



Apache Paimon



StarRocks

喜马拉雅基于Paimon+StarRocks 构建直播实时湖仓



喜马拉雅

王琛 | 数仓专家

Streaming Lakehouse Meetup

CONTENT

目录 >>

01 /

喜马拉雅直播业务简介

02 /

直播数据仓库架构迭代和升级

03 /

实时湖仓的效果和收益

04 /

未来的展望和规划

01.

直播业务介绍与实时湖仓背景



直播业务



音频直播



视频直播



多人厅

直播数据大图



实时背景

通过实时数据监控系统，
可以及时发现问题并迅
速采取行动

**实时监控
与预警**

在市场环境快速变化下，
通过分析实时数据，迅
速调整玩法策略

**快速调
整策略**

**提高业务
决策速度
和质量**

**提升用户
体验和满
意度**

**个性化
推荐**

根据用户玩法行为和偏好，数
据驱动提供个性化的内容推荐，
增加用户黏性和满意度。

**实时反馈
与互动**

即时获取用户反馈，并根据这
些反馈调整直播玩法策略

**增强
安全性**

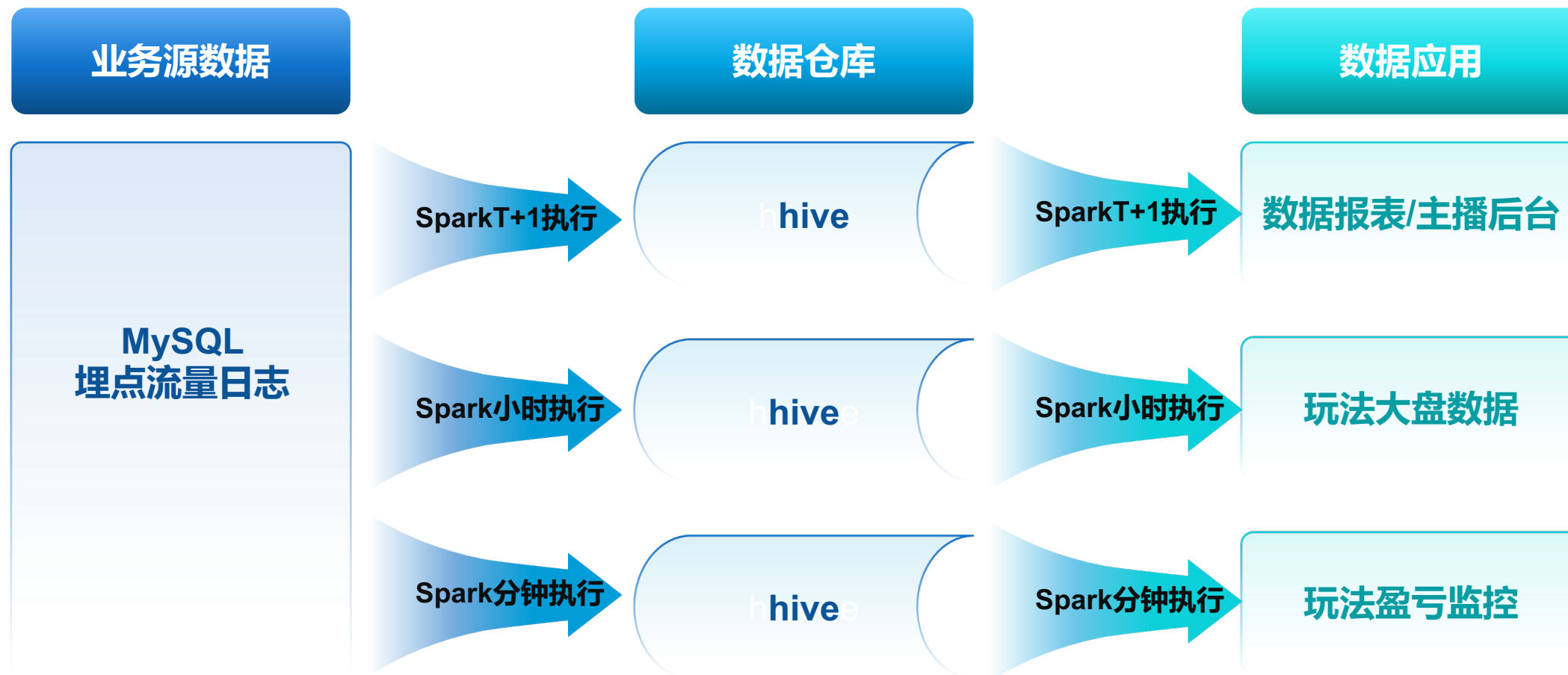
可以实时监控异常行为，提高
系统的安全性

02.

直播数仓架构迭代和升级



批处理（spark任务定时执行）



流处理 (flink+kafka)



架构存在不足

1

及时性差

数据处理延迟，无法满足实时性需求

1、直播收入数据涉及到交易订单数据和用户虚拟币等多个账户交互，来实时跟踪用户交易到公司收入整条链路，其中涉及到多个业务流进行join，与离线数据对不上，排查困难，并且kafka保留周期短，回溯也困难，导致出现问题debug周期特别长

2

开发维护成本高

复杂的架构导致开发和运维成本较高

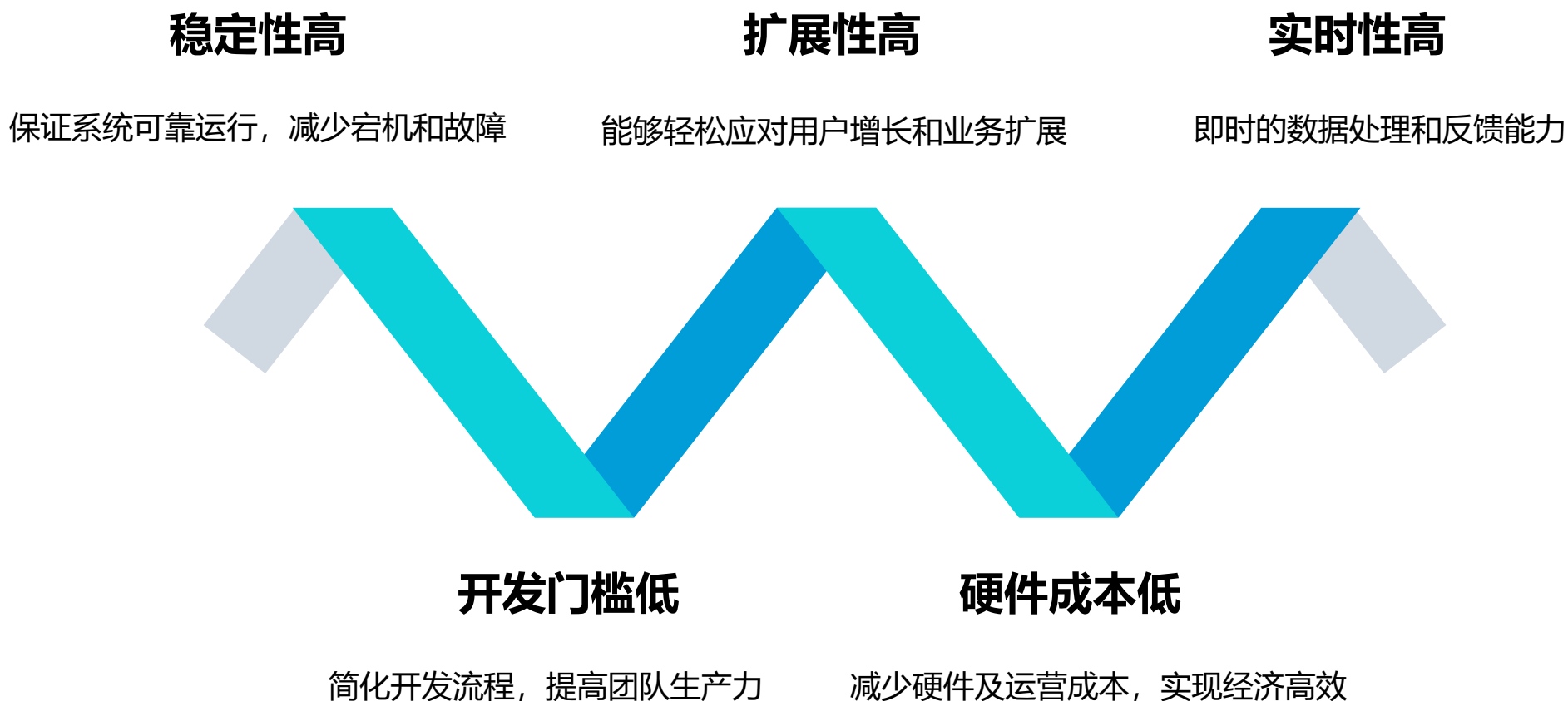
3

费用成本高

需要一定的硬件费用成本

2、多流join需要一直持续保留各流的状态，经常会出现大状态的情况，出现问题需要对任务进行重启的时候，经常重启等待时间长，甚至重启不断失败的情况，当正处于活动期间，任务重启失败会导致业务无法实时感知数据情况

技术选型考虑因素（三高两低）



数据湖选型对比

指标	Delta Lake	Hudi	Paimon
Flink集成	要求Hadoop 3.3.2和Flink 1.16; 不支持Flink CDC (Flink无法更新DeltaLake)	支持	支持
运维复杂度	低	高	低
生态系统	有丰富的文档和社区支持	有丰富的文档和社区支持	相对于Hudi和Delta Lake, Paimon是新兴项目, 生态系统和社区支持可能不如前两者成熟
查询性能	好	大数据量下入库和查询耗费资源很严重	好

OLAP选型对比

	高频更新、删除场景	事务	物化视图	多表关联性能	高并发
ClickHouse	缺少高频率、低延迟的修改或删除已存在数据的能力 只支持更新，不支持删除	不完整	不具备自动化完成视图的更新功能	弱（几乎不支持），会拉高CPU（通过大宽表解决）	低，官方建议不超过100，会拉高CPU
StarRocks	Primary Key 模型（牺牲微小的写入性能和内存占用提升了查询的性能），支持百万级别的TPS	支持事务性的DDL与DML操作（exactly one）	基表变动时，物化视图自动更新维护	支持 broadcast join、Shuffle Join、Colocate Join 等多种分布式 Join 方式，多表关联性能更优（星形、雪花模型）	支持多并发查询，部分场景可以达到 1 万以上 QPS（CPU比较高）

StarRocks模型

明细模型

特点：用于保存和分析原始明细数据，以追加写为主要写入当时，数据写入后几乎无更新

使用场景：日志，操作记录，时序类数据等

聚合模型

特点：用于保存和分析汇总类数据，无需查询明细数据。数据导入后实时完成聚合，数据写入后几乎无更新

使用场景：按时间、地域等维度汇总数据等

主键模型

特点：支持基于主键的更新，大批量导入时保证高性能查询，用于保存和分析需要更新的数据

使用场景：状态会发生变动的订单数据等

更新模型

特点：支持基于主键的更新，更新频率比主键模型更高，用于保存和分析需要更新的数据

使用场景：状态会发生变动的订单数据等

技术选型

	传统离线Hive	Flink+kafka	Flink+Paimon +StarRocks	Flink+StarRocks (OLAP)	Flink+数据湖
指标					
实时性	×	√	√	√	×
扩展性	高	低	高	一般	一般
架构维护	简单	复杂	简单	简单	复杂
开发门槛	低	高	一般	低	一般
费用成本	低	高	一般	高	一般

实时湖仓架构



Flink CDC 实时入仓

1. 替换所有自研Binlog链
2. 平台任务语法层面优化
3. 全量与增量无缝衔接
4. 增量部分自动缩容

```

1 source mysql7_fans_source = 'mysql-cdc-2.0' (
2   id BIGINT
3   , binlog_ts TIMESTAMP_LTZ(3) source_operation_timestamp
4   , props VARCHAR source_complete_data
5   , source_database_name VARCHAR source_database_name
6   , source_table_name VARCHAR source_table_name
7   , PRIMARY KEY(id)
8 )
9
10 with (
11   'hostname' = ''
12   , 'username' = ''
13   , 'password' = ''
14   , 'port' = ''
15   , 'scan.startup.mode' = '' -- initial
16   , 'database-name' = ''
17   , 'server-id' = ''
18   , 'server-time-zone' = ''
19   , 'scan.snapshot.fetch.size' = ''
20   , 'table-name' = '(tb_name_a{[0-9]?[0-9]}$|tb_name_b{[0-9]?[0-9]}$|tb_name_c{[0-9]?[0-9]}$)'
21 );
22
23 sink tb_clearing_acc = kafka 'kafka/cluster/topic' (
24   id BIGINT
25   , binlog_ts BIGINT
26   , props VARCHAR
27   , source_database_name VARCHAR
28   , source_table_name VARCHAR
29   , dt VARCHAR
30 )
31 WITH(
32   'format' = 'debezium-json'
33 )
34 select
35   id
36   , millis2long(binlog_ts) as binlog_ts
37   , props
38   , source_database_name
39   , source_table_name
40   , if(millis2long(binlog_ts) = 0, '20230607', from_unixtime(millis2long(binlog_ts)/1000, 'yyyyMMdd')) as dt
41 from mysql7_fans_source
42 where source_table_name like 'tb_name a%';
43
44 mysql7_fans_source
45
46 '1,mysql7_fans_source'
47 '1,mysql7_fans_source'
48 '1,mysql7_fans_source'
49 '1,mysql7_fans_source'
50 '1,mysql7_fans_source'
51 '1,mysql7_fans_source'
52 '1,mysql7_fans_source'
53 '1,mysql7_fans_source'
54 '1,mysql7_fans_source'
55 '1,mysql7_fans_source'
56 '1,mysql7_fans_source'
57 '1,mysql7_fans_source'
58 '1,mysql7_fans_source'
59 '1,mysql7_fans_source'
60 '1,mysql7_fans_source'
61 '1,mysql7_fans_source'
62 '1,mysql7_fans_source'
63 '1,mysql7_fans_source'
64 '1,mysql7_fans_source'
65 '1,mysql7_fans_source'
66 '1,mysql7_fans_source'
67 '1,mysql7_fans_source'
68 '1,mysql7_fans_source'
69 '1,mysql7_fans_source'
70 '1,mysql7_fans_source'
71 '1,mysql7_fans_source'
72 '1,mysql7_fans_source'
73 '1,mysql7_fans_source'
74 '1,mysql7_fans_source'
75 '1,mysql7_fans_source'
76 '1,mysql7_fans_source'
77 '1,mysql7_fans_source'
78 '1,mysql7_fans_source'
79 '1,mysql7_fans_source'
80 '1,mysql7_fans_source'
81 '1,mysql7_fans_source'
82 '1,mysql7_fans_source'
83 '1,mysql7_fans_source'
84 '1,mysql7_fans_source'
85 '1,mysql7_fans_source'
86 '1,mysql7_fans_source'
87 '1,mysql7_fans_source'
88 '1,mysql7_fans_source'
89 '1,mysql7_fans_source'
90 '1,mysql7_fans_source'
91 '1,mysql7_fans_source'
92 '1,mysql7_fans_source'
93 '1,mysql7_fans_source'
94 '1,mysql7_fans_source'
95 '1,mysql7_fans_source'
96 '1,mysql7_fans_source'
97 '1,mysql7_fans_source'
98 '1,mysql7_fans_source'
99 '1,mysql7_fans_source'
100 '1,mysql7_fans_source'

```

Flink CDC收益

1. 更简单更稳定的架构，保证了数据集成的稳定性，同时也提供了更好的运维性
2. 基于Checkpoint提供了精确一次语义保证
3. 支持横向扩容以提升数据同步能力
4. 支持Schema变更情况下字段透传到下游
5. 数据同步隔离性好，不同任务之间独立运行互不影响



自研数据集成工具

上一步

1 配置数据来源去向

2 配置字段映射

下一步

数据来源配置

数据去向配置

基础配置

* 数据类型: MySQL

* 实例: 请选择

* 数据库: 请选择

* 表: 请选择

* 主键字段: 请选择

元数据字段: 请选择

* 同步方式: 全增量一体同步

* 数据类型: Kafka

* 实例: Kafka

* 主题: StarRocks

* 数据格式: debezium-json

高级配置

集成

代码

配置

状态图

报警

```
1 include '/common/functions.sql';
2 include '/common/paimon_catalog.sql';
3 set 'flink.execution.checkpointing.mode' = 'EXACTLY_ONCE';
4 set 'taskmanager.memory.managed.fraction' = '0.1';
5
6
7 SOURCE source_table = `mysql-cdc-2.0` (
8   `id` BIGINT,
9   `user_id` BIGINT,
10  `account_type` INTEGER,
11  `business_type` INTEGER,
12  `amount` DECIMAL(20,6),
13  `order_no` STRING,
14  `order_title` STRING,
15  `context` STRING,
16  `create_time` TIMESTAMP,
17  `last_update_time` TIMESTAMP,
18  `version` BIGINT,
19  `database_name` STRING source_database_name,
20  `table_name` STRING source_table_name,
21  PRIMARY KEY(`id`)
22 ) WITH (
23
24
25
26
27
28
29
30
31
32 );
33
34 SINK paimon_catalog.ods_dl.s_du_live_treasure_coin_order_delta = paimon (
35   `id` BIGINT,
36   `user_id` BIGINT,
37   `order_no` STRING,
38   `account_type` INTEGER,
39   `business_type` INTEGER,
40   `amount` DOUBLE,
41   `order_title` STRING,
42   `version` BIGINT,
43   `context` STRING,
44   `create_timestamp` TIMESTAMP,
```

Paimon应用



Paimon流读



01



永久保存日志



02



基于主键的
部分更新

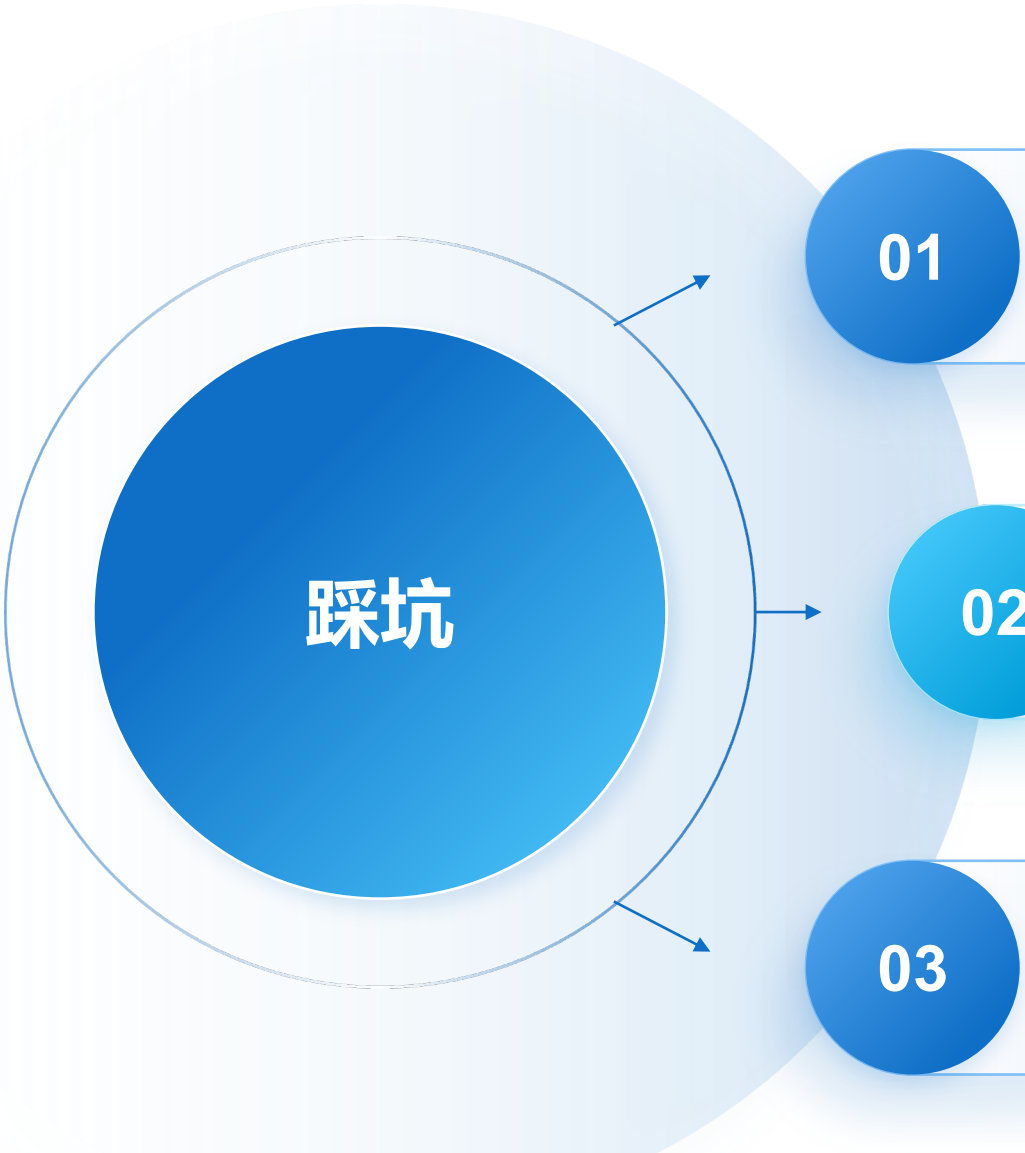


03

目前Paimon完成接入直播数据表**50+**张表

遇到的问题

踩坑



The diagram features a central blue circle with the text '踩坑' (Pitfall) in white. This circle is surrounded by a larger, lighter blue circle. Three arrows originate from the right side of the central circle and point towards three separate blue rounded rectangular boxes arranged vertically on the right. Each box contains a number in a small blue circle followed by a title and a description of a problem.

01

Paimon 流流join 加载慢

每次重启任务发现都需要很长时间才能更新到当前时间的数据，经排查发现每次重启任务都从历史分区开始重新刷数据。

02

join维度表丢数据

流表join维度表一开始运行稳定，过几天开始出现丢数据的情况。经排查是因为维度表没有持久化存储下来，历史维度数据已经过期导致。

03

5张业务表进行union 丢数据

对各类型的用户打赏5张表进行union整合成一张汇总的打赏表,社区的同学确认问题，发现有类似的Bug并且已经修复：

03.

实时湖仓的效果和收益



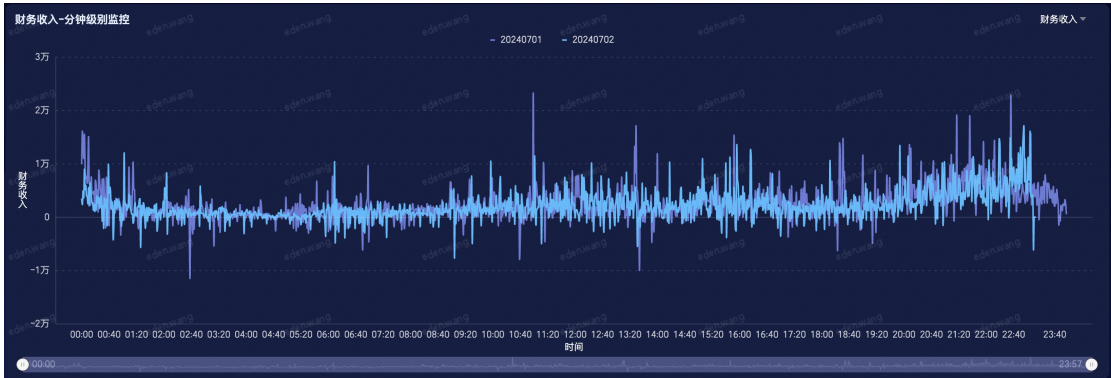
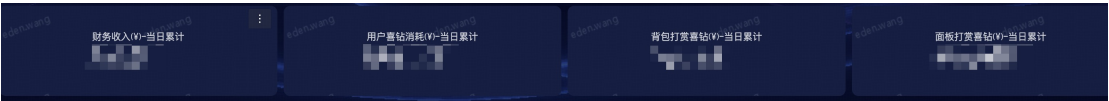
业务收益



业务收益

01

实时数据感知，提高业务效率和决策质量



业务收益

02

实时主播流水榜和用户消耗榜

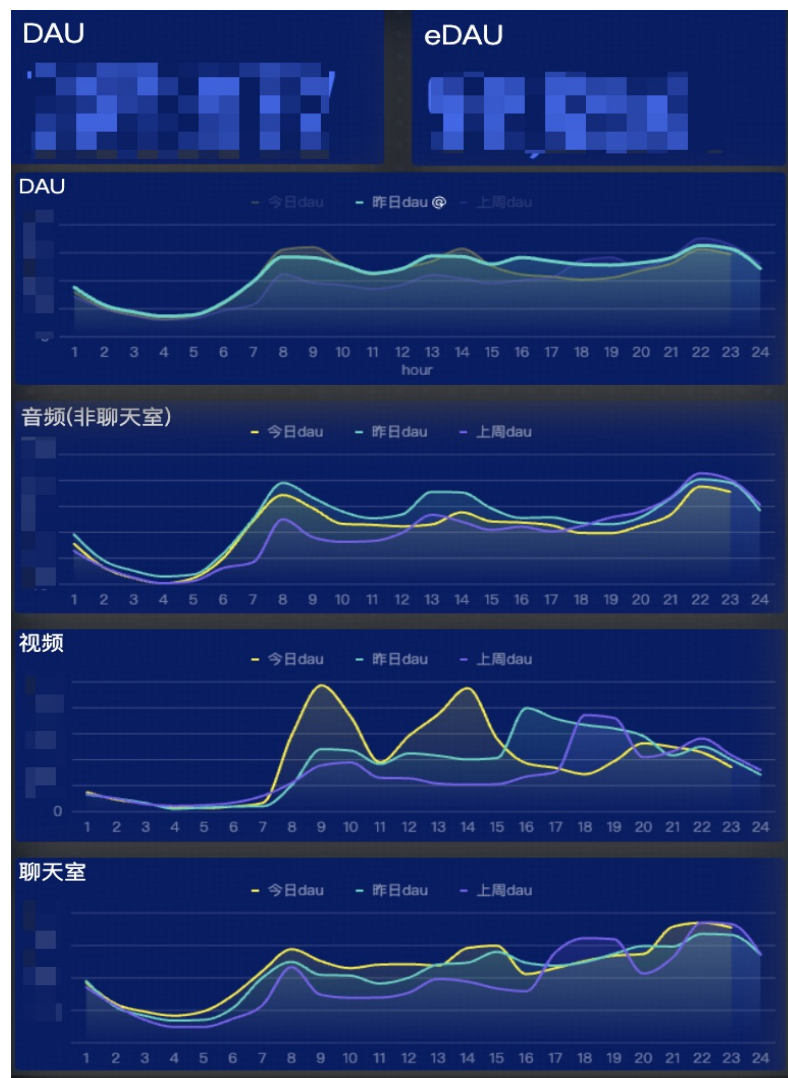
当日用户消耗数据				已选字段(4)
用户ID	用户昵称	财务消耗 (元)	喜钻消耗 (元)	
5114515	倔			
3324577				
3121866	312186662			
4963570	兒			
4813637	住在2月13			
4965218	糖			
4429802	等 等 等 等 一 下			
5104469	无年少			
5092434	509243482			
3374511				
1779114	查无此人			
4045455	客a			
3056860	夜			
5097938	青夕			
5102554	510255452			
4706300	个雪梨			
5120943	512094362			
共 39316 条				20 条/页

当日主播流水					已选字段(5)
主播ID	主播昵称	公会ID	公会昵称	主播流水	
144164	月之下_哏头Bng				
5875413	安_常乐久安				
283753	小白条				
1825597	养快线好喝				
431867	控的狗刺				
5678925	_呦呦				
5050946	州小臭宝儿				
3955471	丶_沐音				
1590789	雪儿i				
3242889	马富婆				
1715916	慕言				
3920663	年_满天星				
1705837	风_屿你				
559190	/追马				
0899031	糖				
5300984	3_声笙				
2728663	公子n				
共 8192 条					20 条/页

业务收益

03

实时DAU和eDAU情况



业务收益

04

盈亏预警和快速响应，避免出现未知损失

盈亏监控

各玩法收支情况

玩法名称	参与用户数	财务收入喜钻	用户喜钻消耗	进虚拟币	出虚拟币	进背包	出背包	人民币直接消耗
背包送礼								
面板送礼								
心愿商店								
心愿商店-派对								
个播-黄金矿工								
守护								
聊天室-黄金矿工								
幸运转转								
星际探险-聊天室								
夺宝								
个播-炼金工坊								
星际探险-个播								
个播-地心宝藏								
奥林匹斯-个播								
派对-地心宝藏								
聊天室-炼金工坊								
派对心动许愿池								
福袋								
奥林匹斯聊天室								
波波球								



04.

未来的展望和规划



未来展望



01

进一步拓展实时湖仓的应用场景和规模，更多的业务接入实时湖仓



03

借助实时湖仓能力，数据助力AI

02 对Paimon和StarRocks特性进行进一步探索，助力业务快速增长



Apache Paimon



StarRocks

Thanks

Streaming Lakehouse Meetup