



Flink China

基于Flink的实时特征平台

姓名:刘康

公司:携程

职位:技术中心大数据资深研发工程师



平台开发背景 及平台现状



原实时特征作业的开发运维

计算平台：依据项目的性能指标（latency，throughput等），在已有的实时计算平台:Storm或Spark中选择

开发运维流程

1. 80%以上的作业需要用到消息队列数据源，但是消息是非结构化数据且没有统一的数据字典，需要通过消费对应的topic，解析消息和确定所需的内容
2. 基于需求中的场景，设计开发计算逻辑
3. 在实时数据不能完全满足数据需求的情况，开发单独的离线作业以及融合逻辑
4. 设计开发数据的校验和纠错逻辑
5. 测试上线&监控和预警

平均每个作业耗时：3天-5天



痛点

- 消息队列数据源结构没有统一的数据字典
- 特征计算逻辑高度定制化，开发测试周期长
- 实时数据不能满足需求时，需要定制离线作业和融合逻辑
- 校验和纠错方案没有形成最佳实践，实际效果比较依赖个人能力
- 监控和预警方案需要基于业务逻辑定制

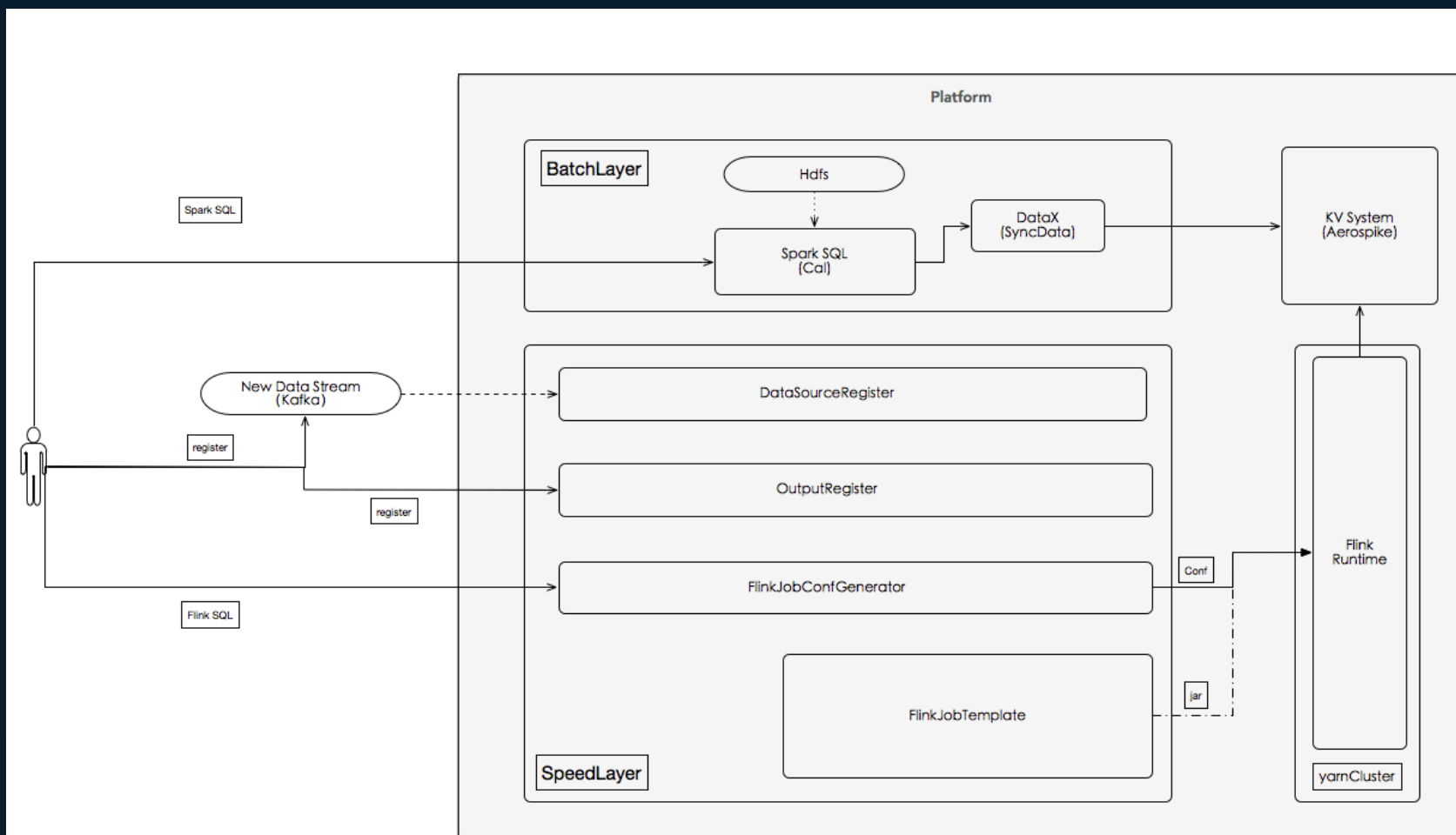


平台的目标

- 实时数据字典：提供统一的数据源注册、管理功能，支持单一结构消息的 topic 和包含多种不同结构消息的 topic
- 逻辑抽象：抽象为 SQL，减少工作量 & 降低使用门槛
- 特征融合：提供融合特征的功能
- 数据校验和纠错：提供离线数据校验和纠错实时特征的功能
- 实时计算延迟：ms 级
- 实时计算容错：端到端 exactly-once
- 统一的监控预警和 HA 方案



特征平台系统架构





平台功能展示-数据源注册

注册数据源信息

数据源名称*

数据源名称

数据源名称必填

Topic类型* ☐ HermesKafka ☒ Kafka

Topic编码* ☐ Avro ☒ Json

上传JsonSchema文件

选择文件 | vac_test.json

```
ts:long
custom_key:string
pkgid:int
slcityid:int
dtcityname:string
bu:string
slcityname:string
dtcityid:int
```

Topic名称*

Topic名称

Topic名称必填

Schema类型* ☐ multiply ☒ single

Topic的Table别名(用于SQL)*

Topic的Table别名(用于SQL)

Table别名必填

列投影设置

显示 10 行

搜索:

| Field名称 | Tab列名 | 类型 | 自定义类型类型 | 默认EventTime属性 |
|------------|------------|--------|---------|-----------------------|
| bu | bu | STRING | | <input type="radio"/> |
| custom_key | custom_key | STRING | | <input type="radio"/> |
| dtcityid | dtcityid | INT | | <input type="radio"/> |
| dtcityname | dtcityname | STRING | | <input type="radio"/> |
| pkgid | pkgid | INT | | <input type="radio"/> |
| slcityid | slcityid | INT | | <input type="radio"/> |
| slcityname | slcityname | STRING | | <input type="radio"/> |
| ts | ts | LONG | | <input type="radio"/> |

显示 1 to 8 of 8 行

上一页 1 下一页

数据源类型* ☒ 注册 ☐ 转换

创建

取消



实时特征编辑-基本信息

特征编辑与测试

首页 / 特征管理 / 特征编辑与测试

离线特征

在线特征

1 基本信息

2 数据源信息

3 计算信息

4 存储信息

Tips: 首次创建特征请先在配置管理中配置数据源,分类与特征输出等配置

实时特征名称*

实时特征名称

实时特征名称必填

实时特征别名*

实时特征别名

Flink使用AS作为数据源时别名将被作为字段名使用,需同一张AS表中唯

是否公开*

公开

版本信息*

1

所属分类

创建

取消



实时特征编辑-数据源选择

特征编辑与测试

首页 / 特征管理 / 特征编辑与测试

离线特征

在线特征

1 基本信息

2 数据源信息

3 计算信息

4 存储信息

KAFKA 数据源位置信息

ds_0



选择数据源:*

test

AeroSpike 数据源 (可选)

已经存储到As的表

default_online_features.ops_user_info

创建

取消



实时特征编辑-SQL计算

特征编辑与测试

首页 / 特征管理 / 特征编辑与测试

离线特征

在线特征

1 基本信息

2 数据源信息

3 计算信息

4 存储信息

计算引擎:

FLINK

实时计算描述:

计算描述

实时计算sql: Tip

```
1 select ID, cityId,cityName from test
2 LEFT JOIN LATERAL TABLE(asGetBins(PRODUCTID)) as T(cityId,cityName) ON TRUE
```

| # | 列名 | 类型 | 映射主键 |
|---|----------|-----------|--------|
| 0 | ID | Key Value | userid |
| 1 | cityId | Key Value | city |
| 2 | cityName | Key Value | |

是否进行Checkpoint:



checkPoint触发间隔(ms)

100

checkPoint间最小间隔(ms)

50



实时特征编辑-选择输出

特征编辑与测试

首页 / 特征管理 / 特征编辑与测试

离线特征

在线特征

1 基本信息

2 数据源信息

3 计算信息

4 存储信息

输出到AeroSpike(必填)

输出到Kafka(可选)

🔄 获取注册的存储地址

存储到As的表*

default_online_features.ops_forTest

存储到As的TTL(time to live: 单位-秒) ⓘ *

-1

🚀 创建

取消



实时特征编辑-选择可选的输出

特征编辑与测试

首页 / 特征管理 / 特征编辑与测试

离线特征

在线特征

1 基本信息

2 数据源信息

3 计算信息

4 存储信息

输出到AeroSpike(必填)

输出到Kafka(可选)

获取注册的Kafka Topic

用于push变化key的Topic*

testTopic

创建

取消



选择Flink的原因



| | storm V1.2.2 | spark Structured Streaming v2.3.1 | flink v1.5 |
|--------------------------------------|--------------------|---|-------------------------------------|
| 延迟 | Streaming ms级 | MicroBatch 100ms级 Streaming ms级 (实验) | Streaming ms级 |
| 容错 | Ack atLeastOnce | CheckPoint&WAL exactlyOnce | CheckPoint&SavePoint exactlyOnce |
| SQL成熟度 (Unsupported Functions) | aggregation、join | distinct、limit、order by(partial) | distinct aggregate |



Flink实践



使用用例

SessionWin



会话窗口用途广泛，可以用于推荐召回、用户召回等；原设计方案需要使用分布式锁做进程间的并发控制，复杂度高

```
select user,count(ts) from vac_ts group by SESSION(userEventTime_ts,
INTERVAL '10' SECOND),user
```

External CheckPoint



实时计算job执行异常设置的重启策略无法恢复需要修复bug时，如果设置了externalCheckPoint，很大几率保证exactly Once的语义



兼容开发

AS Async
Upsert
TableSink



基于flink Asynchronous I/O开发 (支持exactlyOnce语义)

异步写入AS

写入AS成功后可选择push key至Kafka

支持Upsert Stream的TableSink

UDF-AS
Table
Functions



通过定义Table Function类型的UDF支持, SQL中用到AS数据的情况

示例:

```
SELECT ID, cityId, cityName FROM product LEFT JOIN LATERAL  
TABLE(asGetBins(productId)) as T(cityId,cityName) ON TRUE
```

AS=Aerospike



碰到的坑

SQL_TIMESTAMP



内部默认反序列化JSON中SQL_TIMESTAMP类型时，值必须是格式为yyyy-MM-dd' T'HH:mm:ss.SSS'Z'的字符串

场景：直接使用StreamTableEnvironment的方法registerTableSource注册表时，表中的时间属性（eventTime）字段的类型需要配置为SQL_TIMESTAMP

SlidingWin
Offset



窗口的offset需要小于winSize，不能用于时区适配

下面是当前计算win的start值的公式

$$\text{timestamp} - (\text{timestamp} - \text{offset} + \text{windowSize}) \% \text{windowSize}$$



平台当前效果 &未来规划

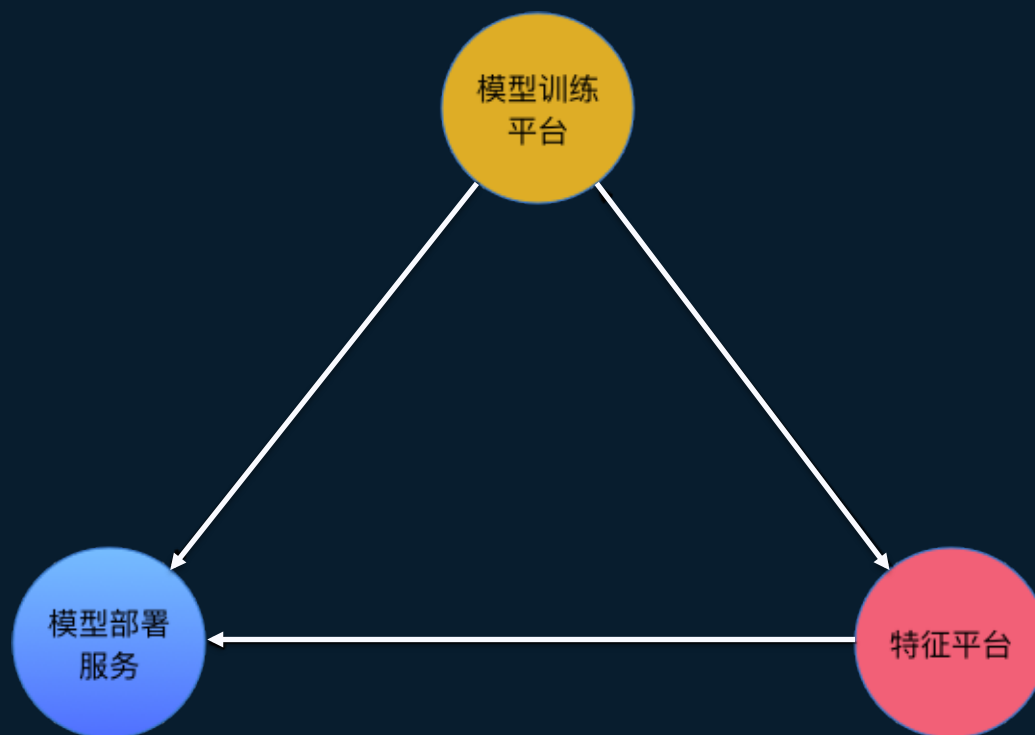


特征平台当前效果&规划

- 当前效果：将实时特征上线周期从原平均3天-5天降至小时级
- 未来规划：
 1. 完善特征平台的功能：融合特征等
 2. 简化步骤，提高用户体验
 3. 根据需求，进一步完善SQL的功能例如支持win的开始时间offset，可以通过countTrigger的win等



模型生态的进一步规划





Flink China

Thanks !