

StarRocks在游族的多维 分析场景

刘成彬 资深大数据开发



目录 CONTENT

01 历史背景

03 应用场景

02 StarRocks优势

04 未来规划

01

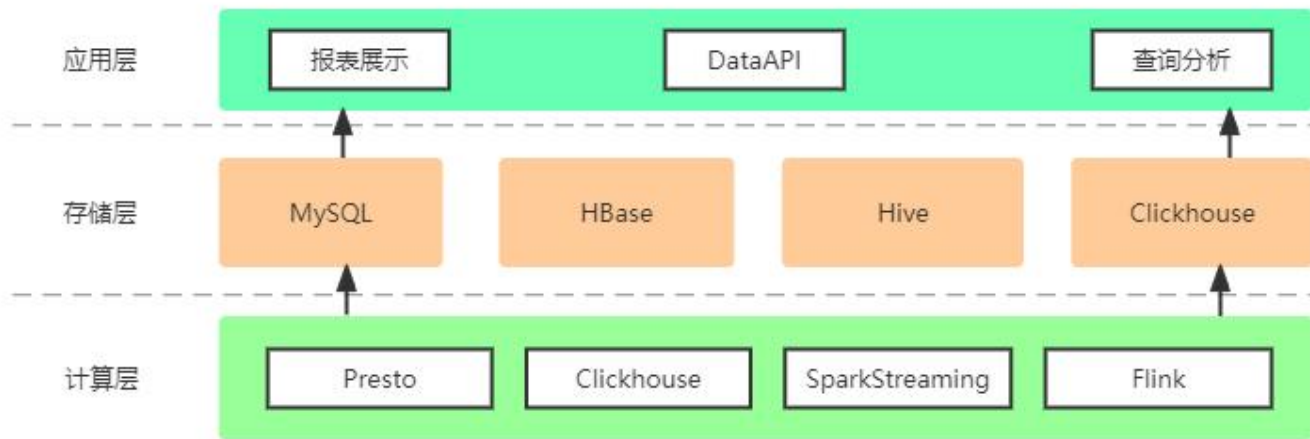
历史背景



历史背景

痛点

1. 维护多套组件，运维成本高
2. 各组件SQL语法存在差异，开发维护任务成本高
3. 同指标数据下，需要保证不同组件计算的结果与口径都一致的成本比较高
4. 结果数据比较大时，MySQL性能比较差



诉求

为了解决我们的痛点，并且能够适应公司未来的实时查询需求，我们需要统一OLAP引擎，选择引擎至少需要满足以下要求：

- 数据秒级写入，低延迟毫秒级响应
- 复杂场景多表关联查询性能好
- 运维简单，方便扩展
- 支持高并发
- 易用性强，开发简单方便

02

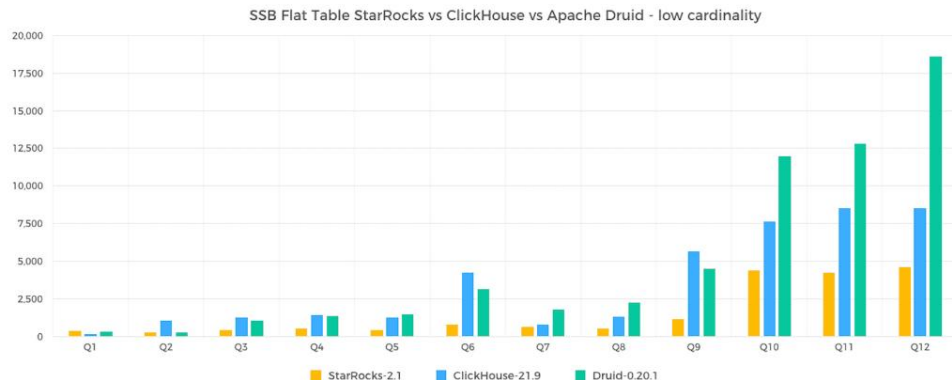
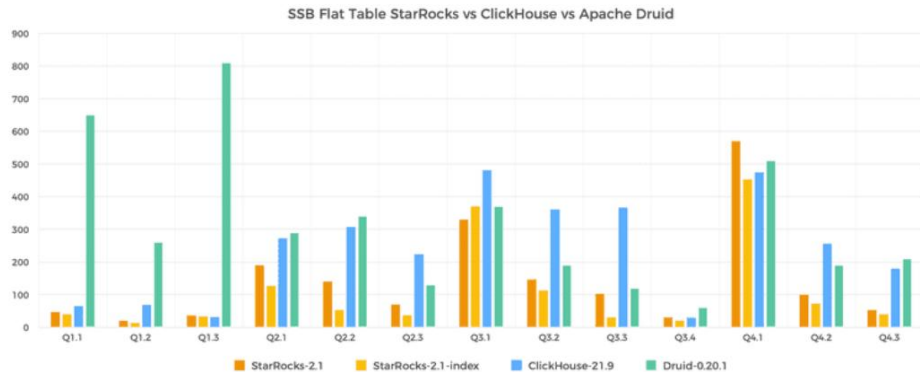
StarRocks优势



StarRocks优势

极致的查询性能

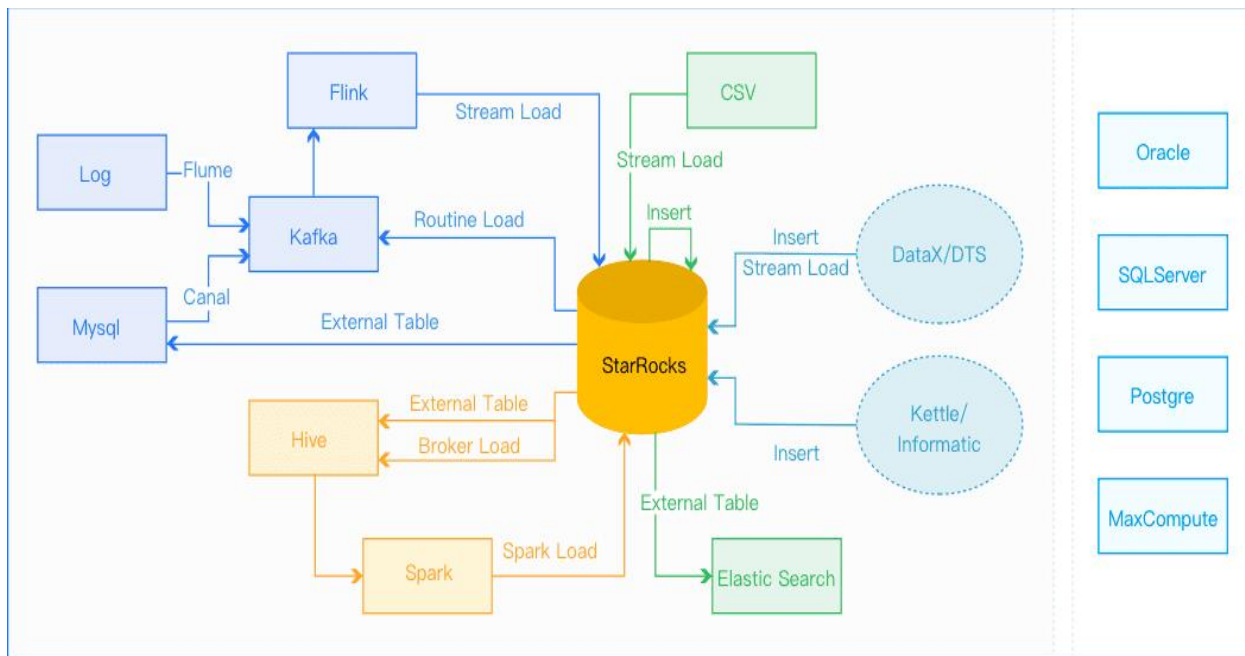
- 分布式执行框架(MPP)
- 列式存储引擎
- 全面向量化引擎
- CBO优化器



StarRocks优势

丰富的导入方式

- Broker Load
- Spark Load
- Stream Load
- Routine Load
- Insert Into



StarRocks优势

运维简单

- 不依赖外部组件，方便缩扩容

丰富的数据模型

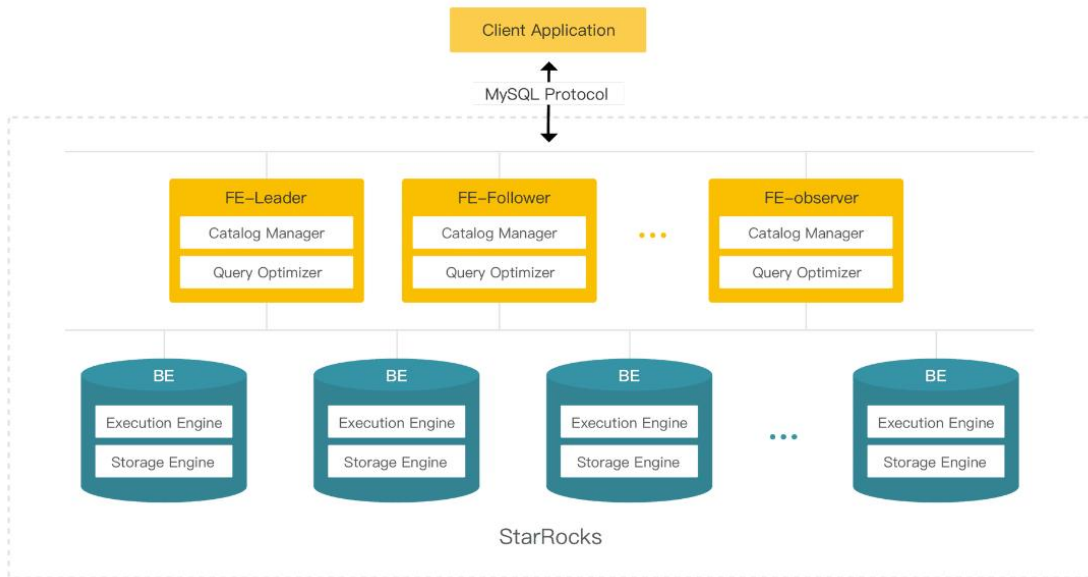
- 支持明细/聚合/更新/主键 四种数据模型，同时支持物化视图

简单易用

- 兼容MySQL协议，支持标准的SQL语法

支持多种外部表

- MySQL ,Elasticsearch , Hive , StarRocks , Apache Iceberg , ...



03

应用场景

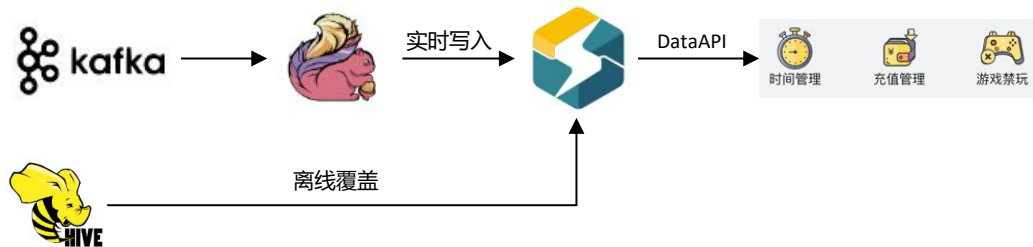


实时计算场景 / 家长监控中心

- 家长监控中心截图



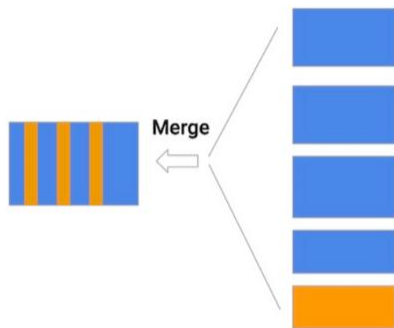
- 数据流转图



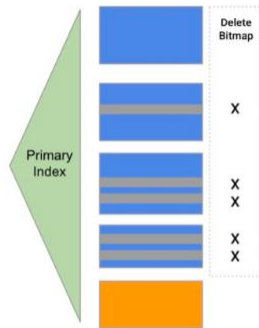
实时更新模型选择

主键模型与更新模型对比

- 更新模型：使用merge on read的方式，在执行时实时合并多个版本的数据
- 主键模型：使用delete and insert 的方式更新数据，主键存储于内存中



Merge on Read



Delete and Insert

主键模型不能使用delete方式删除数据

- 主键模型无法使用delete的方式删除数据，只能通过Stream Load、Broker Load、Routine Load这三种导入方式进行数据的删除

Stream Load 导入语句：

#不指定__op

```
curl --location-trusted -u root: -H "label:lineorder" \  
  -H "column_separator:," -T demo.csv \  
  http://localhost:8030/api/demo_db/demo_tbl1/_stream_load
```

#指定__op

```
curl --location-trusted -u root: -H "label:lineorder" \  
  -H "column_separator:," -H "columns:__op='upsert'" -T demo.csv \  
  http://localhost:8030/api/demo_db/demo_tbl1/_stream_load
```

软删除

- 增加数据有效标志位，通过insert into select 的方式来更变需要删除数据的标志位，达到软删除的目的

```
insert into xxx
select
    game_id,account,event_type,event_time,pay_order_id
    ,event_date,online_time,pay_money
    ,1 as is_delete
from xxx
where event_type in (1,2)
```

变更数据有效标志位

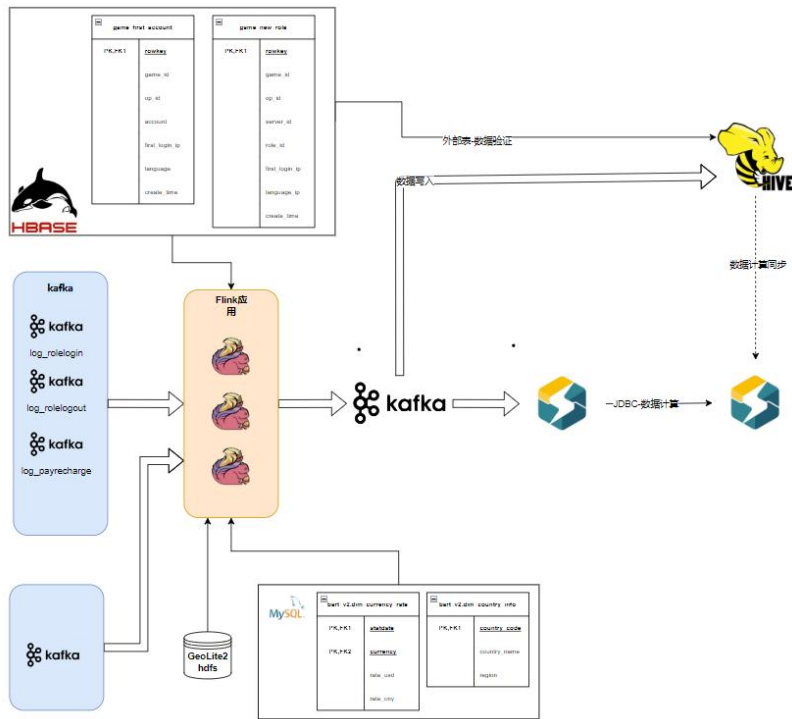
有删除操作的情况下，主键模型与更新模型应该怎样抉择？

- 数据更新频率很频繁
- 更新模型的删除也有一些限制，删除条件复杂的情况下也无法删除
- 软删除后冗余比较少

报表实时指标计算 / 架构介绍

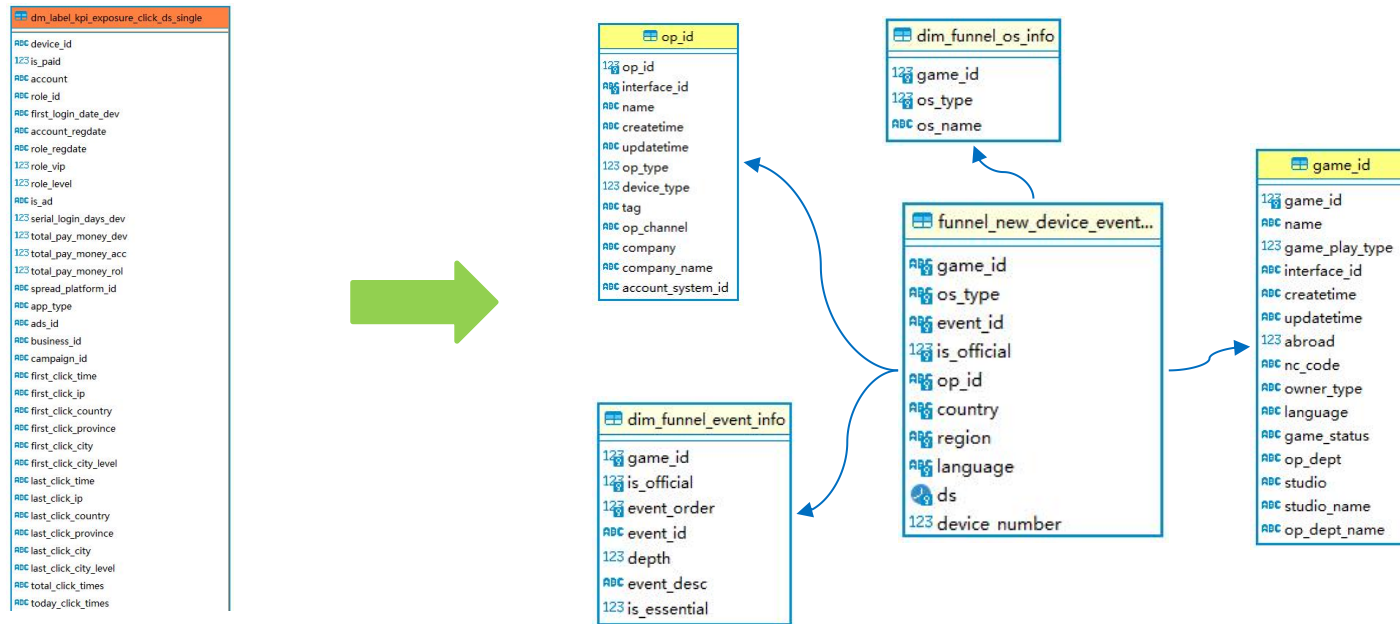
引入StarRocks后

- Flink只做简单的ETL工作
- 统一使用StarRocks做实时指标计算,在StarRocks内部做数据的逻辑分层
- 报表系统直接读取StarRocks结果数据
- 对外提供的数据以DataAPI的形式提供



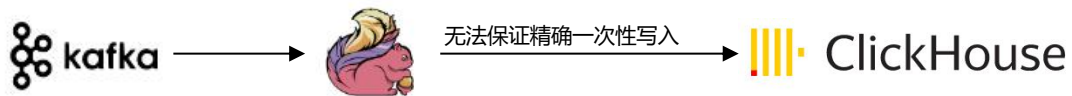
数据关系模型转变

- 会更多的考虑星型模型/雪花模型，以应对更加灵活的多维分析场景



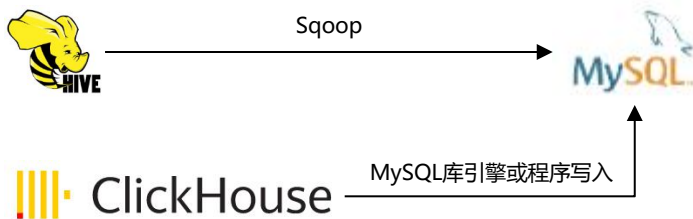
精确一次性保证

- Flink -> Clickhouse 与 Flink -> StarRocks 精确一次性对比



指标存储转变

- 引入StarRocks之前，报表结果数据存MySQL，需要借助外部工具导入

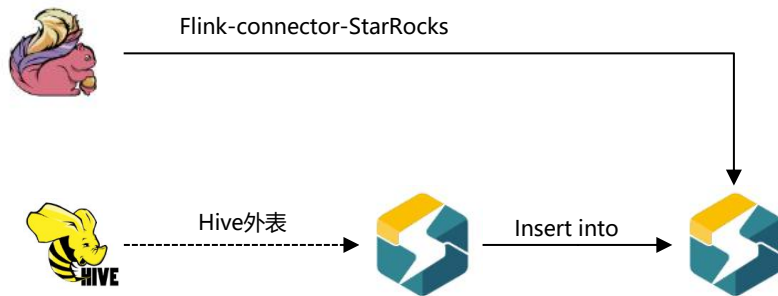


- 引入StarRocks后，以StarRocks为核心，算存一体，查询分析统一



常用数据导入方式

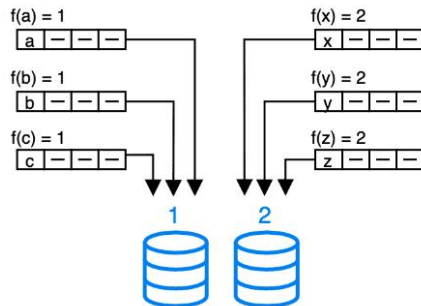
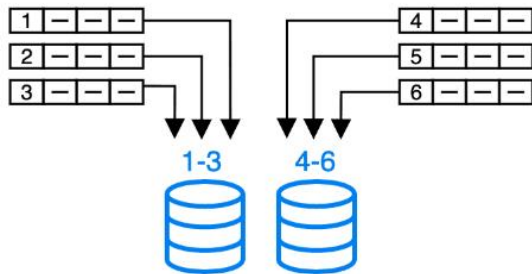
- **实时数据** 使用Flink-connector-StarRocks，其内部实现是通过缓存并批量由 stream load 导入
- **离线数据** 创建Hive外表，用insert into select 的方式直接写入结果表



分区分桶选择

StarRocks使用先分区后分桶的方式，如果不分区，则把整张表作为一个分区


- 分区选择：从数据的管理角度来选择分区键，大多数情况下会选择使用时间分区
- 分桶选择：通常选择高基数的列来作为分桶键，这样可以保证数据在各个桶中尽可能均衡
- 分桶个数：计算数据存储量，将每个tablet设置成 500M 左右



慢查询分析

Profile 与 Plan 查看

- 可以在fe/log/fe.audit.log中看到所有查询和慢查询信息
- Profile是BE执行后的结果，包含了每一步的耗时和数据处理量等数据，可以通过StarRocksManager的图形界面查看详细内容

 **StarRocks** [system](#) [backends](#) [logs](#) [queries](#) [sessions](#) [variables](#) [ha](#) [help](#)

Finished Queries

This table lists the latest 100 queries

User	Default Db	Sql Statement	Query Type	Start Time	End Time	Total	Query State	Profile
default_cluster:datawarehouse		/* ApplicationName=DBeaver 6.3.0 - Main */ select FROM_UNIXTIME(event_time) from realtime_funnel_new_device_event_detail where ds = '2022-06-06' and game_id = 352 and event_id = 'CreateRole' group by FROM_UNIXTIME(event_time) LIMIT 0, 10000	Query	2022-06-09 15:38:10	2022-06-09 15:38:10	19ms	EOF	1cc6796a-e7c7-11ec-b296-b4969133287c
default_cluster:datawarehouse		/* ApplicationName=DBeaver 6.3.0 - Main */ SELECT kc.CONSTRAINT_NAME,kc.TABLE_NAME,kc.COLUMN_NAME,kc.ORDINAL_POSITION FROM INFORMATION_SCHEMA.KEY_COLUMN_USAGE kc WHERE kc.TABLE_SCHEMA='realtime' AND kc.REFERENCED_TABLE_NAME IS NULL AND kc.TABLE_NAME='funnel_new_device_event_detail' ORDER BY kc.CONSTRAINT_NAME,kc.ORDINAL_POSITION	Query	2022-06-09 15:38:07	2022-06-09 15:38:07	3ms	EOF	1ac5ee6d-e7c7-11ec-b296-b4969133287c
default_cluster:datawarehouse		/* ApplicationName=DBeaver 6.3.0 - Main */ select * from realtime_funnel_new_device_event_detail where ds = '2022-06-06' and game_id = 352 and one_id = '2022041600000352002319e4c97e081d5b9b7ddc43a9bbb72a7' LIMIT 0, 10000	Query	2022-06-09 15:38:07	2022-06-09 15:38:07	34ms	EOF	1ab9b969-e7c7-11ec-b296-b4969133287c

04 未来规划



未来规划

- 将剩余的实时场景全部迁入StarRocks
- 建立以StarRocks为核心统一的查询分析平台
- DataAPI服务集成StarRocks
- 完善StarRocks监控

非常感谢您的观看

YOOZOO
游族网络

| DataFun.

