



GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE

# 基于flink的实时指标聚合系统(Petra)实践

鞠大升 美团点评 研究员



# GIAC

## 全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE



关注msup  
公众号获得  
更多案例实践

GIAC 是中国互联网技术领域行业盛事，组委会从互联网架构最热门领域甄选前沿的有典型代表的技术创新及研发实践的架构案例，分享他们在本年度最值得总结、盘点的实践启示。

2018年11月 | 上海国际会议中心



高可用架构  
改变互联网  
的构建方式

# 大纲



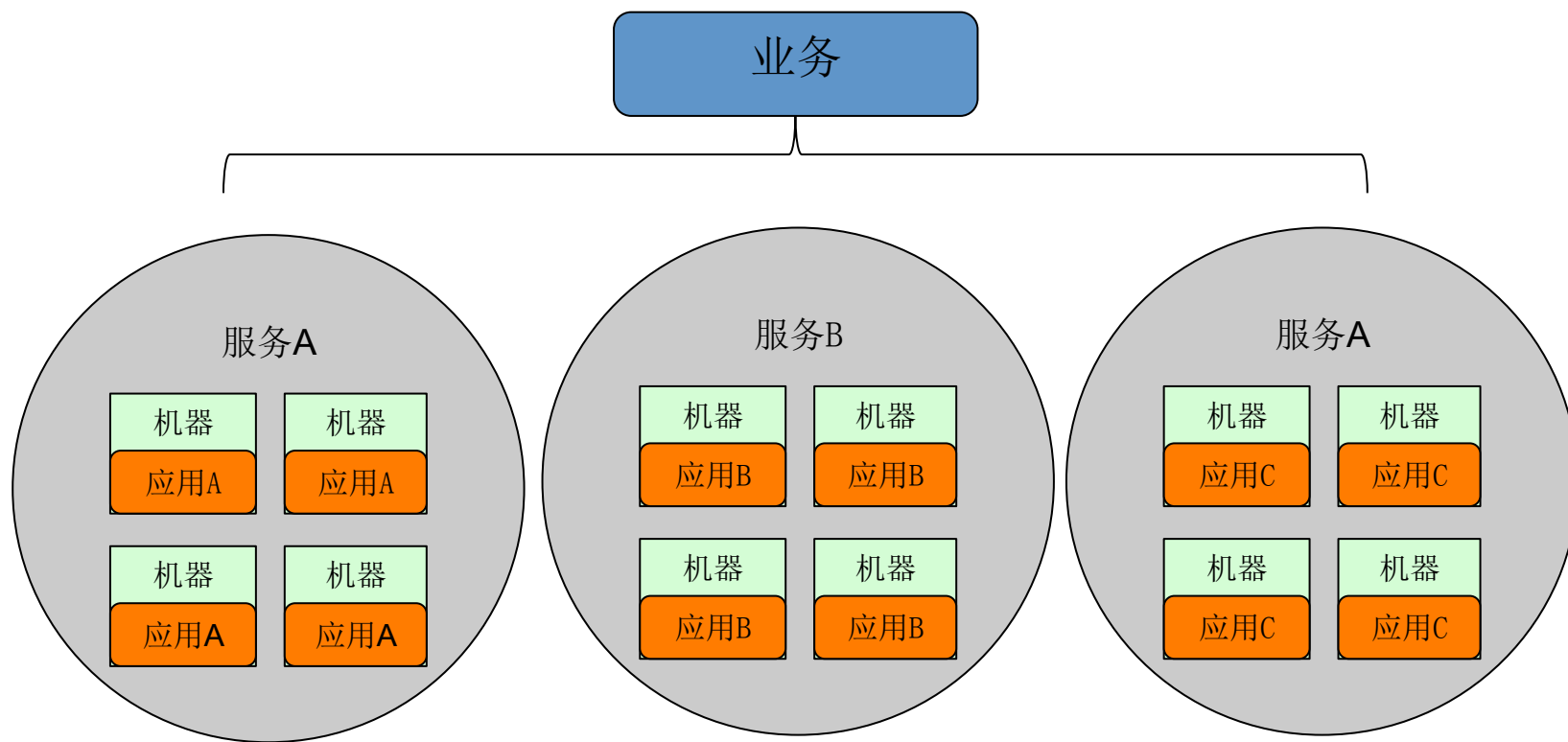
- ✓ 背景和问题
- ✓ 系统分析和挑战
- ✓ 系统架构和演进
- ✓ 应用案例
- ✓ 未来

# 大纲



- ✓ 背景和问题
- ✓ 系统分析和挑战
- ✓ 系统架构和演进
- ✓ 应用案例
- ✓ 未来

# 如何监控业务指标？



如何从系统、应用、服务、业务四个层面进行监控？

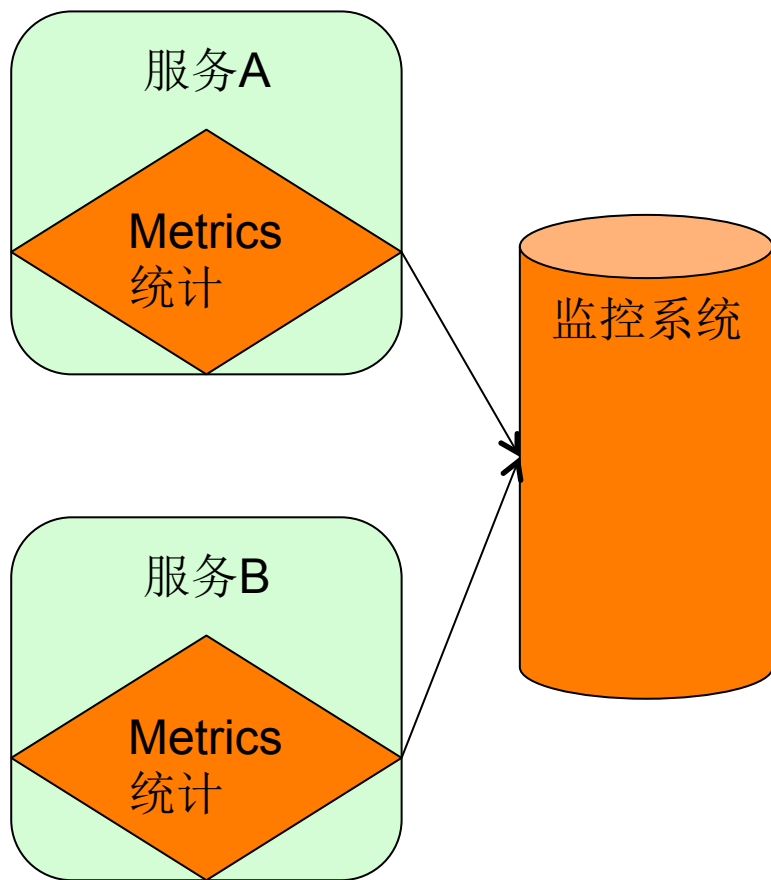
# 如何监控业务指标？



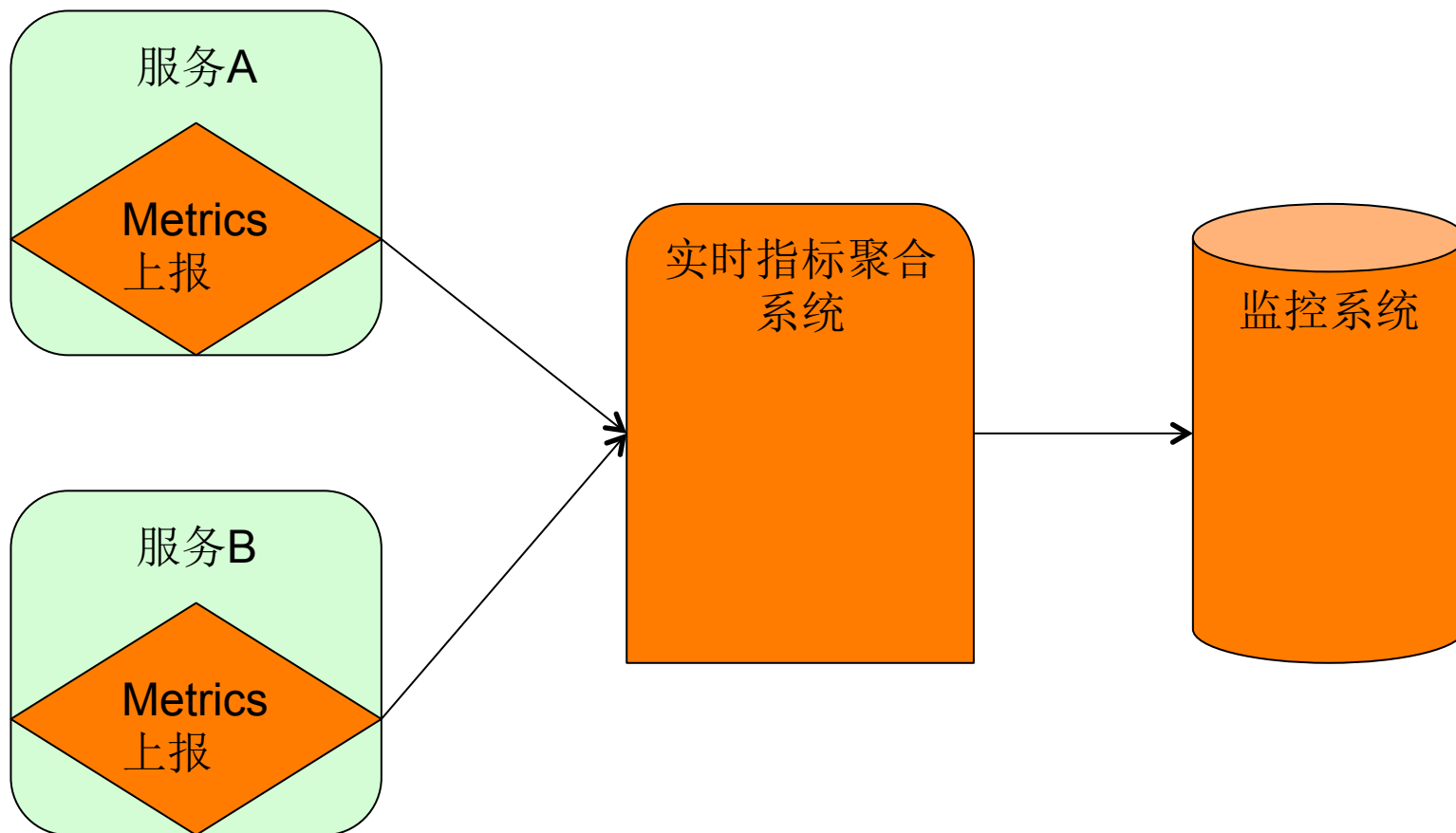
对象	监控内容	特点	解决方案
业务	1. 业务指标	1. 跨服务统计 2. 业务指标、纬度不固定	?
应用、服务	1. 可用性指标 2. 性能指标 3. 业务指标	1. 应用、服务纬度 2. 指标、纬度固定 3. 业务指标、纬度不固定	服务框架、服务治理系统
系统	1. 机器指标 2. 系统指标	1. 机器纬度 2. 指标、纬度固定	成熟的监控系统，如open-falcon, zabbix等

# 一般解决方案

- ✓ 每个服务有自己独特的指标统计模块，定时汇报业务指标
- ✓ 问题
  - ✓ 指标聚合逻辑耦合在业务逻辑中，导致更新困难
  - ✓ 多纬度聚合引入大量代码
  - ✓ 不能实现跨服务的聚合



# 实时指标聚合(Petra)方案





# 实时指标聚合(Petra)方案



## ✓ 优势

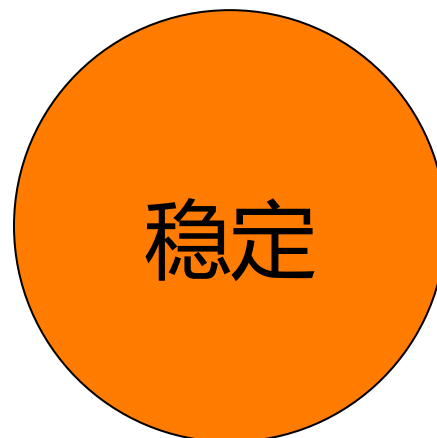
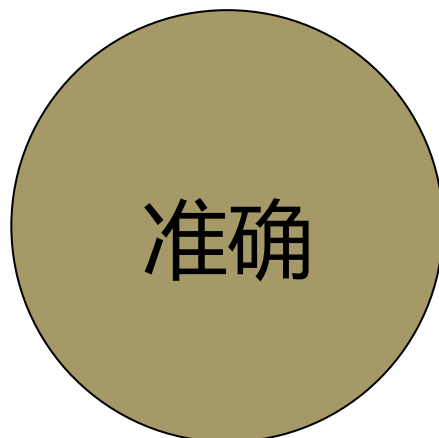
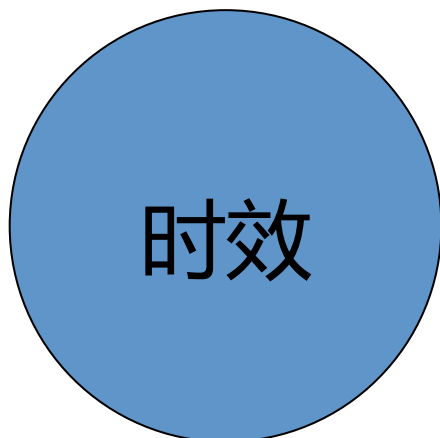
- ✓ 业务逻辑和指标聚合逻辑解耦；
- ✓ 支持多纬度聚合；
- ✓ 支持实时聚合；
- ✓ 支持跨服务的聚合；

# 大纲

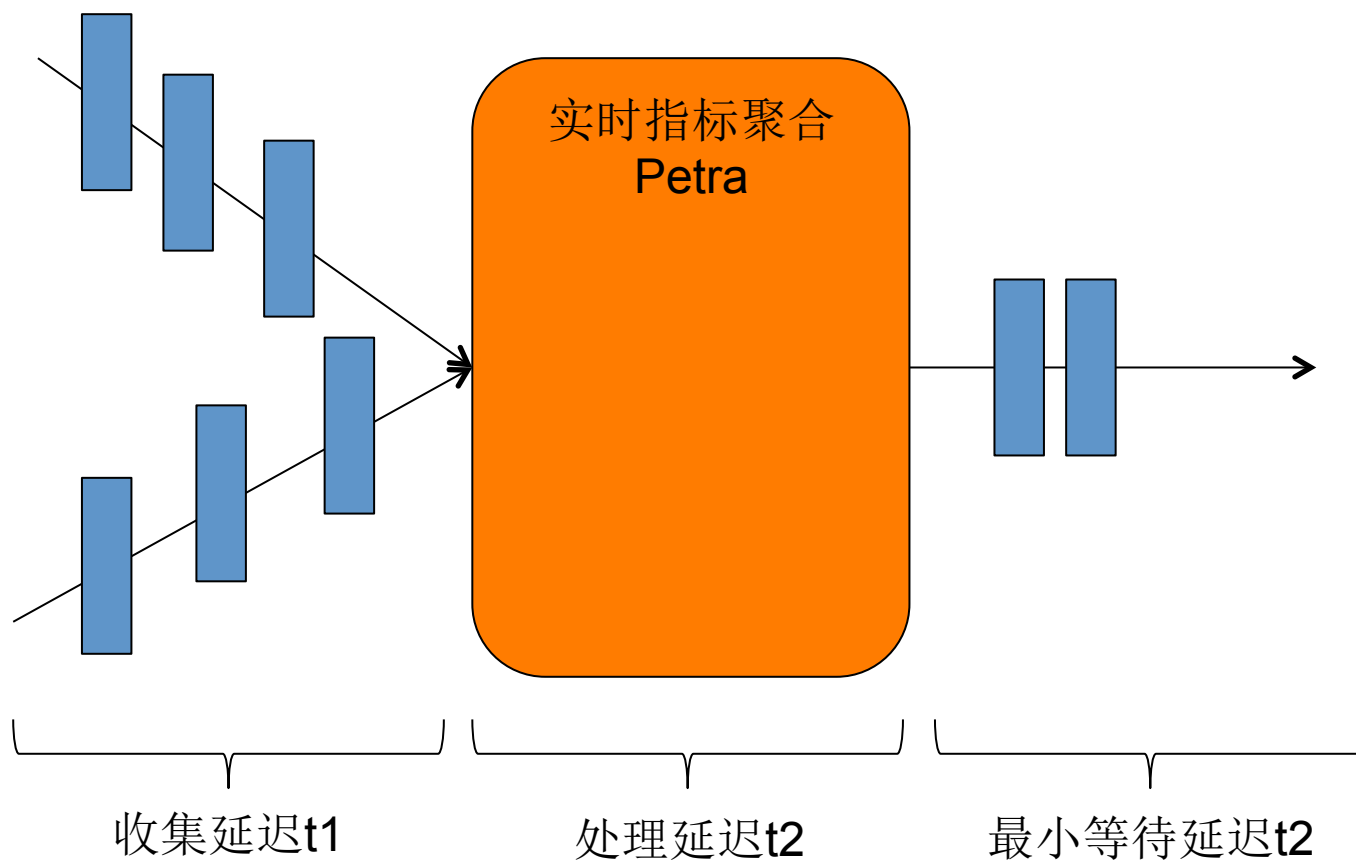


- ✓ 背景和问题
- ✓ 系统分析和挑战
- ✓ 系统架构和演进
- ✓ 应用案例
- ✓ 未来

# Petra需要满足什么



# Petra的挑战 - 时效



# Petra的挑战 - 时效

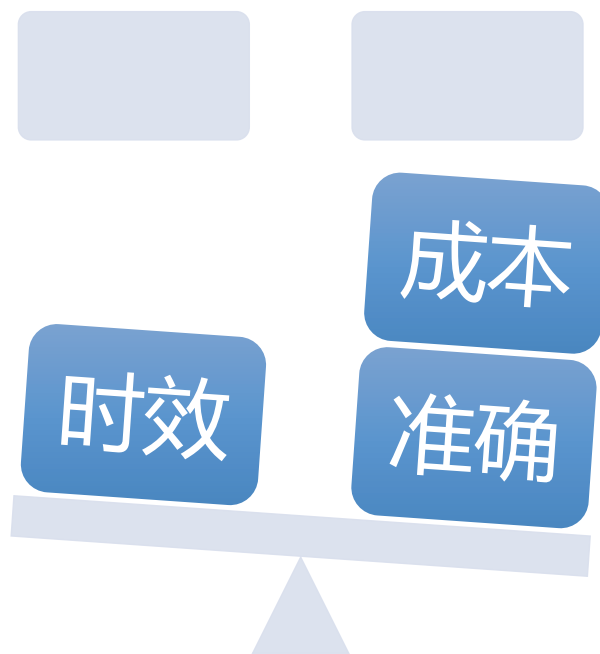


- ✓ 定义：延迟 $t = \text{收集延迟}t_1 + \text{处理延迟}t_2 + \text{等待延迟}t_3$
- ✓ 收集延迟 $t_1$ ，取决于效率权衡，可以到秒级，甚至毫秒；
- ✓ 处理延迟 $t_2$ ，取决于聚合时间纬度的粒度，10秒、30秒、1分钟、5分钟、10分钟；
- ✓ 等待延迟 $t_3$ ，取决于对于『晚到数据』的容忍度，要时效还是准确？

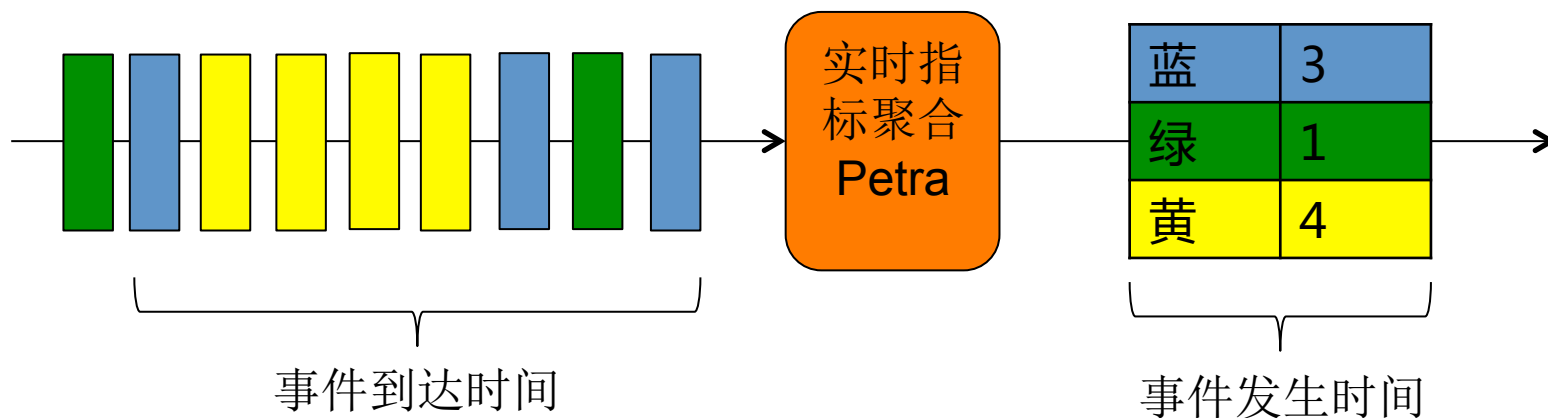
# Petra的挑战 - 时效

## ✓ 挑战

- ✓ 收集数据，时效和成本的权衡；
- ✓ 处理晚到/乱序数据，时效和准确的权衡；



# Petra的挑战 - 准确



- ✓ Exactly once : 事件仅仅被统计一次 ;
- ✓ 晚到数据 : 事件到达时间, 晚于统计窗口, 导致被『抛弃』 ;

# Petra的挑战 – 准确



- ✓ Exactly once
  - ✓ 收集阶段什么机制保证不丢？如何处理系统异常？
  - ✓ 计算阶段什么机制保证不丢？如何处理系统异常？
- ✓ 晚到数据
  - ✓ 对于超出时间窗口的晚到数据，是否需要回补？对不同的场景（展示、报警）回补的形式？
- ✓ 审计
  - ✓ 作为一个统计系统，如何自证？



# Petra的挑战 – 稳定



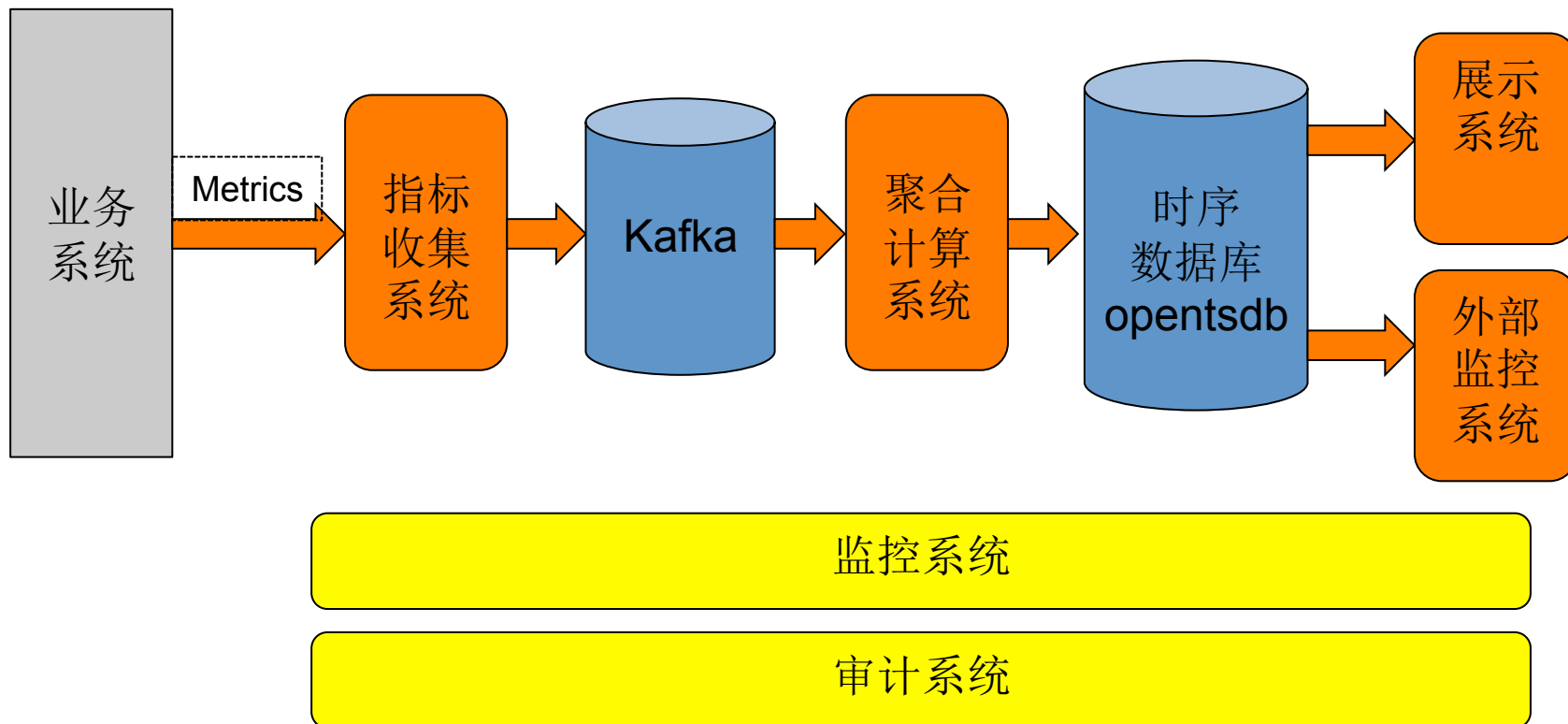
- ✓ 计算稳定性
  - ✓ 作业运行环境 ( Yarn/Hdfs ) 稳定 ;
  - ✓ State存储的稳定 ;
  - ✓ Metrics的纬度值倾斜导致的计算瓶颈问题 ;
- ✓ 存储稳定性
  - ✓ Metrcis的纬度值空间大导致的时序数据库问题 ;

# 大纲



- ✓ 背景和问题
- ✓ 系统分析和挑战
- ✓ 系统架构和演进
- ✓ 应用案例
- ✓ 未来

# Petra系统设计



# Metrics定义

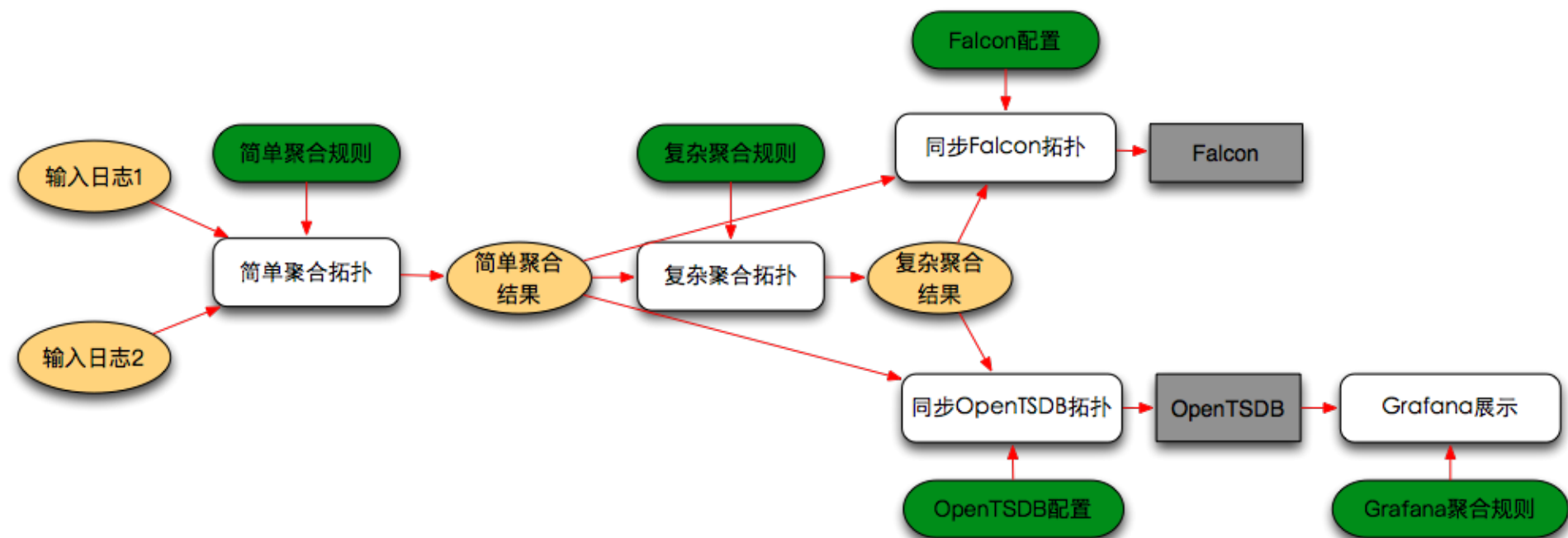
```
{  
  "kvs": {  
    "sucess_count": [10, 11],  
    "fail_count": [1]  
  },  
  "tags": {  
    "hostname": "a.b.c.com",  
    "idc": "dx"  
  },  
  "ts": 1431502034  
}
```

指标项，支持多个key/value对

纬度项，标识指标项所属纬度

时间戳，单位秒

# 聚合计算



# Exactly Once

## ✓ Storm

- ✓ At most once , 性能好 , 可丢 , 导致『丢点』问题 ;
- ✓ At least once , 性能一般 , 不丢可重 , failover时导致『延迟』 , 甚至『丢点』问题 ;
- ✓ Trident , 性能差 , 维护State存储成本高 ;

## ✓ Flink

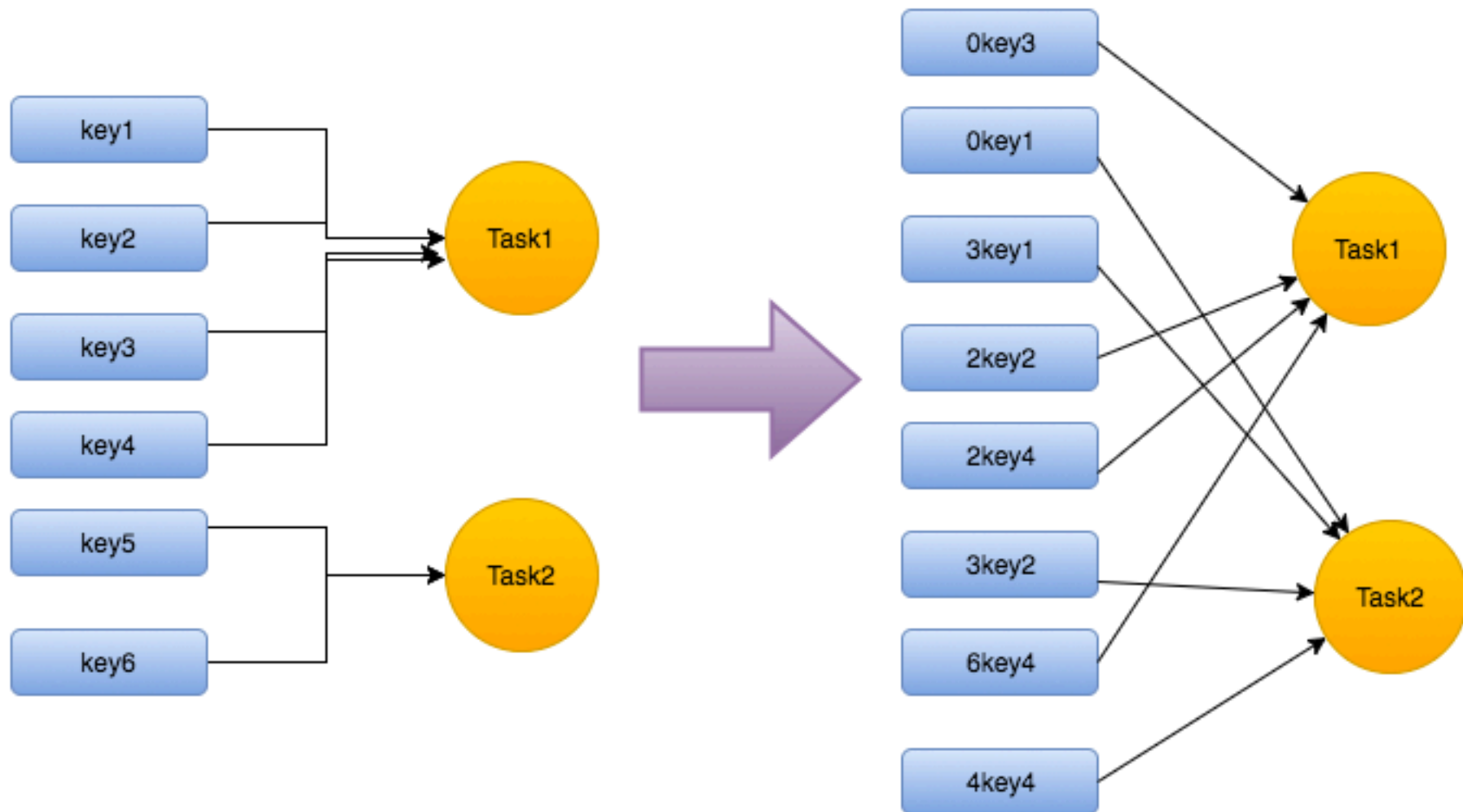
- ✓ 通过checkpoint机制来保证 , 不用自己维护State ;
- ✓ failover时延迟是两个checkpoint间隔 ;

保证准确性的成本降低 !

- ✓ 晚到/乱序数据
  - ✓ 触发旧窗口计算，计算性能下降；
  - ✓ 结果修正，不同下游（展示和报警）的处理方式不一样；
- ✓ 容忍晚到数据窗口大小
  - ✓ 决定State的大小，影响checkpoint和恢复；
  - ✓ 决定内存大小；

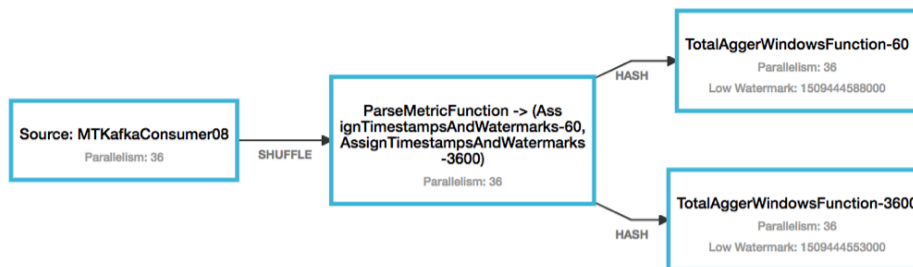
谨慎决定是否容忍晚到数据，以及窗口大小

# Metrics纬度值倾斜



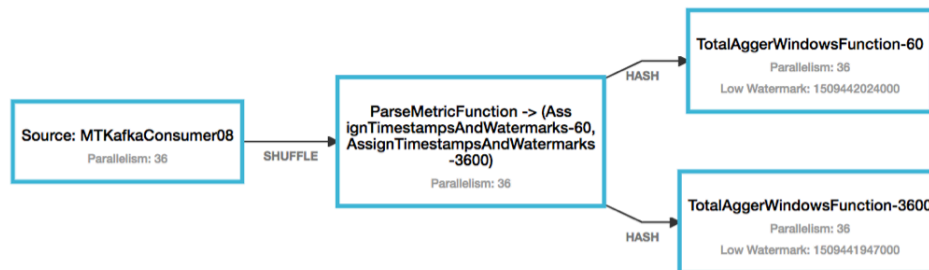


# Metrics纬度值倾斜



Start Time	End Time	Duration	Name	Bytes received	Records received	Bytes sent	Records sent	Parallelism	Tasks	Status
2017-11-02, 15:03:08	2017-11-02, 15:09:41	6m 33s	Source: MTKafkaConsumer08	0 B	0	1.36 GB	14,283,930	36	<div><div>0</div><div>0</div><div>36</div><div>0</div><div>0</div><div>0</div></div>	RUNNING
2017-11-02, 15:03:08	2017-11-02, 15:09:41	6m 33s	ParseMetricFunction -> (AssignTimestampsAndWatermarks-60, AssignTimestampsAndWatermarks-3600)	1.32 GB	13,834,495	6.30 GB	27,668,967	36	<div><div>0</div><div>0</div><div>36</div><div>0</div><div>0</div><div>0</div></div>	RUNNING
2017-11-02, 15:03:08	2017-11-02, 15:09:41	6m 33s	TotalAggerWindowsFunction-60	3.15 GB	13,832,772	0 B	1,753,867	36	<div><div>0</div><div>0</div><div>36</div><div>0</div><div>0</div><div>0</div></div>	RUNNING

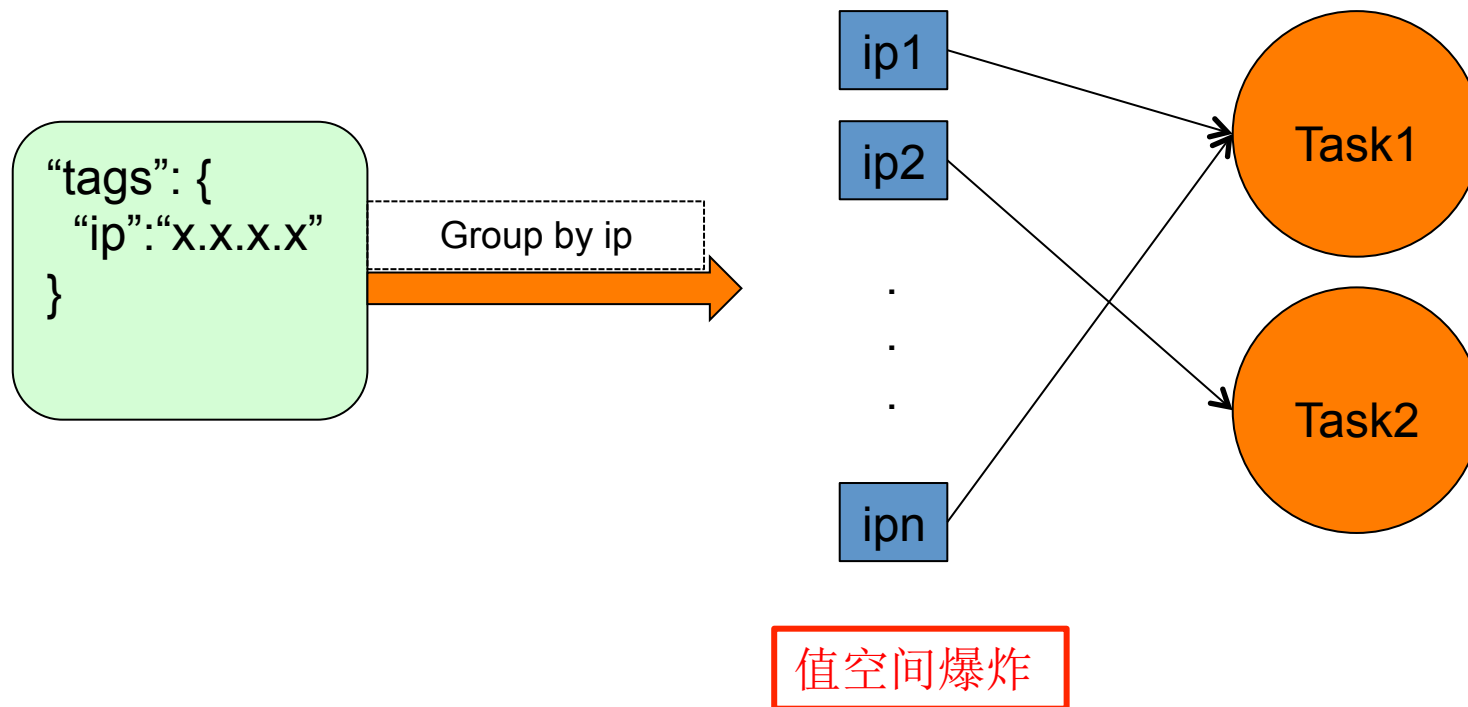
# Metrics纬度值倾斜



Subtasks   Subtasks by TaskManager   Task Metrics   Watermarks   Accumulators   Checkpoints   Back Pressure

Start Time	End Time	Duration	Name	Bytes received	Records received	Bytes sent	Records sent	Parallelism	Tasks	Status
2017-11-02, 14:22:40	2017-11-02, 14:29:04	6m 23s	Source: MTKafkaConsumer08	0 B	0	5.45 GB	57,481,199	36	<div><div>0</div><div>0</div><div>36</div><div>0</div><div>0</div></div>	RUNNING
2017-11-02, 14:22:40	2017-11-02, 14:29:04	6m 23s	ParseMetricFunction -> AssignTimestampsAndWatermarks-60, AssignTimestampsAndWatermarks-3600)	5.41 GB	57,035,222	27.2 GB	114,070,413	36	<div><div>0</div><div>0</div><div>36</div><div>0</div><div>0</div></div>	RUNNING
2017-11-02, 14:22:40	2017-11-02, 14:29:04	6m 23s	TotalAggerWindowsFunction-60	13.6 GB	57,005,543	0 B	7,122,781	36	<div><div>0</div><div>0</div><div>36</div><div>0</div><div>0</div></div>	RUNNING

# Metrics纬度值空间大



- ✓ Task中，State变大，内存大小和性能问题
- ✓ 结果存储中，时序数据库Key变多，性能问题

- ✓ 计算环境相关
  - ✓ 离线/实时系统yarn/hdfs部署隔离
  - ✓ Yarn/hdfs重启
- ✓ State相关
  - ✓ State checkpoint失败
- ✓ 存储相关
  - ✓ Kafka leader切换，导致作业挂掉
  - ✓ 和外部存储交互慢，导致数据堆积
  - ✓ Rocksdb依赖的磁盘的稳定性

# 聚合计算系统Petra



应用名称	unionid		
责任人		状态	上线
创建时间	2016-03-07 14:34:08	修改时间	2018-03-21 12:48:15
应用描述			
文档链接			

应用配置

输入配置

简单聚合配置

复杂聚合配置

Falcon同步配置

结果信息查看

线上数据追踪

聚合规则配置

✕ 删除配置

+ 新增配置

批量导入

Search

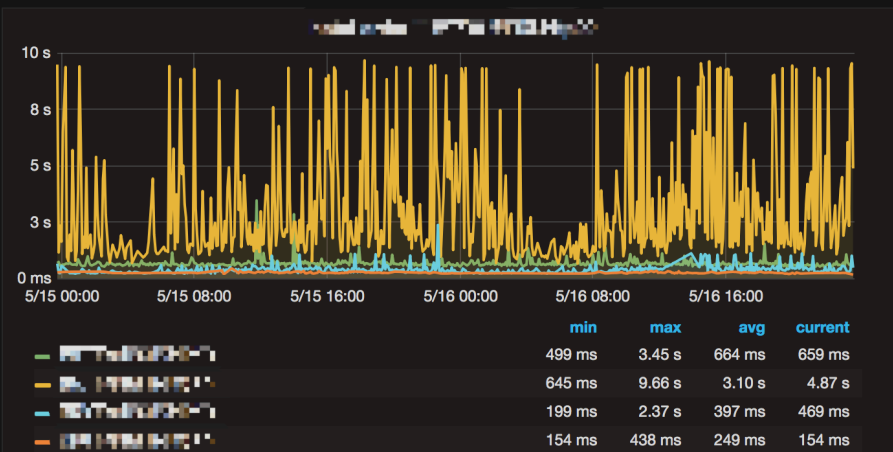
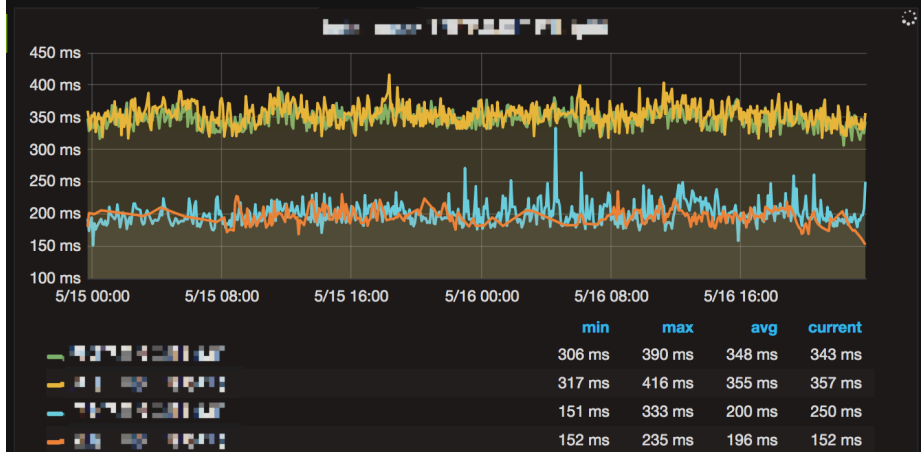
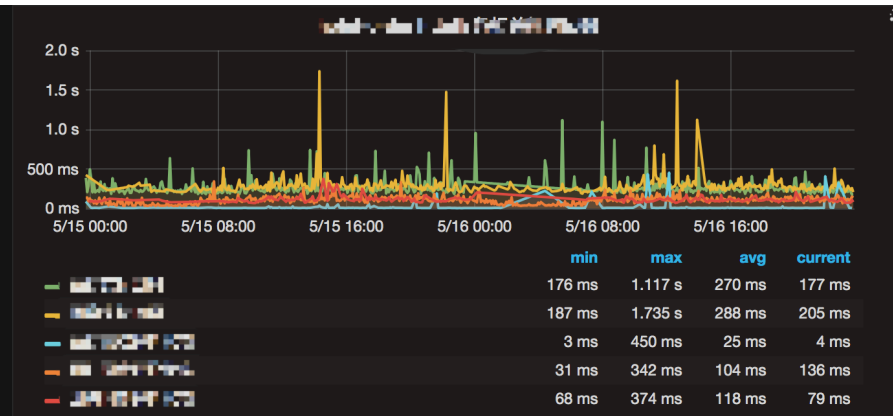
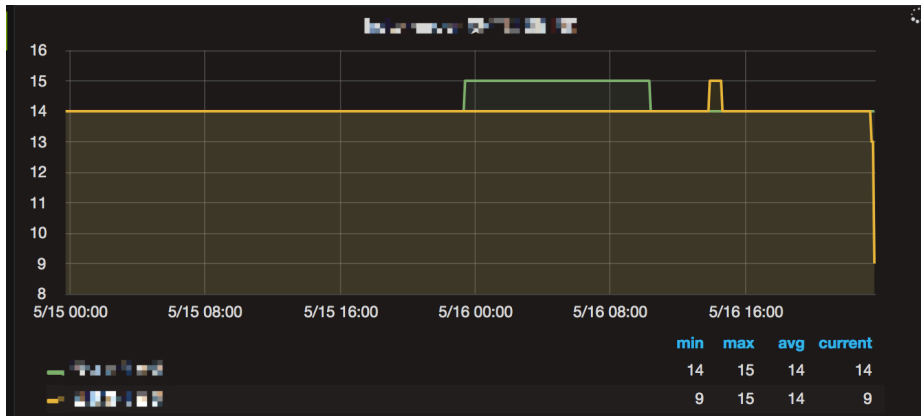
<input type="checkbox"/>	metric名称	聚合类型	聚合维度	聚合粒度	数据延迟时间	同步falcon	操作
<input type="checkbox"/>	<a href="#">count</a>	<a href="#">counter</a>	<a href="#">host, method, sub_name</a>	<a href="#">60</a>	<a href="#">Empty</a>	<a href="#">False</a>	

Showing 1 to 1 of 1 rows

数据过滤配置

流量细分配置

# 聚合计算系统Petra



# 大纲



- ✓ 背景和问题
- ✓ 系统分析和挑战
- ✓ 系统架构和演进
- ✓ **应用案例**
- ✓ 未来

# 业务监控系统 - 哨兵

- ✓ 整体大盘：整体大盘、业务大盘
- ✓ 交互监控：app监控、fe监控
- ✓ 系统运维：业务报警趋势、慢查询趋势、部署时间线、系统大盘、公网入口切换、emp功能降级...
- ✓ 智能报警：样本标注、报警配置
- ✓ 策略监控：实时特征监控、性能监控、效果监控



# 业务监控系统 - 哨兵



Watcher

业务监控

智能预警

交互监控

系统运维

配置管理

我的收藏

帮助

查找页面...



数据BI

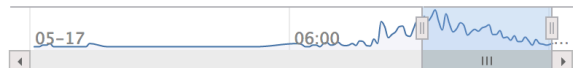
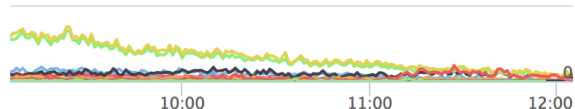
变更布局

通用查询QPM

2018/05/17



Zoom 1h 3h 6h 0.5d 1d 全部



— data.statCommApi.request.count{method=beanListAndSum}  
— data.statCommApi.request.count{method=checkCountBean}  
— data.statCommApi.request.count{method=checkCountGroup}

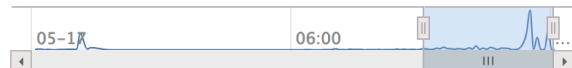
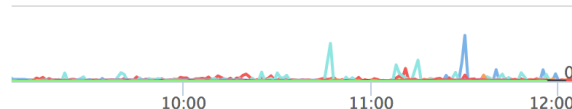
▲ 1/8 ▼

通用查询耗时

2018/05/17



Zoom 1h 3h 6h 0.5d 1d 全部



— data.statCommApi.request.cost{method=beanListAndSum}  
— data.statCommApi.request.cost{method=checkCountBean}  
— data.statCommApi.request.cost{method=checkCountGroup}

▲ 1/8 ▼

运营监测(物理)QPM

2018/05/17



Zoom 1h 3h 6h 0.5d 1d 全部



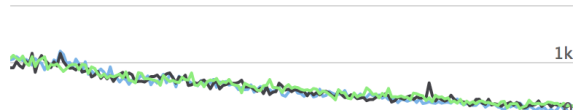
— 20180517.data.statCommApi.request.count{dim=1}  
— 20180516.data.statCommApi.request.count{dim=1}  
— 20180510.data.statCommApi.request.count{dim=1}

运营监测(组织)QPM

2018/05/17



Zoom 1h 3h 6h 0.5d 1d 全部

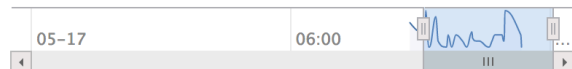
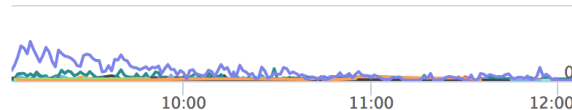


各产品QPM

2018/05/17



Zoom 1h 3h 6h 0.5d 1d 全部

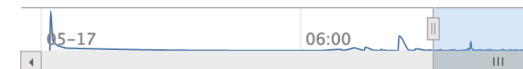
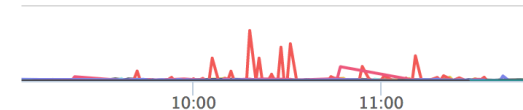


各产品耗时

2018/05/17



Zoom 1h 3h 6h 0.5d 1d 全部



# 大纲



- ✓ 背景和问题
- ✓ 系统分析和挑战
- ✓ 系统架构和演进
- ✓ 应用案例
- ✓ 未来

- ✓ 资源利用率
  - ✓ flink(exactly once)的资源使用量是storm(at least once)的5倍，state性能不够(rocksdb)
- ✓ State管理
  - ✓ 大state是瓶颈；state管理；
- ✓ 外部IO性能差
- ✓ Debug困难
  - ✓ 日志查找、用户自定义metrics
- ✓ SQL

# GIAC

## 全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE



关注公众号获得  
更多案例实践