

# 实时BI监控架构与实践

杨旋 陶然

滴滴

滴滴一下 美好出行

13a

13a

# 目录

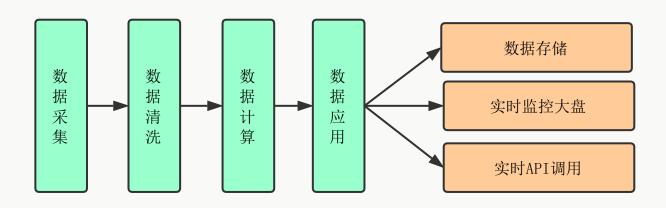
- 1. BI实时监控背景
- 2. BI实时监控架构
- 3. Spark Streaming实践
- 4. Druid实践
- 5. 未来规划

## 1 实时BI监控项目背景

- 滴滴每天会有上千万的成交单量,所以要面对非常复杂的业务场景。
- 如何满足滴滴各个业务线获取实时业务指标需求进行运营和决策?
- 如何对复杂业务场景进行高效,精准,多维度组合和细粒度的实时数据监控,并针对各种不同情况进行实时的风险控制和报警提醒呢?

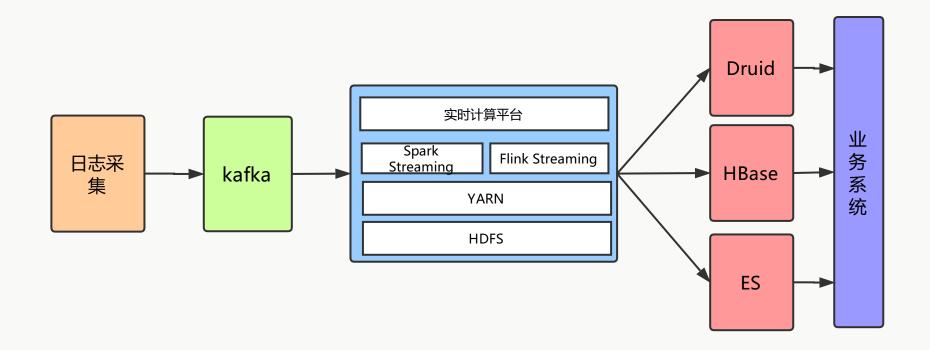
## 数据处理流程

实时计算和实时数据处理的数据流程包括:数据采集,数据清洗,数据计算,数据应用。数据应用通常会让数据有如下几个应用出口:Sink到持久化存储系统以便后续进行离线分析,实时监控大盘,实时API调用。



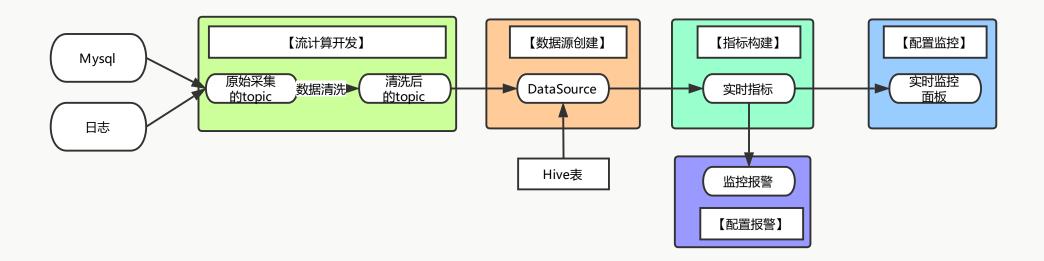
# 2 实时BI监控架构

根据实时计算和实时数据处理的数据流程,构建了支撑实时BI监控系统整体方案。



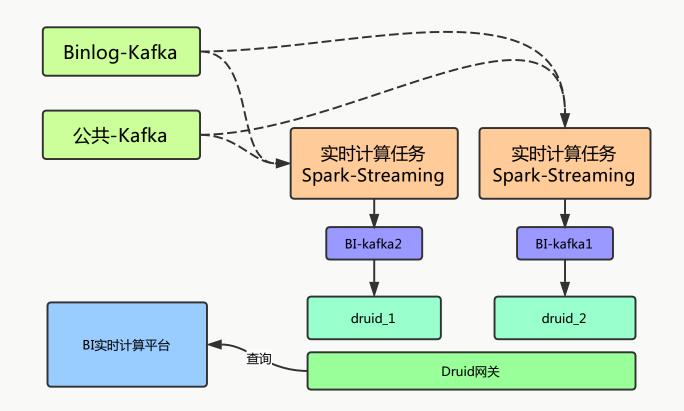
# 监控与告警

数据经过采集和计算后接入Druid, 通过创建DataSource以及配置指标进行监控和告警。



# 实时双链路建设

- 1. 构建链路双活体系,提升BI监控稳定性。
- 2. 采用BI专用Kafka集群,提升集群稳定性。
- 3. 统一流计算引擎。



# 3 Spark Streaming实践

- 01 Spark Streaming在滴滴的应用
- 02 Spark Streaming与Kafka集成
- 03 Spark Streaming监控与告警
- 04 Spark Streaming最佳实践

# 01 Spark Streaming在滴滴的应用

#### 主要场景

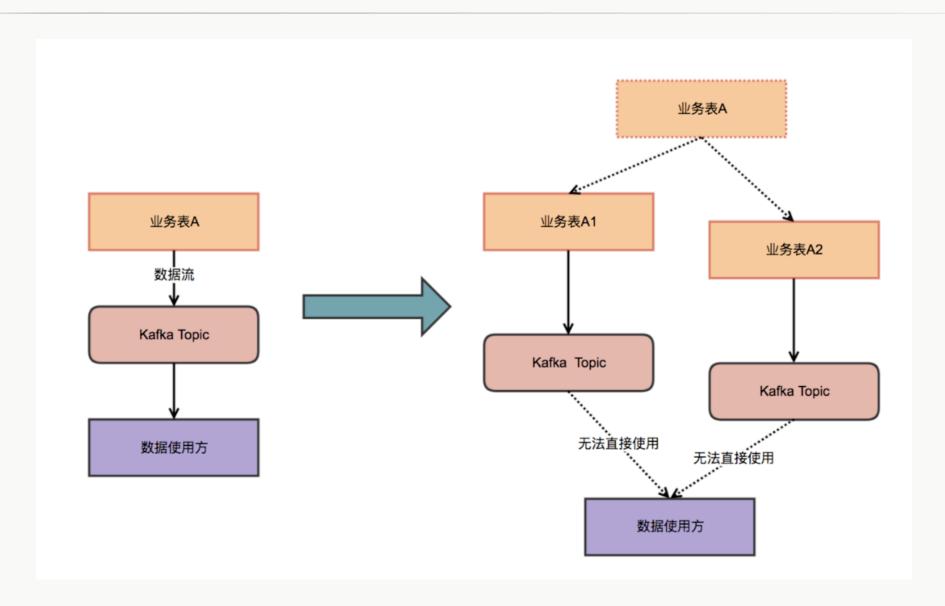
- 1)海量异构数据处理,包括日志、binlog、事件消息等数据。
- 2) 低延迟、高吞吐。
- 3)监控系统,数据质量系统等实时系统



# Spark Streaming在滴滴的业务规模

- 1. 集群规模:约500台机器(Spark & Flink)
- 2. 400+Spark Streaming任务
- 3. 千万/s 以上的流量, 承担核心BI监控的数据处理

# Spark Streaming - BI监控中的多流Join问题



# Spark Streaming - BI监控中的多流Join实践

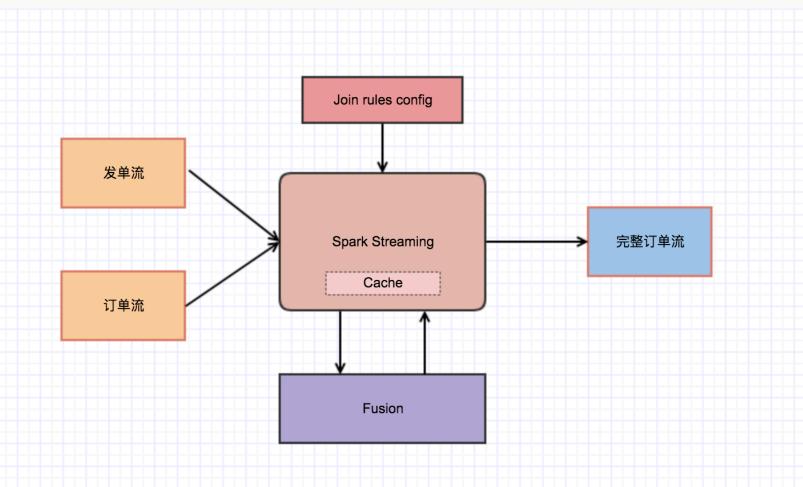
多个不同流根据一定规则join的问题(例如:订单发单流与接单流join问题)

#### 特点:

- 不同流需要join的数据时间跨度较长(例如:发单与接单时间跨度最长一周之久)
- 数据源格式不定 (例如:binlog数据和业务日志的join)
- · join规则多样化
- 系统要求吞吐量大(订单表流量是5M/s)、延迟低(秒级)

# Spark Streaming-BI监控中的多流Join方案

#### 我们实现的多流Join的方案如下:



收益:保证了BI监控的核心链路高可用,为业务方提供了快速的流join方案

# 02 Spark Streaming与Kafka集成

这里不再讨论旧的基于Receiver的方式。

Kafka的offset保存,这里有几种方式:

- 1)开启checkpoint,保存在分布式文件存储中,缺点:代码发生变化后,不能很好的从checkpoint恢复,甚至会丢失数据。
- 2)提交到Kafka中。(我们修改源码,实现了该方式)
- 3)保存到自定义的存储中。

收益:通过实现对Offset的存储,针对流计算任务的重新打包、运行或重启时,保证了数据不会丢失,随时追回延迟数据,保障了BI监控数据的安全和准确性。

# 03 Spark Streaming监控与告警

#### 1. Spark WebUI 增加OffsetLag

Input	Metadata
Kafka direct stream [0]	topic: m3_niepan_driver_listen_etl partition: 3 offsets: 1960153472 to 1960170262 lag:74854 topic: m3_niepan_driver_listen_etl partition: 1 offsets: 1957408236 to 1957412486 lag:0 topic: m3_niepan_driver_listen_etl partition: 4 offsets: 1950367640 to 1950384430 lag:3337 topic: m3_dos_order_merge partition: 4 offsets: 1150807015 to 1150810893 lag:0 topic: m3_dos_order_merge partition: 3 offsets: 1150731764 to 1150735620 lag:0 topic: m3_niepan_driver_listen_etl partition: 0 offsets: 1967884998 to 1967901788 lag:3417 topic: m3_dos_order_merge partition: 5 offsets: 1150511139 to 1150514812 lag:0 topic: m3_niepan_driver_listen_etl partition: 2 offsets: 1962176457 to 1962176857 lag:0 topic: m3_dos_order_merge partition: 2 offsets: 1150810947 to 1150814885 lag:0 topic: m3_dos_order_merge partition: 8 offsets: 1150678033 to 1150682002 lag:0 topic: m3_dos_order_merge partition: 8 offsets: 1151037323 to 1151041002 lag:0 topic: m3_dos_order_merge partition: 0 offsets: 1151037323 to 1151041002 lag:0 topic: m3_dos_order_merge partition: 9 offsets: 1151927592 to 1151931299 lag:0 topic: m3_dos_order_merge partition: 7 offsets: 1151484450 to 1151488105 lag:0

# 03 Spark Streaming监控与告警

#### 2. Spark框架改造增加自定义metrics并接入Druid,作为监控指标



# 03 Spark Streaming监控与告警

延迟指标:OffsetLag、Batch堆积

告警方式:钉钉、邮件、电话、短信

收益:通过提供各个维度的告警服务(默认提供任务异常告警和任务延迟告警),可以做到对BI监控的各个任务及时响应、处理,为运营和决策人员提供可靠、可用、及时的数据,支撑整个BI监控系统。

# 04 Spark Streaming最佳实践

#### 解决大部分问题

- 1、开启反压: spark.streaming.backpressure.enabled
- 2、消费速度: spark.streaming.kafka.maxRatePerPartition
- 3、Kafka抖动导致 No leader found: spark.streaming.kafka.maxRetries
- 4、driver端oom: spark.driver.memory
- 5、包冲突: spark.driver/executor.userClassPathFirst
- 6. Spark-Streaming-Kafka: --executor-cores \* --num-executors <=topicPartitons
- 7、job级别的并发: spark.streaming.concurrentJobs

# 4 Druid实践

- 01 Druid特性
- 02 Druid在BI监控的应用
- 03 Druid在BI监控的建设

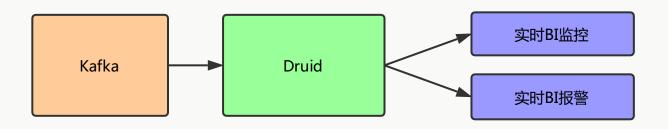
### 01 Druid特性

#### 为什么选择Druid呢???

- 对于大部分查询场景可以亚秒级响应。
- 事件流实时写入与批量数据导入兼备。
- 数据写入前预聚合节省存储空间,提升查询效率。
- 水平扩容能力强
- 社区活跃

# 02 Druid在BI监控的应用

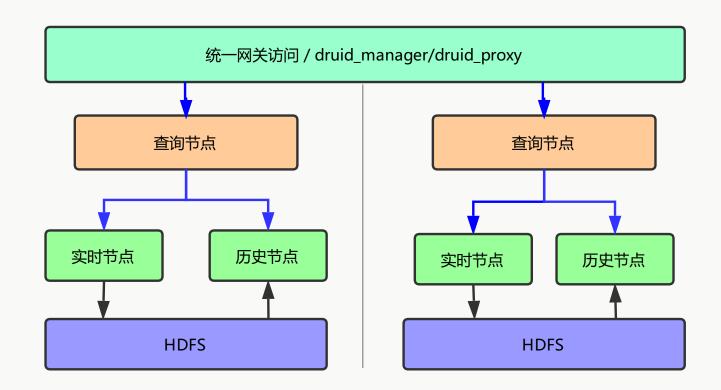
Druid能够对数据进行实时聚合计算,支持TopN, GroupBy, Filter, Count等即时查询,也因此非常适合BI实时监控的业务需求。



### 03 Druid在BI监控的建设

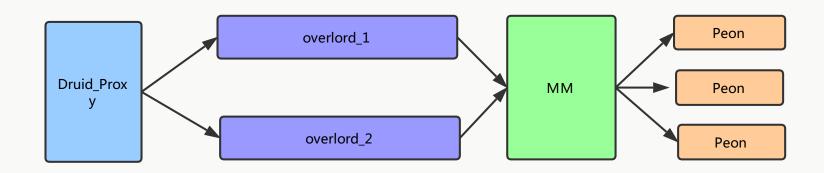
构建了Druid异地双活体系,多集群对业务透明,并且支撑了核心业务监控与报警。

- 1. 建设druid\_manager以满足管理员运维druid集群。
- 2. 建设druid\_proxy以满足用户管理自己的数据源。
- 3. 建设druid网关统一查询入口。



## 03 Druid在BI监控的建设

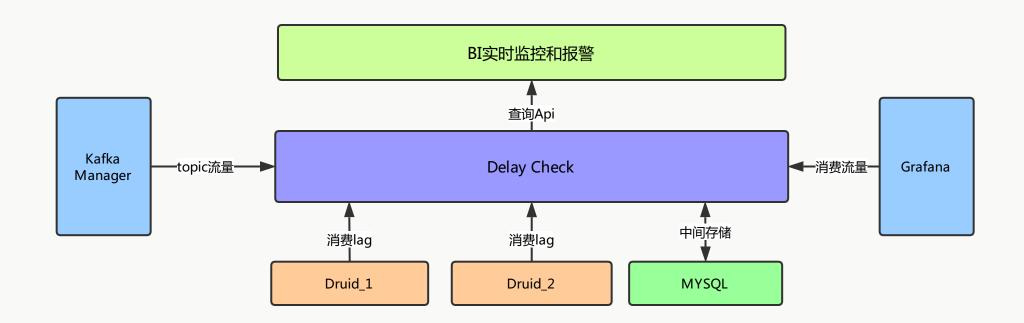
随着业务的发展每个druid集群的数据源个达到360个左右,任务数量达到1000左右。对于一个单节点的实时节点来说压力非常大,常常会产生任务调度过来从而产生数据延迟。如何解决呢?



# 03 Druid在BI监控的建设

#### druid消费是否有延时?数据本身有延时呢?

- 1. Druid的延时API,辅助BI报警。
- 2. Druid支持多粒度查询,提升BI监控报警的精确度。



# 实时BI监控一未来规划

- Druid-on-YARN, 提升集群的资源利用率。
- 资源隔离,保证集群的稳定性。
- Streaming SQL开发流计算像写Hive SQL/Spark SQL一样简单。
- 流计算任务血缘体系构建。
- 实时计算诊断系统。

