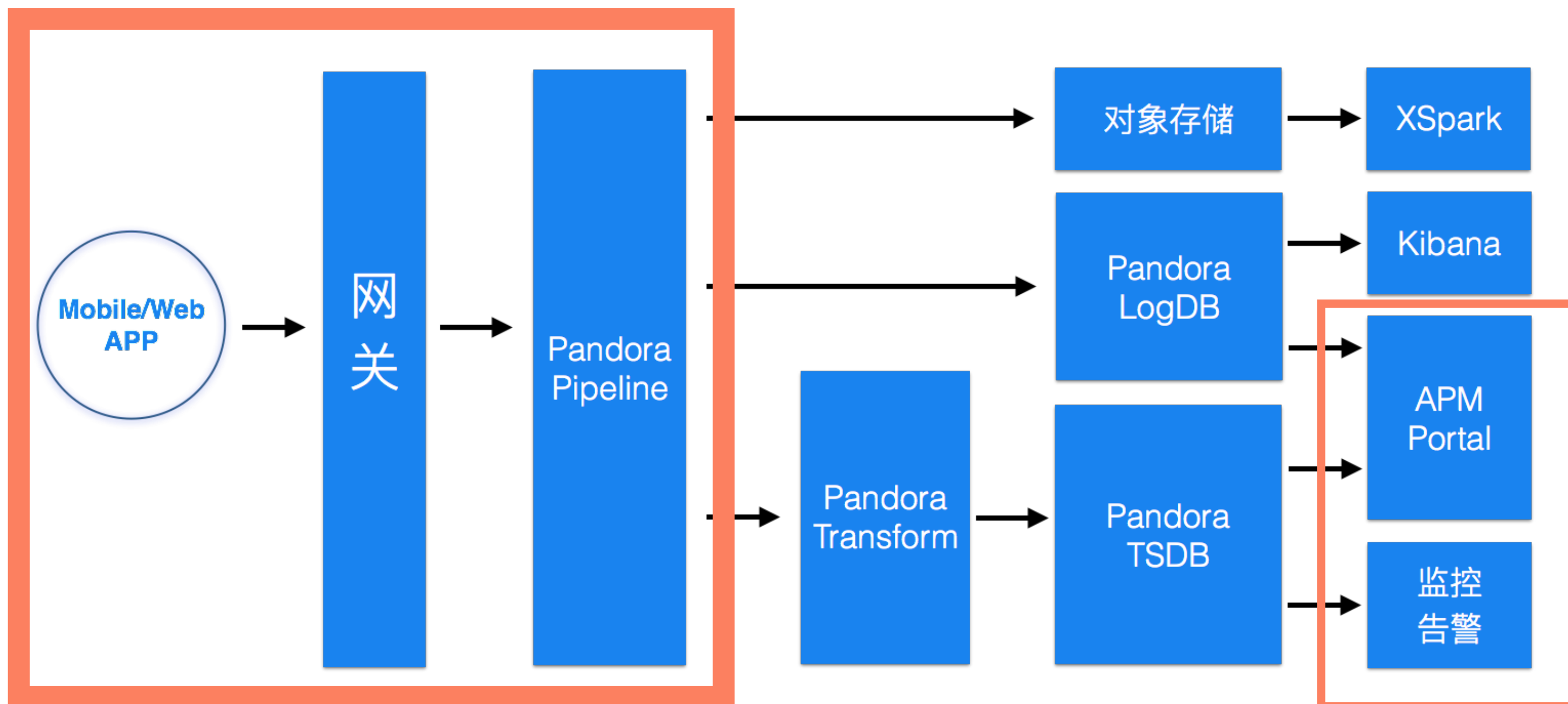
数据汇聚

管道

聚合

持久化

展示



APM 实时计算 workflow

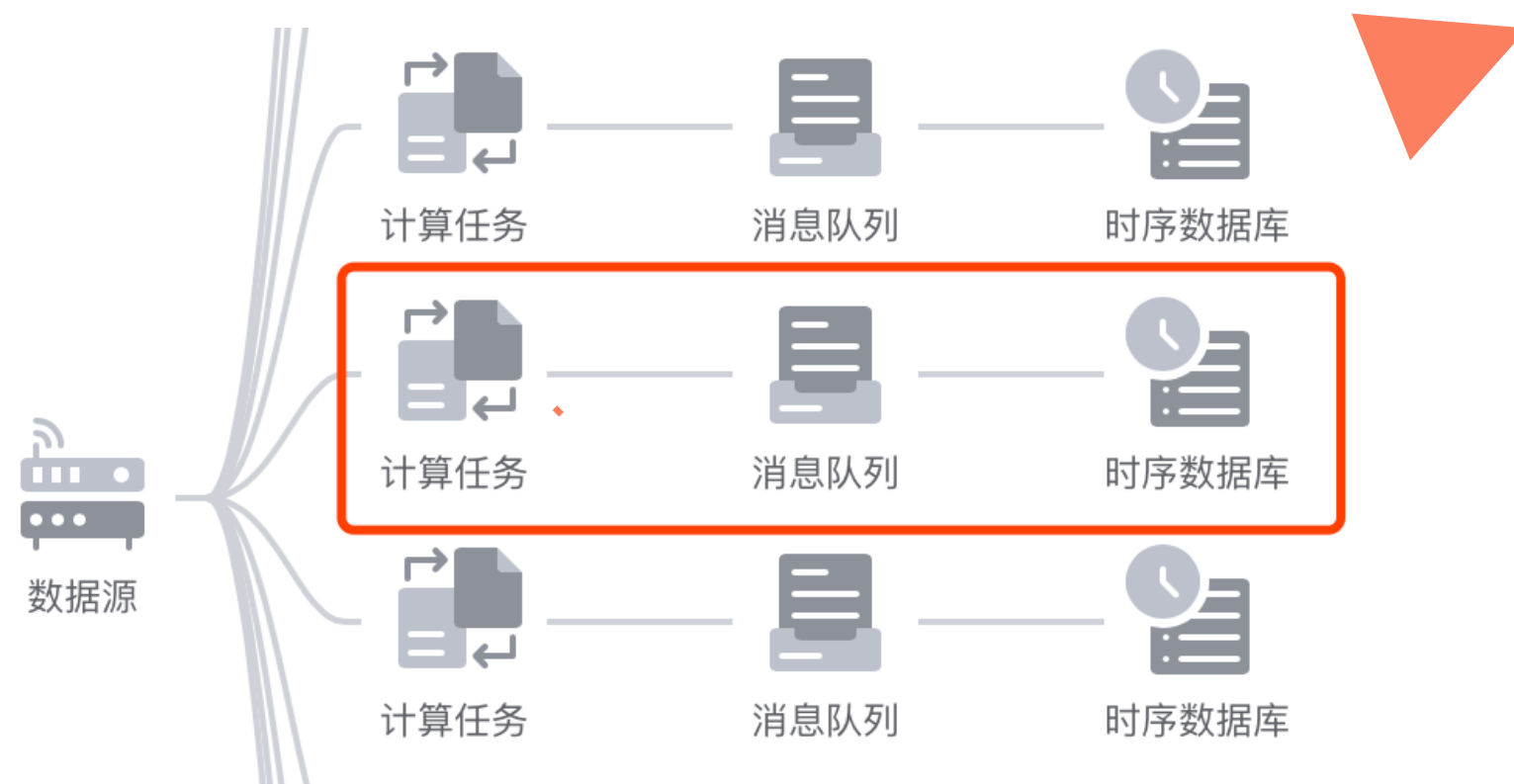


- 树状拓扑
- 每个分支代表一个统计计算或数据导出功能

直播卡顿率定义

- 每分钟上报一次，缓冲或者无缓冲
- 卡顿人数/总人数，即为该时间段内卡顿率
- 按 CDN / 运营商 / 省份 / 服务器 IP 等多种维度进行聚合

APM 实时计算 workflows



```
select
    cast(cast(time as bigint) / 5 as bigint) * 5
as time,
    cdn,
    country,
    province,
    isp,
    platform,
    sum(v4) as total,
    count(v4) as number
from stream
where
    tag = 'monitor'
group by
    cast(cast(time as bigint) / 5 as bigint) * 5,
    cdn, country, province, isp, platform
```

导出的卡顿率时序数据结构

输入SQL

select * from lag order by time desc limit 10

保存SQL语句

常用SQL





搜索结果

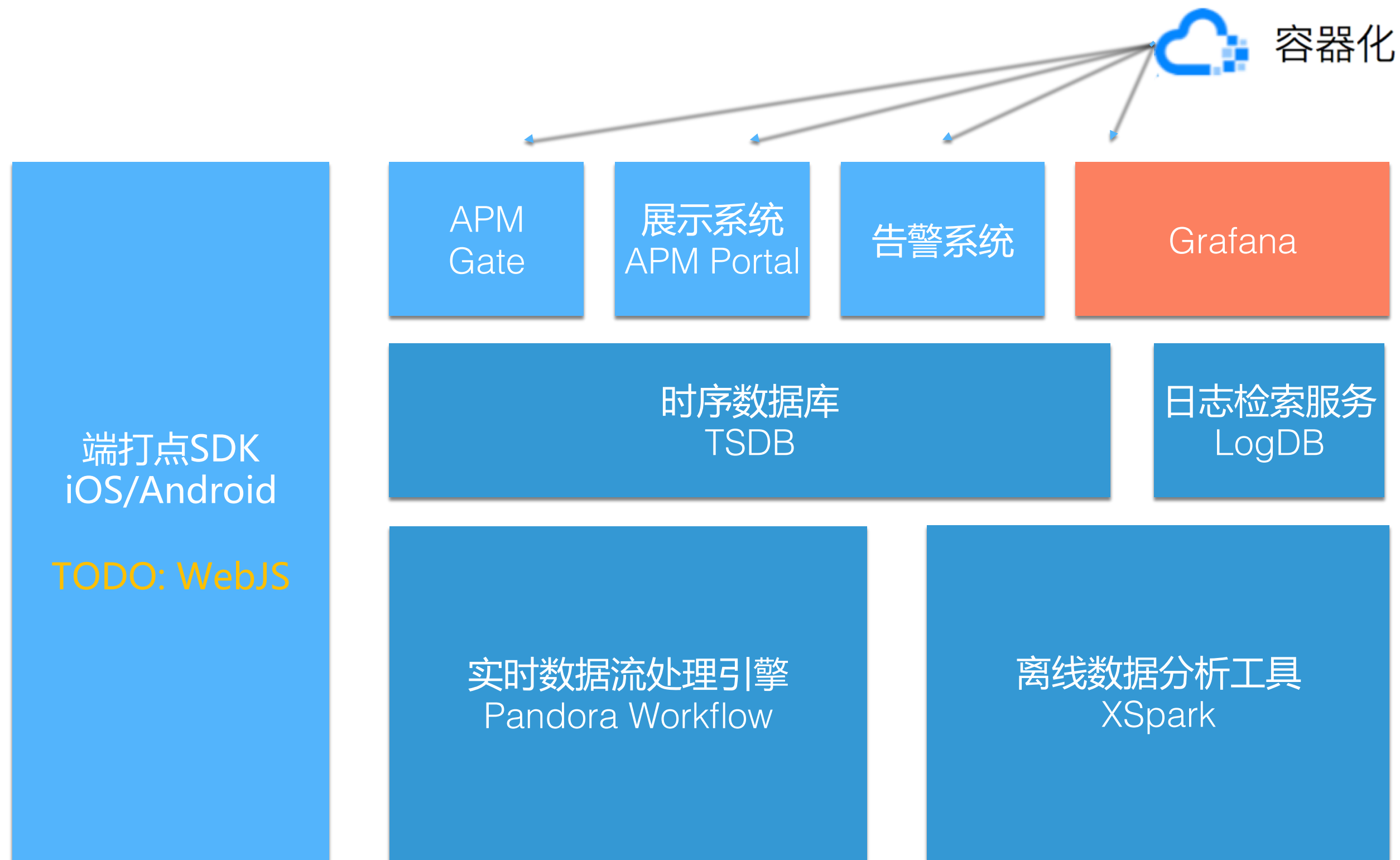
自动刷新关 ☐ 开

使用自动刷新功能按SQL使用计费,默认每分钟刷新一次。

默认显示上限为500条记录

time	cdn	country	definition	isp	number	platform	province	total
2017-07-27T06:40:25Z		中国	超清	电信	1	2	江苏	0
2017-07-27T06:40:25Z		中国	超清	电信	1	2	广西	0
2017-07-27T06:40:25Z		中国	超清	电信	1	2	广东	0
2017-07-27T06:40:25Z		中国	超清	电信	1	2	四川	0
2017-07-27T06:40:25Z		中国	超清	电信	3	1	广东	0

APM组件



吞吐量

- 每分钟实时写入的数据增量达到数百 GB，每日增量数据数百 TB
- 每分钟实时写入的数据条目达到数十亿

APM 目标

视频 APM

- 多维度 (CDN, 省份, 服务器 IP, ISP) 联合分析
- 实时监控数据
- 快速查询定位
- 快速统计单个主播终端在线数等简单运营统计

移动 APM

- 崩溃报告
- 网络性能监测
- App 响应时间监测
- 对性能指标的多维度联合分析
- 自定义统计数据分析

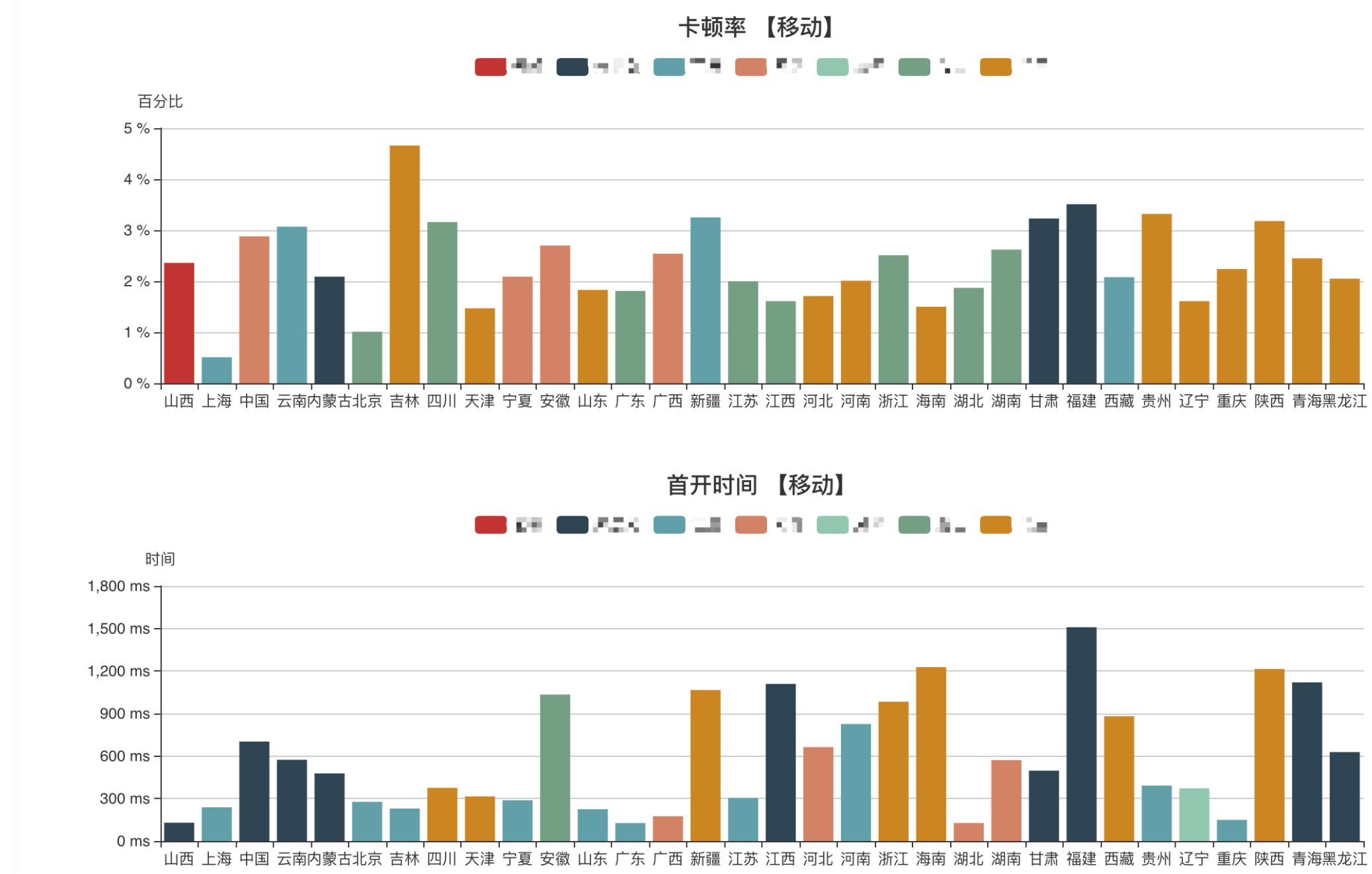
多维度质量对比

移动

全部平台

TSDDB 拉取

卡顿率、首开时间的
多维度联合展示

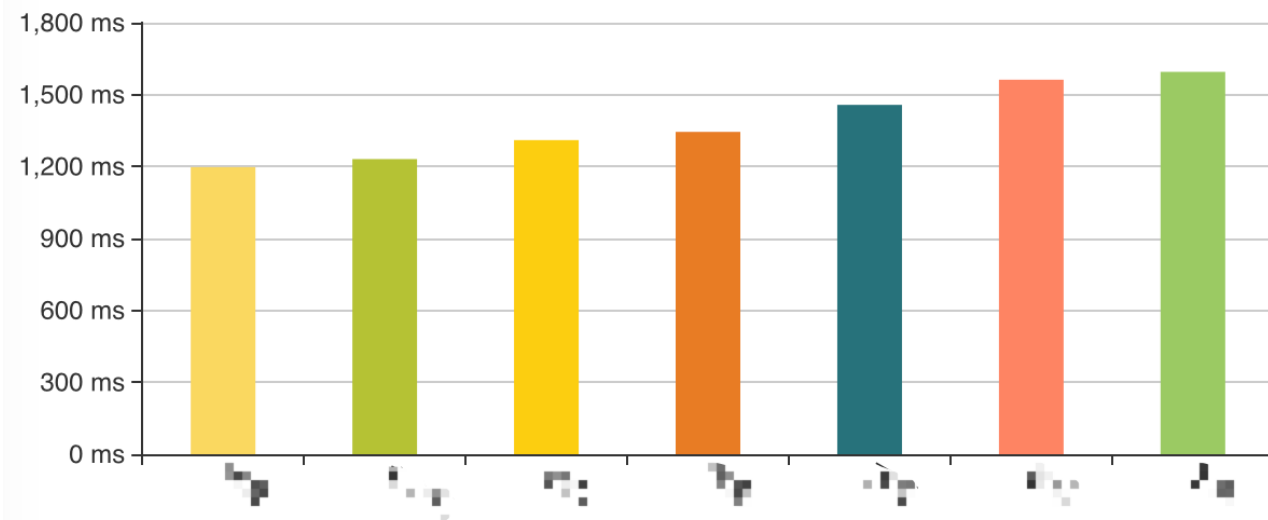


多维度质量对比

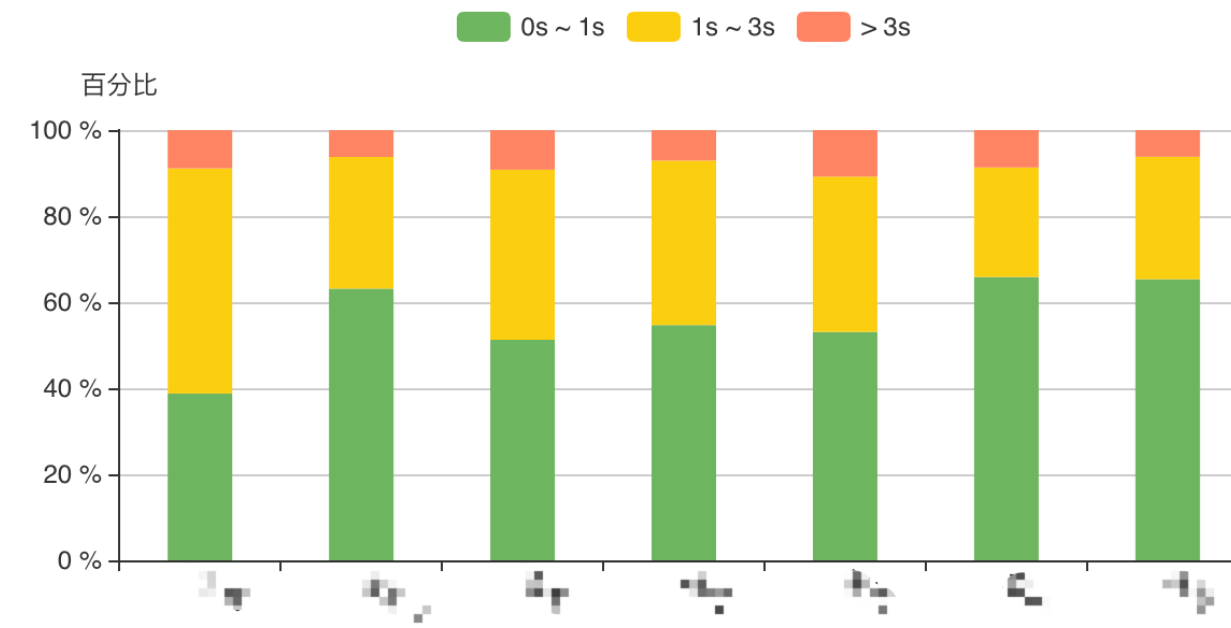
TSDDB 拉取

卡顿率、首开时间的多维度联合展示

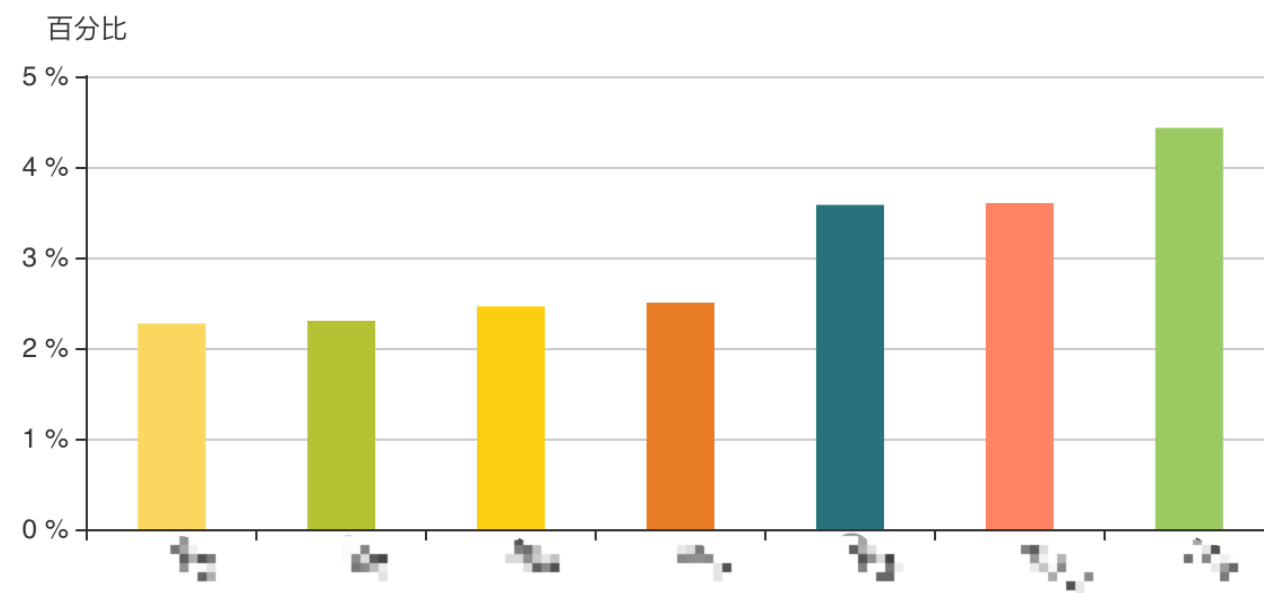
【全部运营商】 【全部省份】 【全部平台】 首开时间 竞品对比



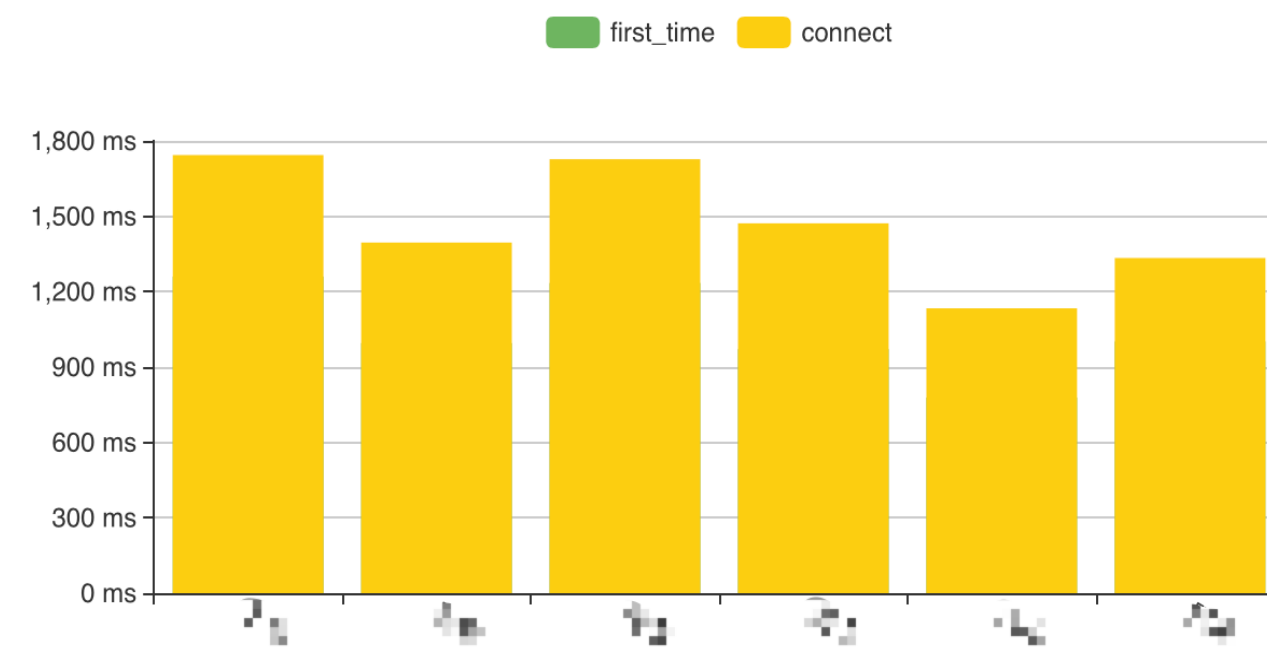
【全部运营商】 【全部省份】 【全部平台】 首开时间分布 竞品对比



【全部运营商】 【全部省份】 【全部平台】 卡顿率 竞品对比



【全部运营商】 【全部省份】 【全部平台】 速度分布 竞品对比



日志检索

输入条件

device:Av25DOiSwBEBjFCIlzDc2cl0TL2Pe1-BXgPxCwAuidGM







您的时间排序字段为: **time** 数据总量为:40 耗时:42ms [帮助文档](#)

表格 Json数据

时间排序	source
> 2017-05-22 18:27:55.000000+08:00	a: first c: cdn city: 绵阳 client_ip: 118.117.50.226 country: 中国 device: Av25DOiSwBEBjFCIlzDc2cl0TL2Pe1-BXgPxCwAuidGM isp: 电信 net: 0 p: 1 province: 四川 rcat: 5 refer: cross_room_half rid: 5288622 screen: 2 sh: 1280 sw: 720 time: 2017-05-22 18:27:55.0000 00+08:00 ua: Mozilla/5.0 (Linux; Android 5.1.1; F103S Build/LMY47V; wv) AppleW..... uid: 9296086 v1: yf v2: 110.187.3.136 v3: 超清 v4: 0 v5: 1086.....
> 2017-05-22 18:27:55.000000+08:00	a: connect c: cdn city: 绵阳 client_ip: 118.117.50.226 country: 中国 device: Av25DOiSwBEBjFCIlzDc2cl0TL2Pe1-BXgPxCwAuidGM isp: 电信 net: 0 p: 1 province: 四川 rcat: 5 refer: cross_room_half rid: 5288622 screen: 2 sh: 1280 sw: 720 time: 2017-05-22 18:27:55.0000 00+08:00 ua: Mozilla/5.0 (Linux; Android 5.1.1; F103S Build/LMY47V; wv) AppleW..... uid: 9296086 v1: yf v2: 110.187.3.136 v3: 超清 v4: 0 v5: 811.....
> 2017-05-22 18:27:11.000000+08:00	a: connect c: cdn city: 绵阳 client_ip: 118.117.50.226 country: 中国 device: Av25DOiSwBEBjFCIlzDc2cl0TL2Pe1-BXgPxCwAuidGM isp: 电信 net: 0 p: 1 province: 四川 rcat: 5 refer: cross_room_half rid: 557668 screen: 2 sh: 1280 sw: 720 time: 2017-05-22 18:27:11.00000 0+08:00 ua: Mozilla/5.0 (Linux; Android 5.1.1; F103S Build/LMY47V; wv) AppleW..... uid: 9296086 v1: yf v2: 110.187.3.134 v3: 超清 v4: 0 v5: 4561.....
> 2017-05-22 18:27:11.000000+08:00	a: first c: cdn city: 绵阳 client_ip: 118.117.50.226 country: 中国 device: Av25DOiSwBEBjFCIlzDc2cl0TL2Pe1-BXgPxCwAuidGM isp: 电信 net: 0 p: 1 province: 四川 rcat: 5 refer: cross_room_half rid: 557668 screen: 2 sh: 1280 sw: 720 time: 2017-05-22 18:27:11.00000 0+08:00 ua: Mozilla/5.0 (Linux; Android 5.1.1; F103S Build/LMY47V; wv) AppleW..... uid: 9296086 v1: yf v2: 110.187.3.134 v3: 超清 v4: 0 v5: 4718.....



利用日志检索服务做单用户排障

App 网络测速



App 崩溃日志收集

Crash log 展示

时间区间2017/07/23 ~ 2017/07/29

AppBundleId	AppId	AppName	AppVersion	CrashLogKey	CrashTime
pre-engineering.PreDemObjcDemo	00000002	PreDemObjcDemo	1.0.0	00000002/crash/3761be760faf402b8cac9222d2c5e262	2017-07-23 10:07:00
pre-engineering.PreDemObjcDemo	00000002	PreDemObjcDemo	1.0.0	00000002/crash/50025277520ee62c968	2017-07-23 10:07:00

APP 信息 平台信息 设备信息 返回

appBundled	appName	appVersion	sdkId	sdkVersion
qiniu.predem.example	AndroidPreDem	1.0.0	PreDemSDK	1.0.0

崩溃信息

at java.lang.VMThread.sleep(Native Method)

at java.lang.Thread.sleep(Thread.java:1013)

at java.lang.Thread.sleep(Thread.java:995)

at qiniu.predem.example.MainActivity.SleepAMinute(MainActivity.java:55)

at qiniu.predem.example.MainActivity.access\$300(MainActivity.java:33)

at qiniu.predem.example.MainActivity\$7.onClick(MainActivity.java:182)

at android.view.View.performClick(View.java:4438)

at android.view.View\$PerformClick.run(View.java:18422)

at android.os.Handler.handleCallback(Handler.java:733)

at android.os.Handler.dispatchMessage(Handler.java:95)

at android.os.Looper.loop(Looper.java:136)

at android.app.ActivityThread.main(ActivityThread.java:5001)

LogDB 检索

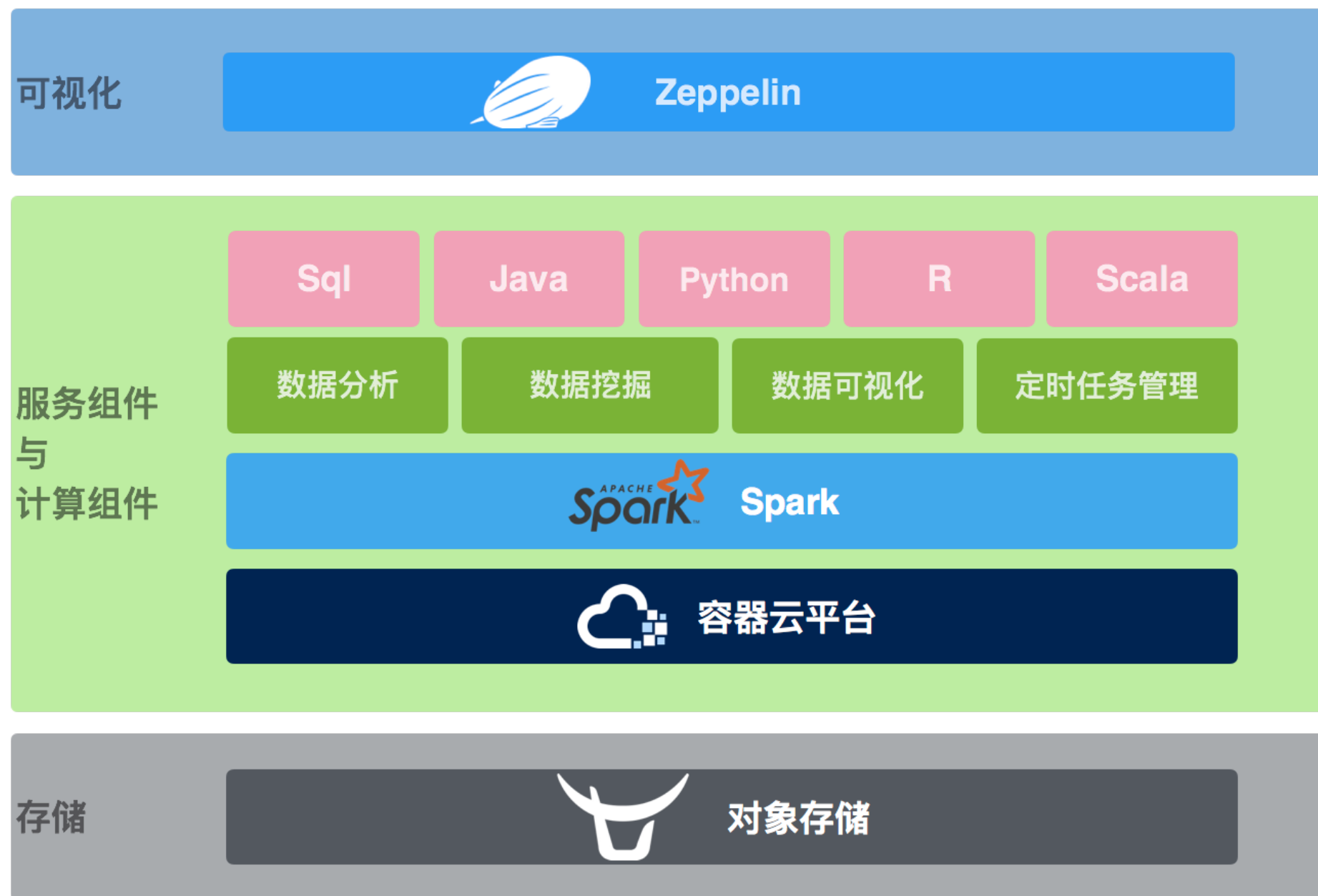
自定义事件上报



- 自动创建 pipeline
- 自动生成 export

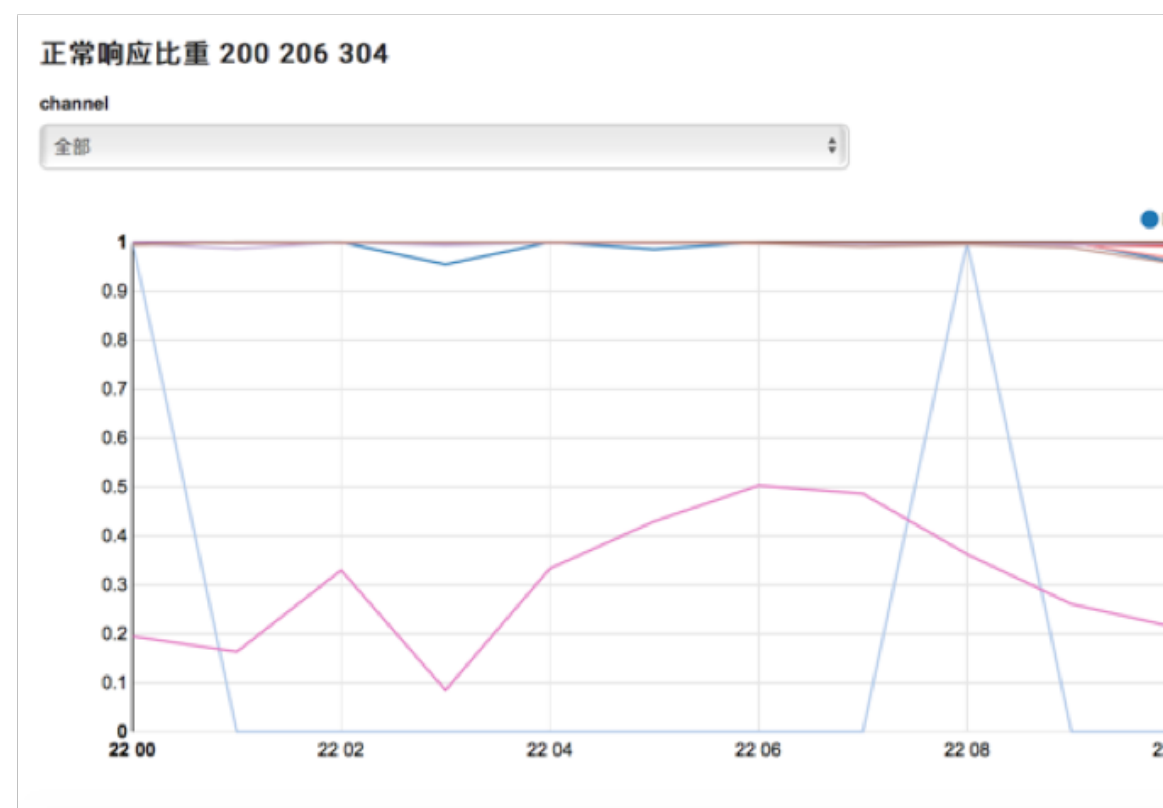
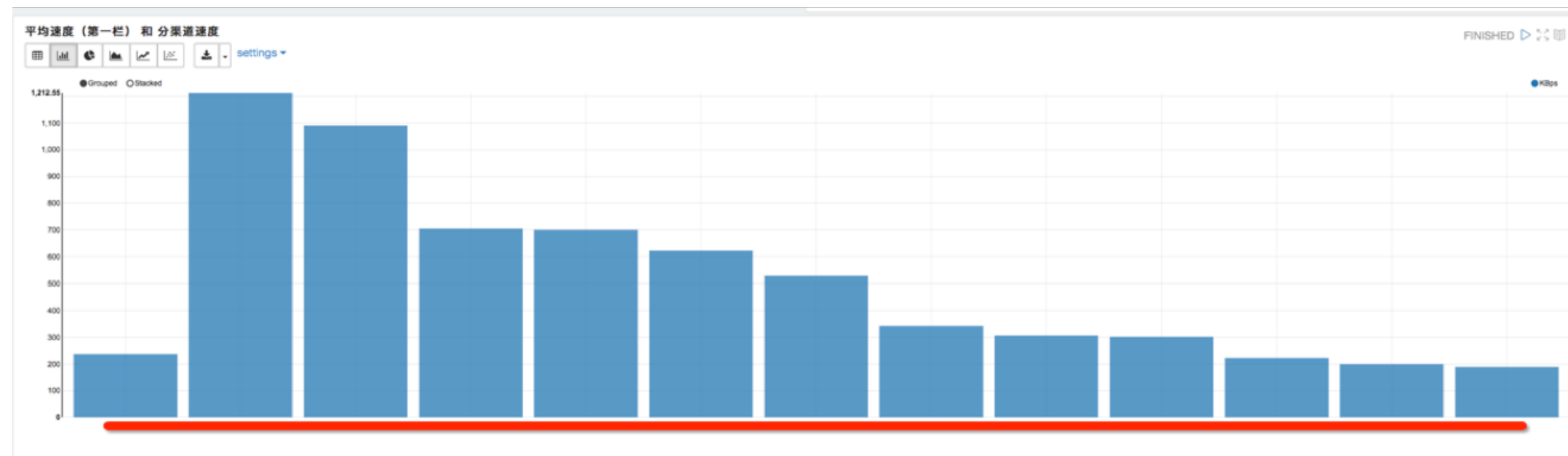
离线分析工具

XSpark 架构



- 日志存放在对象存储
- Spark 分析
- Zeppelin 可视化呈现

离线日志分析



- 日活
- 按省份、运营商、设备、系统等维度的关联分析

监控告警

- 基于 TSDB , 利用 Grafana 做告警
- 基于 TSDB , 利用 Kapacitor 进行实时聚合计算+告警

监控告警

遇到的问题

- TSDB 适用于做查询，高频的聚合计算极耗 CPU/内存，特别是 Tag 极多的情况
- Kapacitor TICKScript 学习成本高，定义告警任务麻烦

监控告警

- 基于 TSDB , 利用 Grafana 做告警
- 基于 TSDB , 利用 Kapacitor 进行实时聚合计算+告警 , 降低告警任务创建门槛
- 基于 Pipeline , 利用实时计算做复杂运算告警

监控告警

TEST ALERT

Select a Time Series

```
SELECT "usage_idle" FROM "kapacitor_example"."autogen"."cpu" WHERE time > now() - 1h AND "cpu"='cpu-total'
```

Databases	Measurements	Fields
kapacitor_example.autogen	cpu	usage_guest, usage_guest_nice, usage_idle, usage_iowait, usage_irq, usage_nice, usage_softirq

Rule Conditions

Alert Type: Threshold

Send Alert where: usage_idle is less than 0.4

Alert Message

Send this Alert to: smtp http tcp exec

Email Addresses (Separated by Spaces): huangdeheng@qiniu.com

Hello world



计算任务

名称: transform_presniff_monitor_task_2017_07_13_14_01_48

容器类型: M16C4 容器数量: 1

☐ 自定义计算 ☒ SQL 计算

SQL 代码:

```
SELECT * FROM (SELECT cast(cast(time as bigint)/30 as bigint)*30 AS time, sum(networkErrorCount) AS value, domain, COUNT(*) AS count FROM stream GROUP BY domain,
```

- 基于 TSDB
- 利用 Kapacitor 进行实时聚合计算+告警

- 基于 Pipeline
- 导出运算结果到 TSDB，利用 Kapacitor 告警

七牛 APM 产品小结

- 构建于七牛 Pandora 这个大数据 PaaS 之上的 SaaS
- 深耕数字化体验监控 (DEM)
- 可定制化的运维、运营数据分析呈现工具
- 开放、自由

展望

- 专注于 DEM 的第三层次，结合更多机器学习的数据分析
- 更智能的告警



简单 · 可信赖

谢谢 | Q&A