

Apache Doris 在公交行业 的应用实践

杜忠猛

智达信科 数据研发部大数据专家

目录

1. 行业背景
2. 平台架构演进
3. 公交应用实践
4. 总结与展望

1 行业背景

智达科技成立于2002年，是国内专注于“公共交通智能化”领域的专业化高新技术企业。公司聚焦于“有核心技术和产品的智慧出行整体解决方案提供商”核心战略，以“数字底座”为核心技术，产品线覆盖“企业、政府、公众”三大客户群体，为公交行业数字化转型提供一站式解决方案。

覆盖全国60余主要城市，直辖市占有率50%



过去十年，国家政策支撑为公交行业发展带来了历史的发展机遇，随着这十年的高速发展，公交行业的基础设施和治理手段有了质的飞跃，如智能硬件、智能调度、智能助手等的应用。

随着发展，供给侧和需求侧的的矛盾日益突出，一方面是人民日益增长的灵活且多样化的出行需求，一方面是传统公交经营模式难以升级的困境，使得公交行业的数字化升级成为了必由之路。

为解决公交行业数字化转型面临的问题，智达决定构建数据中台。

智达数据中台实现了收、存、治、算、用，五大基础能力，让数据从采集、存储、处理、计算到应用变成标准化的能力。实现业务数据的可视化，辅助集团管理层科学决策，推动业务系统优化。

2 平台架构演进

一期架构

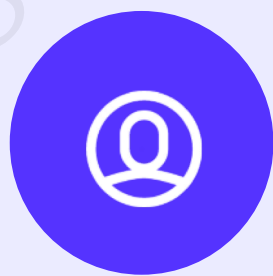


一期架构的问题



数据时效性

- 对于日增千万级别的数据，Mysql无法承载
- Hive数仓落盘时效性10分钟级别



查询并发

- Presto测试查询并发百级别
- 查询不稳定，对于同一个SQL两次查询耗时相差数倍



查询延迟高

- Presto查询时效性在秒至分钟级别
- 对于大数据量实时OLAP响应及时性差



计算复杂

- 由于Mysql承载数据量有限，大量计算逻辑、状态需要在内存完成
- 开发周期长

平台需求

高性能

- 直接承载后端API查询
- 通用的SQL查询语法
- 查询响应快速，毫秒至秒级别
- 可以支持上千并发

OLAP能力

- 明细数据级别多表join能力
- 支持分区，单表百亿级别数据量
- 支持复杂SQL逻辑，索引，视图

运维便捷

- 故障自动恢复
- 集群扩展方便
- 社区活跃，方便与社区沟通

功能丰富

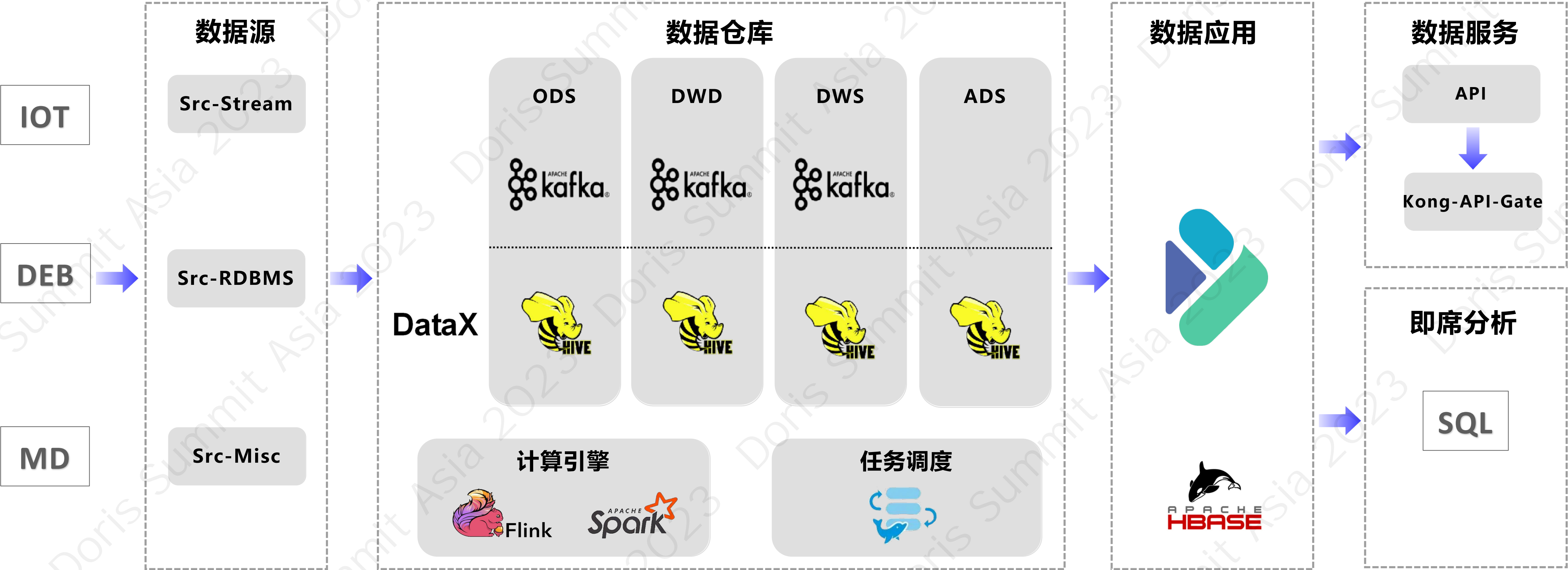
- 可以方便地与Flink、Spark框架结合使用
- 数据导入便捷，数据导入时效性高，秒级
- 外部表能力，可以分析其他存储数据，简化整体



OLAP 技术选型

OLAP引擎	标准SQL	查询延迟	导入延迟	导入语义	Join支持	明细查询	并发	生态	运维难度
Doris	非常完善	低	实时	Exactly-Once	支持	支持	支持	独立	低
ClickHouse	较完善	低	实时	At-least-Once	一般	支持	较差	独立	中
Druid	较完善	低	实时	Exactly-Once	较低	支持	支持	不独立	低
Kylin	非常完善	非常低	分钟级	无	不支持	不支持	支持	不独立	高
Presto	非常完善	一般	分钟级	无	支持	支持	支持	不独立	低

二期架构



Doris 测试结果

-- 定位数据表结构

```
CREATE TABLE `TI_GPS_TABLE` (  
  `RQ` date NOT NULL COMMENT 'GPS日期',  
  `GPSHOUR` int NOT NULL COMMENT 'GPS时间小时',  
  `GPSMIN` int NOT NULL COMMENT 'GPS时间分钟',  
  `GPSTIME` bigint NOT NULL COMMENT 'GPS数据时间',  
  `BUSNAME` varchar(50) NOT NULL COMMENT '车辆名称',  
  `LATITUDE` bigint REPLACE_IF_NOT_NULL NULL COMMENT '纬度*36000',  
  `LONGITUDE` bigint REPLACE_IF_NOT_NULL NULL COMMENT '经度*36000',  
  `SPEED` double REPLACE_IF_NOT_NULL NULL COMMENT '速度 km/h'  
  ...  
)  
AGGREGATE KEY(`RQ`, `GPSHOUR`, `GPSMIN`, `GPSTIME`, `BUSNAME`)  
...
```

-- 定位数据提取，数据量15亿

```
SELECT * FROM db.TI_GPS_TABLE  
WHERE rq = '' AND GPSHOUR = 10 AND BUSNAME = '123';
```

-- 查询指定车辆定位数据，关联组织机构

```
SELECT  
  a.*,b.*  
FROM  
  (SELECT * FROM db.TI_GPS_TABLE WHERE rq = '' AND GPSHOUR = 10  
  AND busname = '123') as a  
LEFT JOIN  
  (SELECT * FROM db.TD_MD_CL_TABLE tmhcd WHERE rq = '') as b  
ON a.rq = b.rq AND a.busname = b.clzbh;
```

数据回放

提升70%

Join 场景

提升40%

每秒 QPS

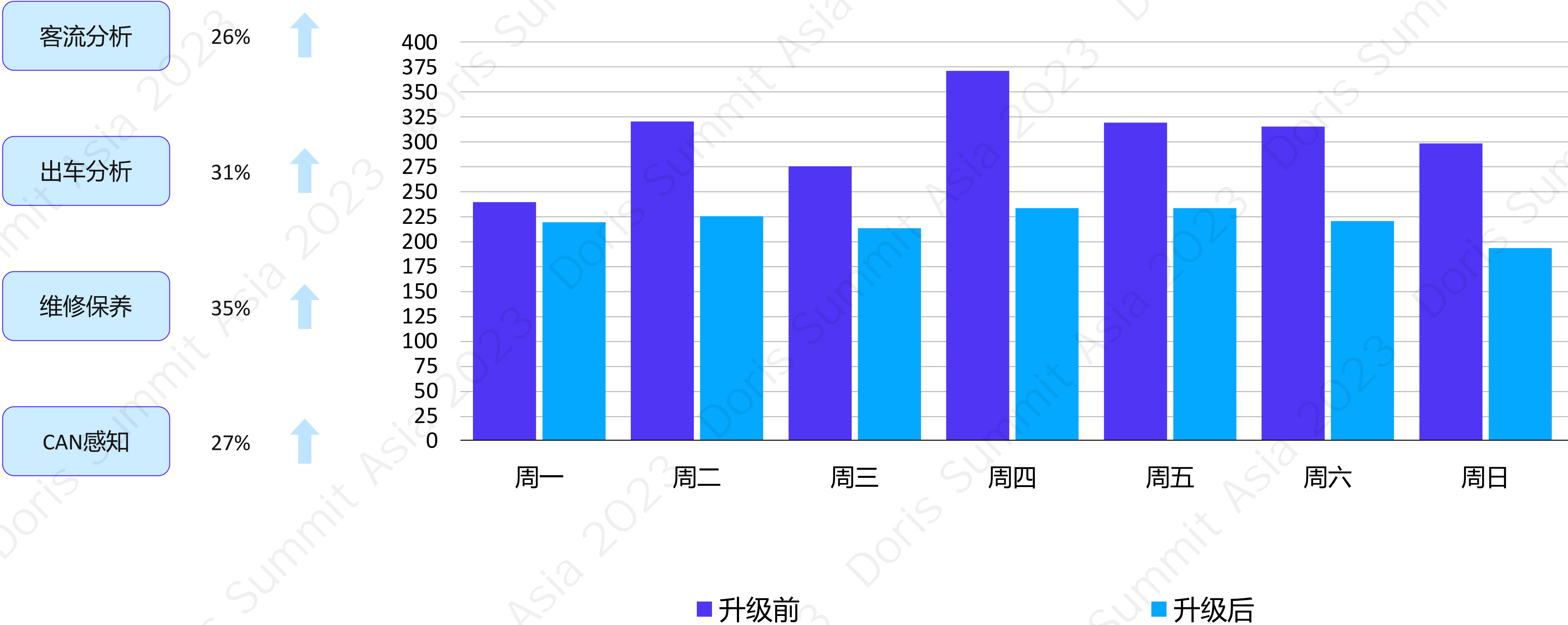
提升10倍

3 公交应用实践

优化实践

向量化升级，提高查询性能

95分位查询用时-ms



Stream Load 在智达的使用

第一阶段

使用重试机制，写入失败时，定时重试，重试后仍然失败的批次写入hdfs，Spark定时任务检测hdfs，写入Doris。

第二阶段

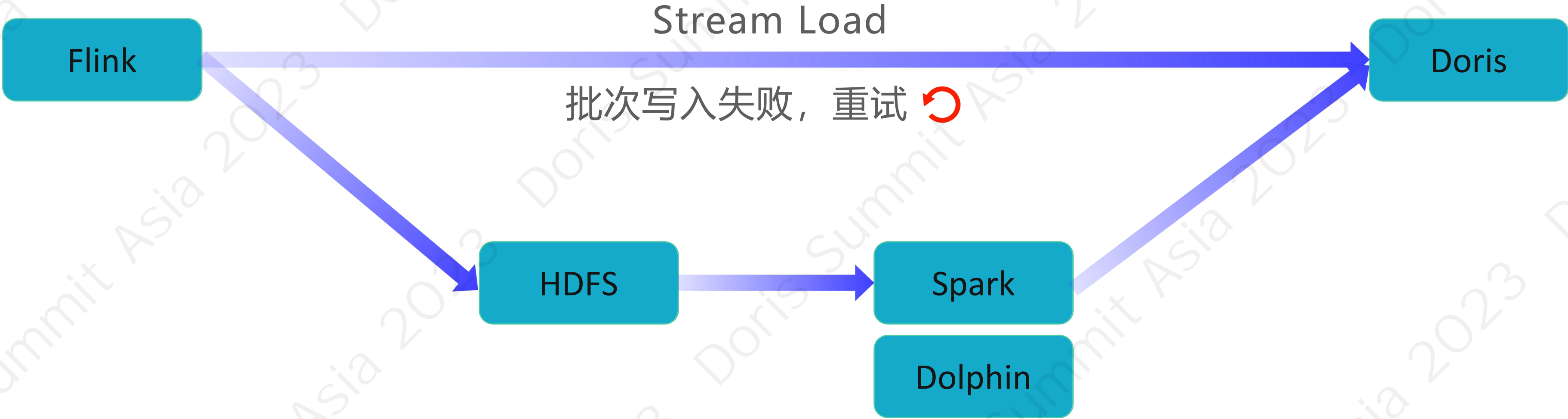
Flink Checkpoint保证数据精准一次性写入。

第三阶段

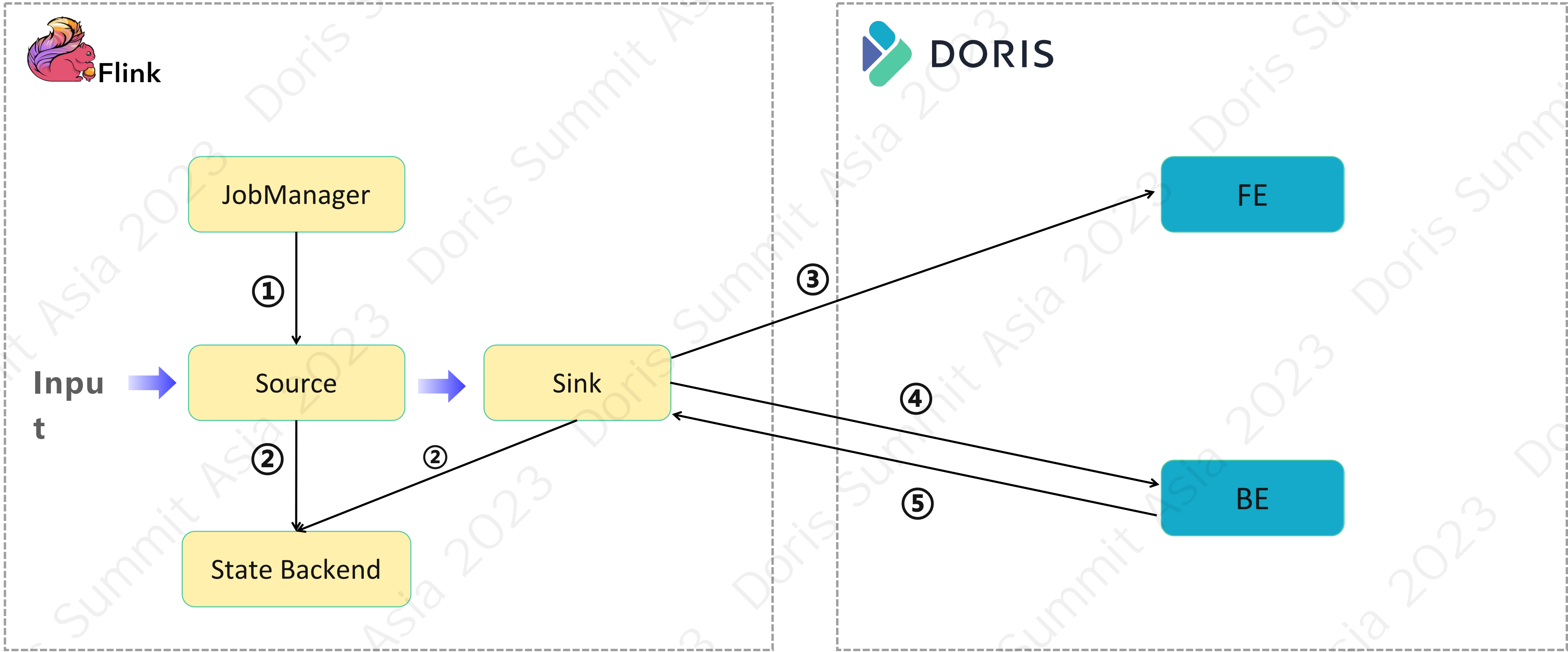
使用Flink Doris Connector保证数据精准一次性写入。



第一阶段



第二阶段



① Start Checkpoint

④ Stream Load

② Snapshot state

⑤ Aborted

③ Sink Request

第三阶段

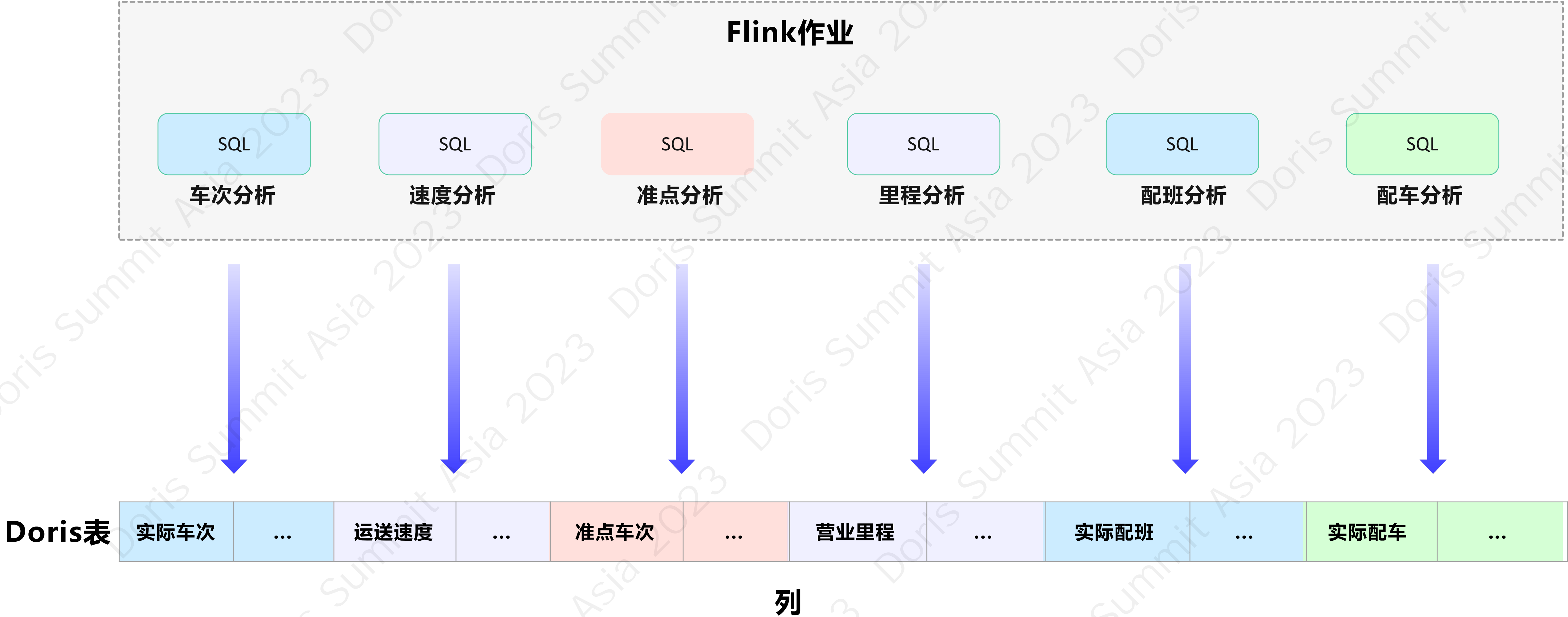
```
-- 创建“刷卡刷码”数据Kafka数据源
CREATE TABLE icqrKafkaTable (
  rq STRING,      // 检录日期
  hour STRING,    // 检录小时
  lineID STRING,  // 线路编号
  perkType STRING, // 检录类型, 刷卡/扫码
  direction STRING, // 线路方向
  `ts` TIMESTAMP(3) METADATA FROM 'timestamp',
  procTime as PROCTIME()
) WITH (
  'connector' = 'kafka',
  'topic' = 'ICQR',
  ...
)

-- 创建Doris写入表
CREATE TABLE flink_doris_sink (
  RQ STRING,      // 日期
  HOUR STRING,    // 小时
  LINID STRING,   // 线路编号
  LINENAME STRING, // 线路名称, 新增
  PERKTYPE STRING, // 检录类型, 刷卡/扫码
  DIRECTION STRING, // 线路方向
  RC STRING       // 人次, Doris字段聚合类型为SUM
) WITH (
  'connector' = 'doris',
  ...
)
```



```
-- 开窗汇总乘客客流人次, 并关联Mysql线路维表, 增加线路名称字段
insert into flink_doris_sink
select
  c.rq,
  c.hour,
  c.lineID,
  c.lineName,
  c.perkType,
  c.direction,
  count(1)
(select
  a.rq,
  a.hour,
  a.lineID,
  b.lineName,
  a.perkType,
  a.direction,
  a.procTime
from icqrKafkaTable a
LEFT JOIN Mysql_xl FOR SYSTEM_TIME AS OF a.procTime as b // 时态表
  on a.lineId = b.lineId) as c
GROUP BY c.rq,c.hour,c.lineId,c.lineName,c.perkType,c.direction,
  TUMBLE(c.procTime,INTERVAL '30' SECONDS);
```

部分列更新，提高运调业务开发效率



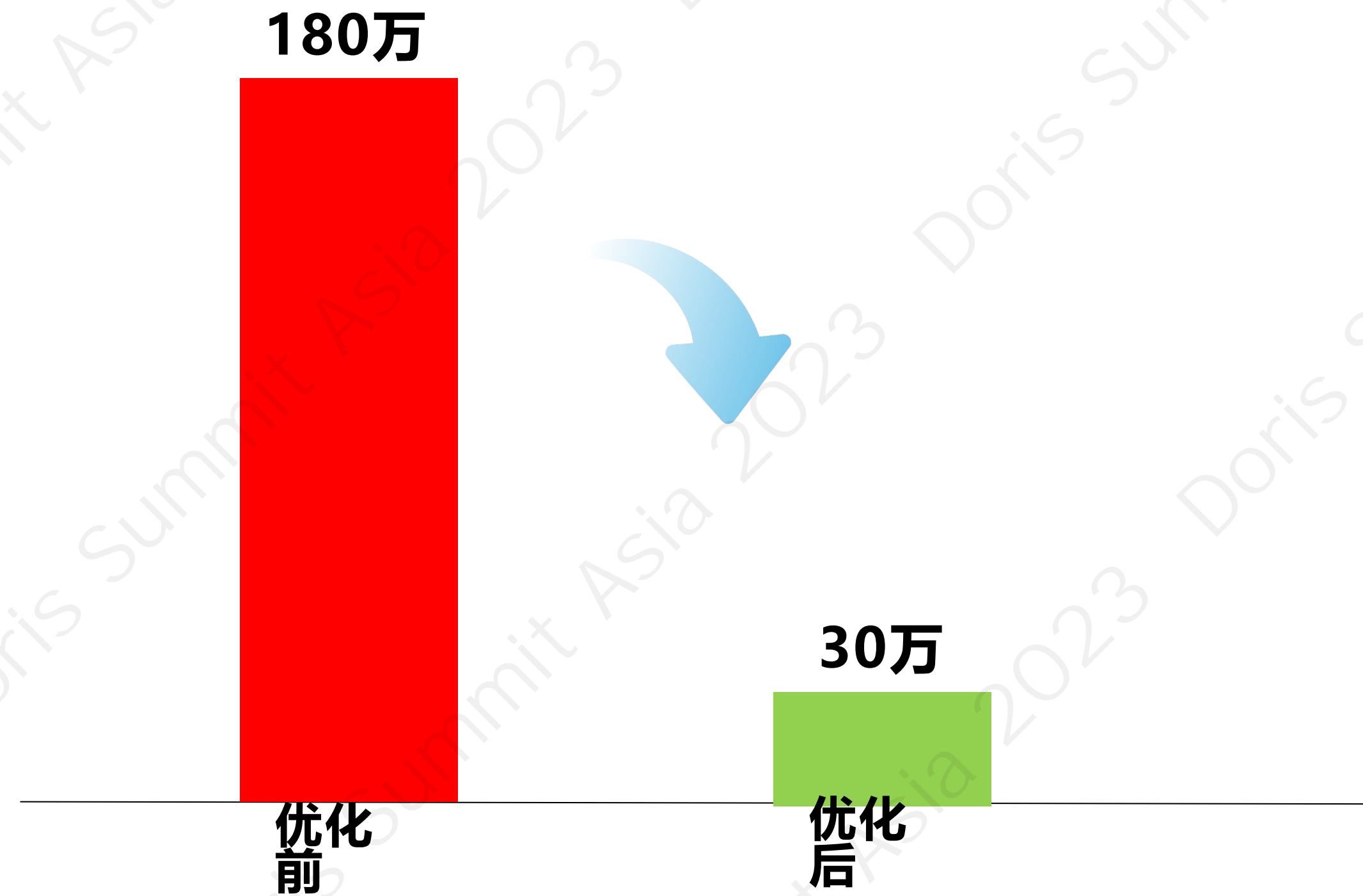
元数据压力

问题现象

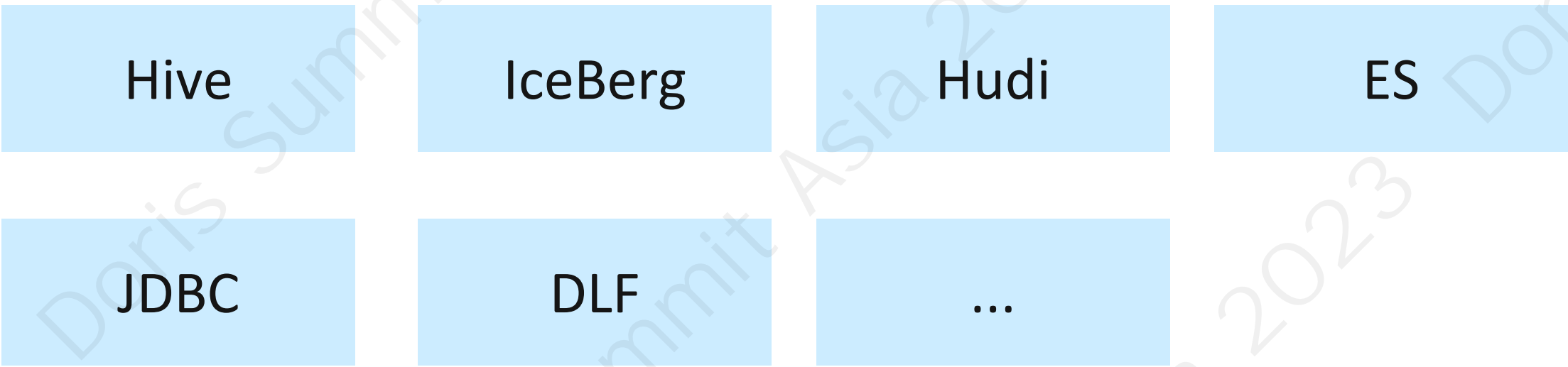
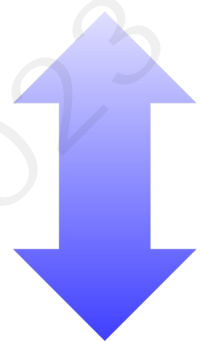
- 建表耗时由原来的秒级增加到分钟级
- 建表分区数多，经常会超时

解决方案

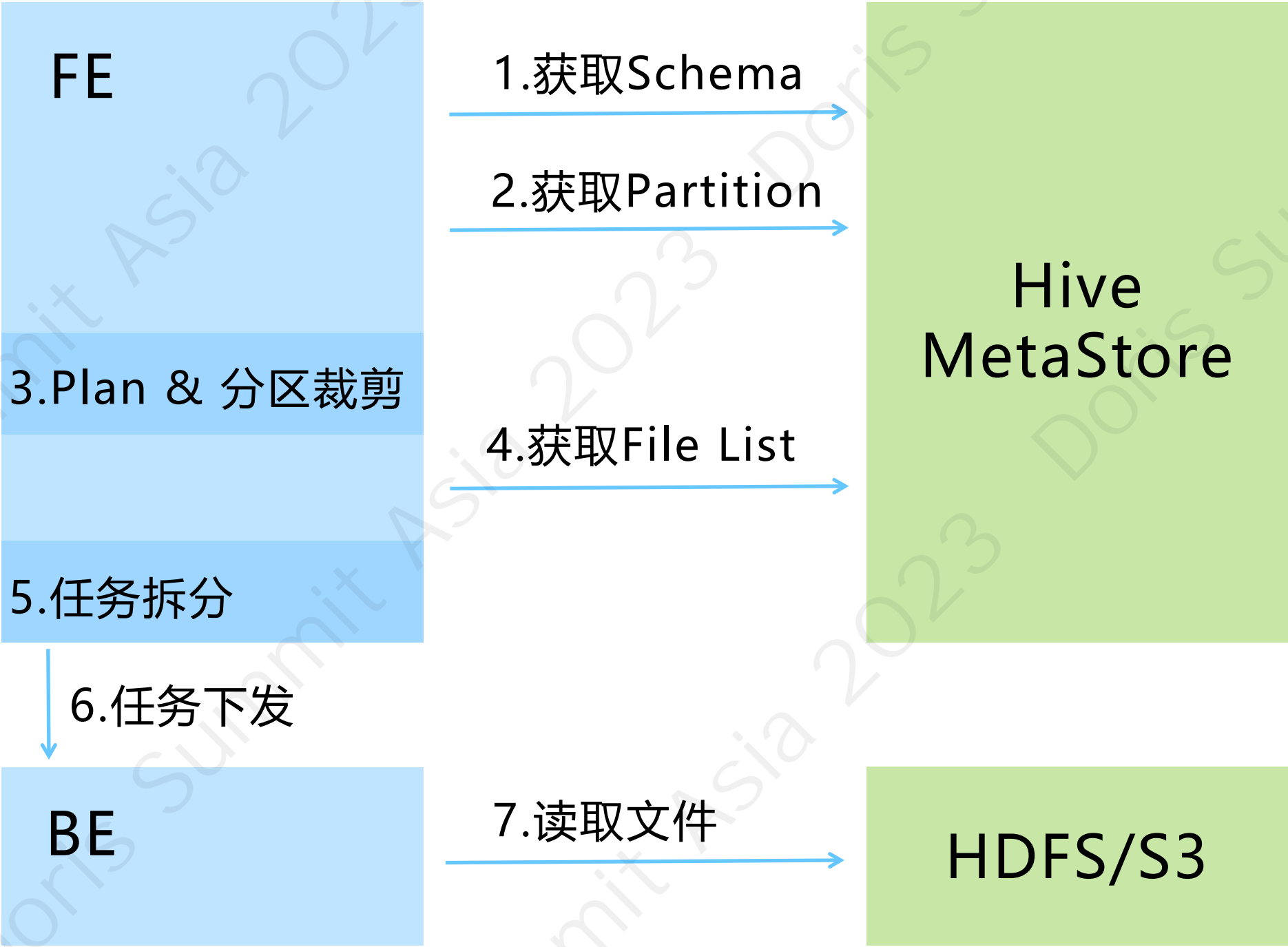
- 官网建议单个Tablet 1G-10G
- 采用方案：
 - ADS层表，单Tablet 50M
 - ODS层大数据量表，单Tablet 100M-500M



数据湖分析



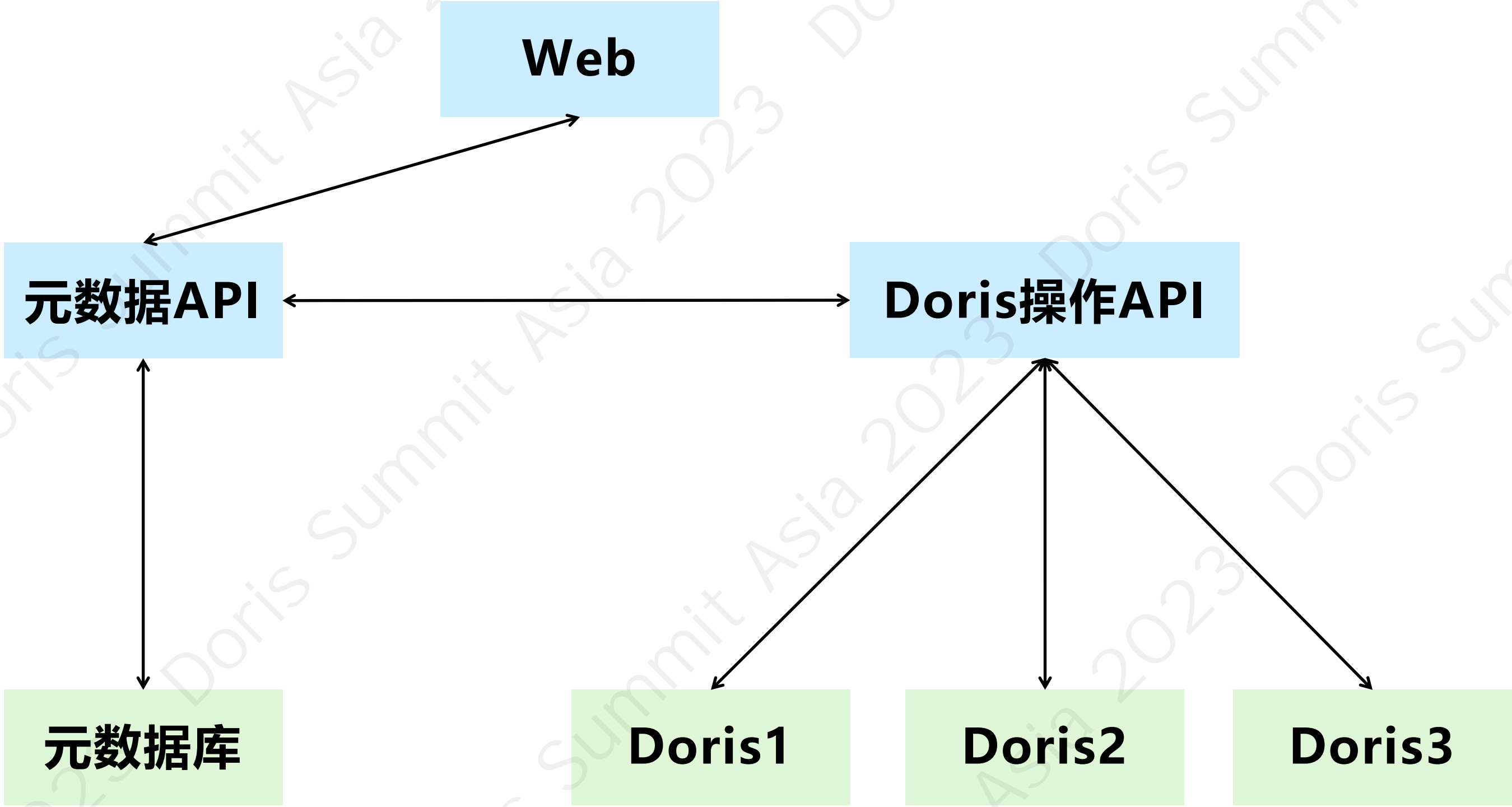
Hive Catalog访问流程



元数据管理

问题点

- 如何检索多个Doris集群表的元数据
- 有多少张生产表、多少张测试表
- 表是做什么的，从哪一个业务系统产生
- 表处于数仓中的哪一层
- 表是哪一种数据模型
- 表中的数据是什么类型，
主数据、事务数据还是感知数据
- 数据的保留周期是多久
- 谁创建，谁修改过表



元数据管理

功能点

- 多集群元数据管理
- 数据库创建、检索
- 表创建(表单、SQL)、检索、字段检索
- 业务域、数据维度、数据密级、数据类型、操作人信息

集群Doris集群

框架实例mrs_doris集群-doris

库名请输入数据库名称

库中文名请输入数据库中文名

生产是

搜索重置

+ 新增

集群	框架实例	库	库中文名	数据库IP	数据库端口	生产环境	操作
Doris集群	mrs_doris集群-doris					1	查看 修改 删除
Doris集群	mrs_doris集群-doris					1	查看 修改 删除
Doris集群	mrs_doris集群-doris					1	查看 修改 删除
Doris集群	mrs_doris集群-doris					1	查看 修改 删除
Doris集群	mrs_doris集群-doris					1	查看 修改 删除

集群Doris集群

框架实例mrs_doris集群-doris

库名请选择库名

表名请输入表名

生产是

数据维度请选择数据维度

分区类型请选择分区类型

保留周期请选择保留周期

字段名请输入字段名

搜索重置

+ 新增(表单)

+ 新增(SQL)

+ 提取元数据

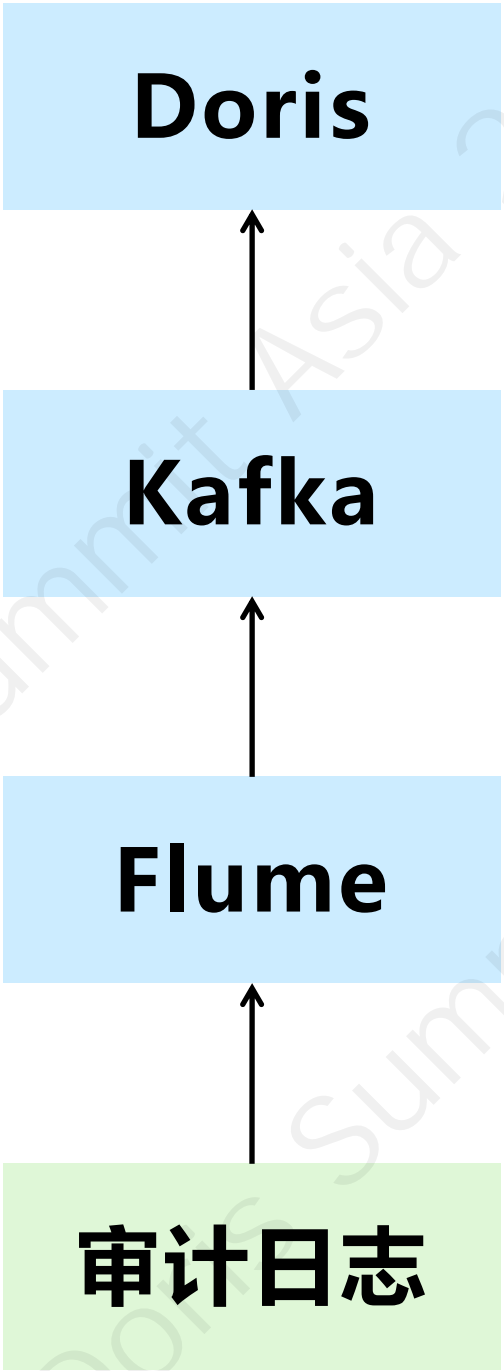
集群	框架实例	库名	表名	表中文名	数据维度	生产环境	操作
Doris集群	mrs_doris集群-doris				DWD	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				ADS	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				ADS	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				ADS	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				DIM	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				DIM	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				DIM	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				DIM	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				DIM	1	查看 修改 删除
Doris集群	mrs_doris集群-doris				DIM	1	查看 修改 删除

共 809 条 10条/页 1 2 3 4 5 6 ... 81 前往 1 页

元数据同步

- 元数据校验
- 元数据自动同步、告警
- 一键同步表元数据

Query 监控



用户日查询量	各Doris用户账号日查询量、查询平均延迟
慢查询SQL	用户的慢查询SQL列表
错误查询SQL	用户的错误查询SQL列表
扫描数据量	扫描数据量超过百万行SQL列表

- SQL改写
- BitMap索引
- BloomFilter

4 总结与展望

应用现状

应用表

800 张+

日查询

280 万+

数据条数

5000 亿+

指标数

600 +

API

550 +

未来展望

- 将 Doris 融入到更多的业务场景中
- 承载更多的实时+离线业务，推动离线指标实时化
- Doris 新版本跟进，将更多新特性融入并改善现有业务
- 积极与社区交流，为社区提供更多 Doris 在公交领域的使用场景实践



获取更多社区动态与最佳实践

Apache Doris 官方平台:

- Apache Doris 官网: doris.apache.org
- Apache Doris GitHub: github.com/apache/doris/

获取更多峰会资料:

- Doris Summit 峰会官网: doris-summit.org.cn
- Doris Summit 峰会回放: <https://space.bilibili.com/1196172099/channel/collectiondetail?sid=1824324>