

# Apache Doris

## 在货拉拉的应用与稳定性实践

杨秋吉 货拉拉 大数据技术与产品部大数据专家  
张斌 货拉拉 大数据技术与产品部高级大数据工程师

# 目录

1. 背景介绍
2. 基于 Apache Doris 的 OLAP 架构演进
3. Apache Doris 应用实践
4. 稳定性挑战的应对实践
5. 未来展望

# 1 背景介绍



# 货拉拉介绍

360  
国内城市

68万  
月活司机

950万  
月活用户

8+  
业务线

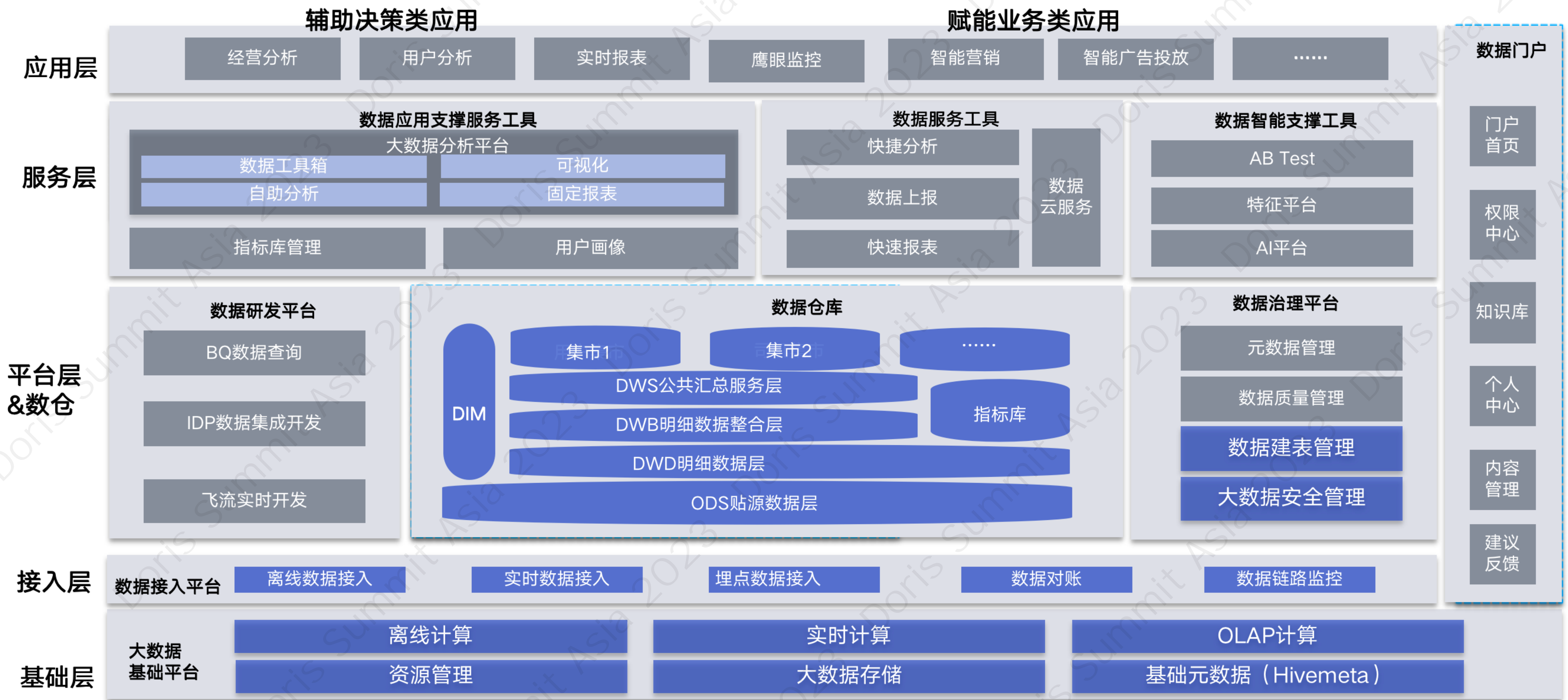
7+  
IDC

1000+  
机器数

20PB+  
存储量

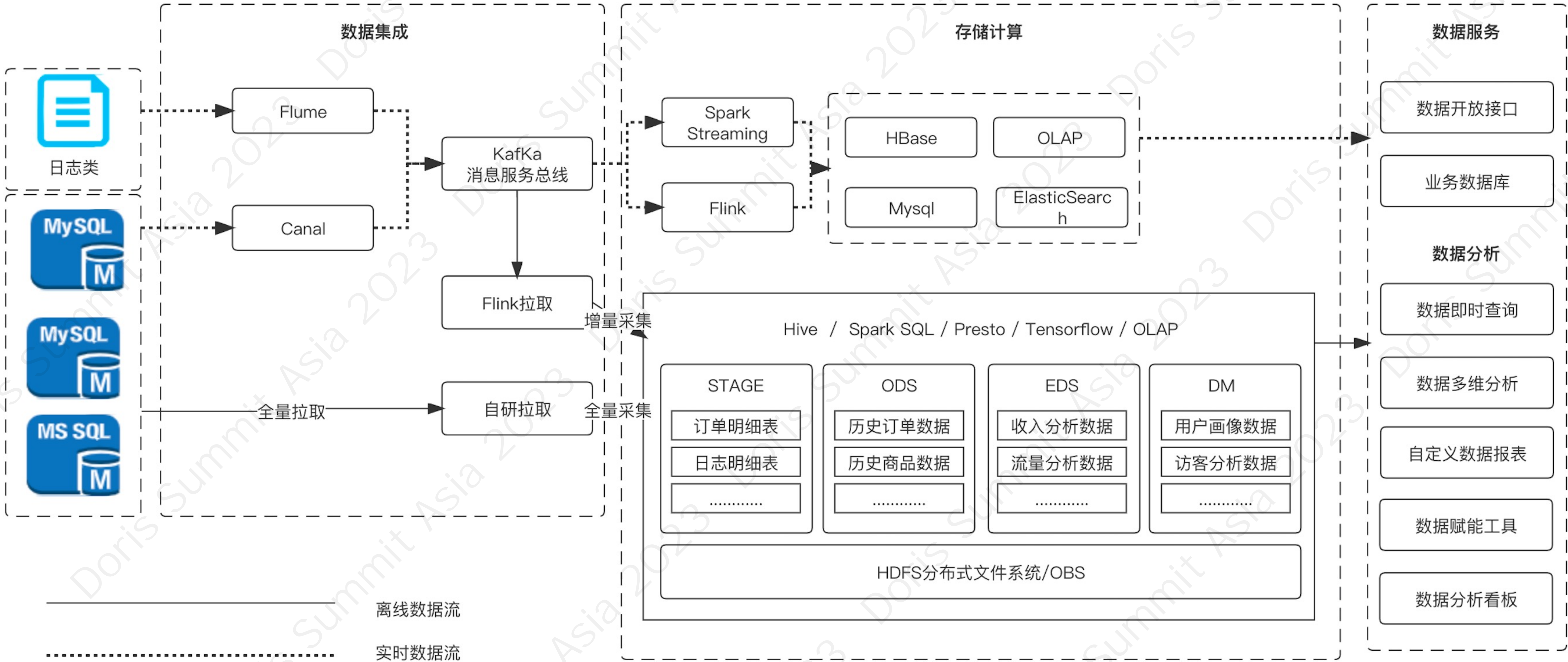
20k+  
日均任务数

# 货拉拉-大数据介绍



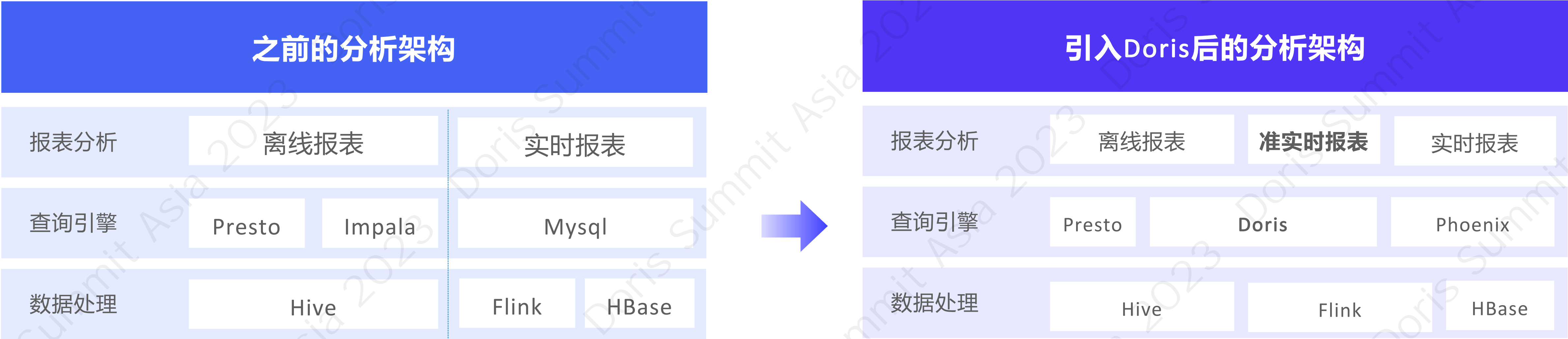


# 货拉拉-大数据介绍



## 2 基于 Apache Doris 的架构演进

# 基于 Apache Doris 的架构演进



## 存结果数据

- 维度爆炸，Mysql存储成瓶颈
- 开发成本高，效率低
- 不具备下钻分析能力
- 不具备即席分析能力

## 引入Doris引擎

- 引入时间：2022年
- 查询性能强：P90 2秒
- 使用简单：使用Mysql协议、高度兼容Mysql语法、完善的文档
- 版本：1.1.3和1.2.4
- 规模：约60台BE

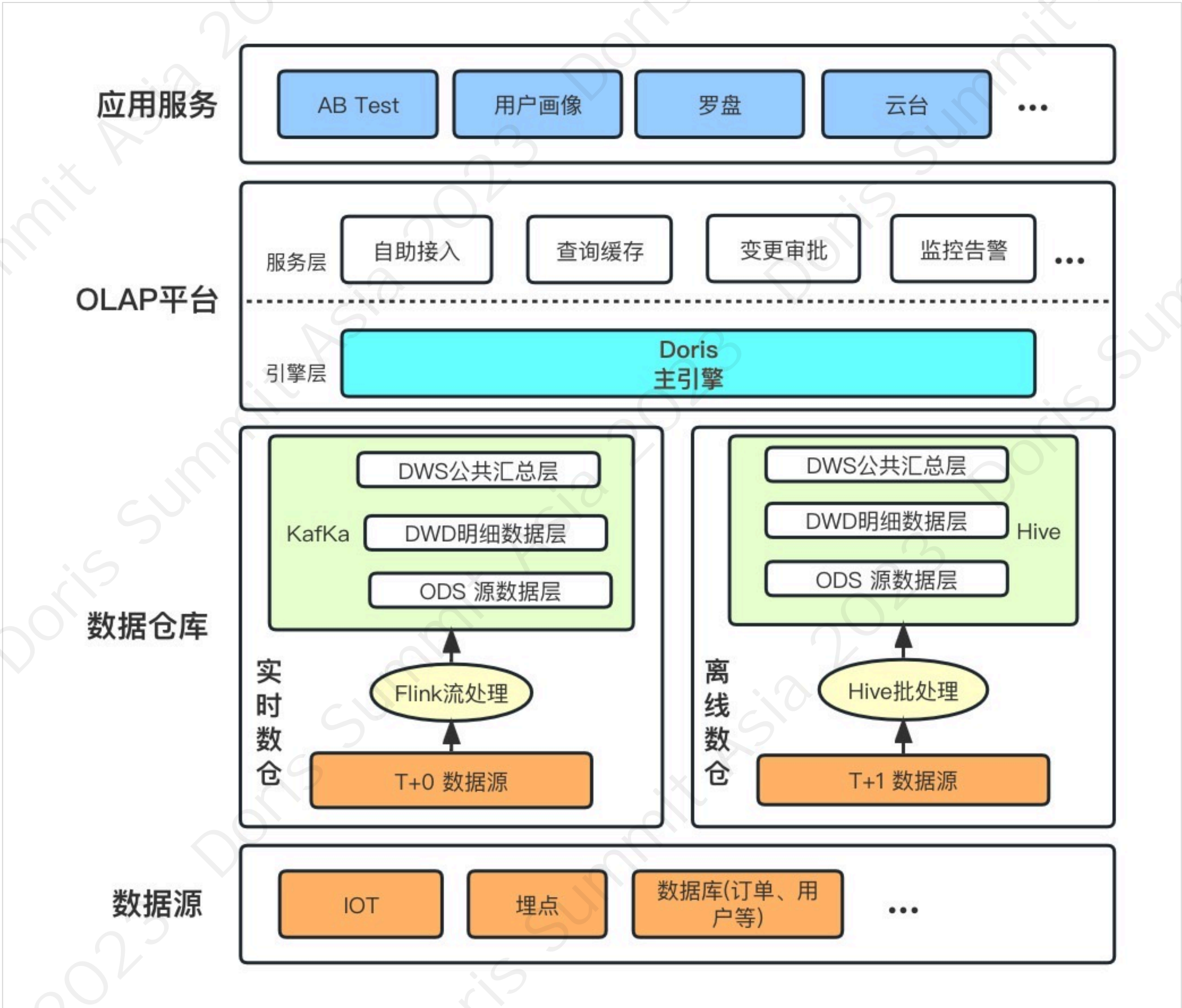
## 准实时报表

- 数据时效性：5分钟 ~ 15分钟
- 减缓原实时报表压力
- 基于Doris构建准实时数仓
- 支持分钟级调度Doris Insert SQL



# 3 Apache Doris 应用实践

业务介绍



# 应用实践



Point 1  
数据导入



Point 2  
数据查询



Point 3  
数据研发



Point 4  
Doris调优/制定规范



## Point 1 - 数据导入

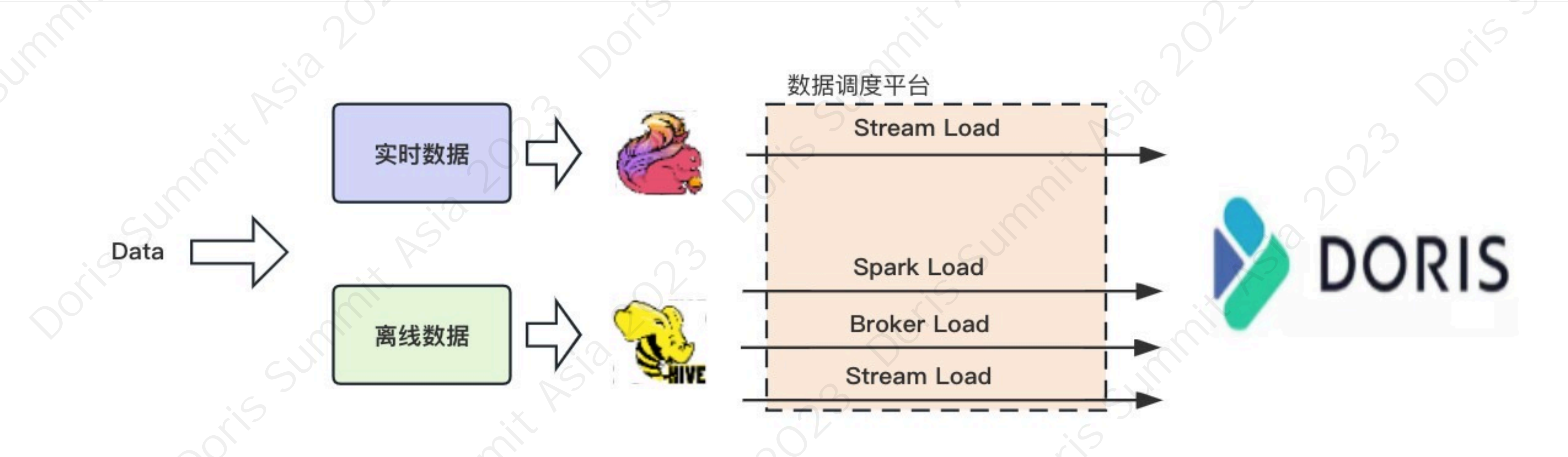
货拉拉内部主要通过Stream Load与Spark Load以及少量的Broker Load方式进行Doris的数据导入，并分为实时与离线两个部分：

### 实时数据写入：

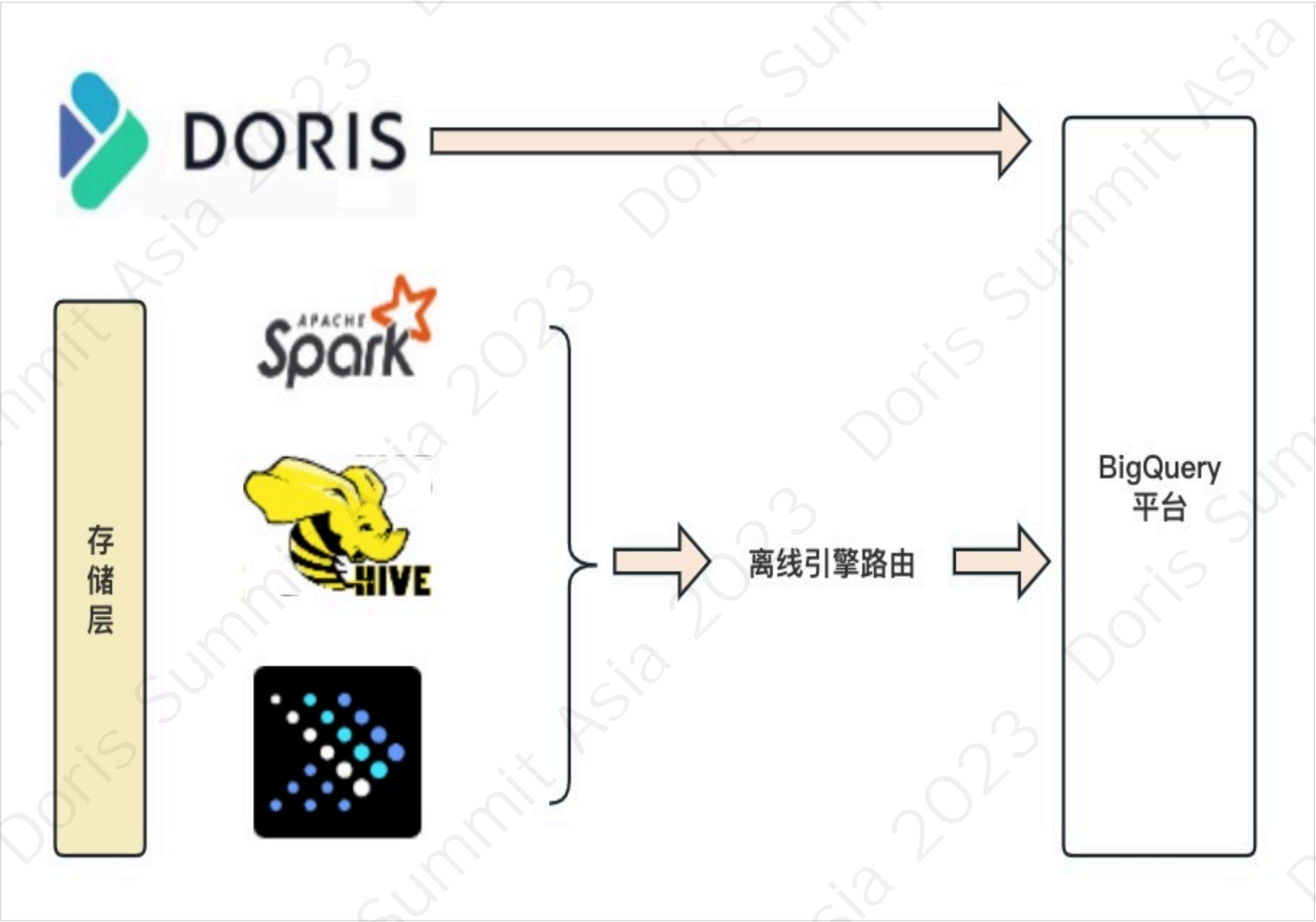
- Flink SQL通过Stream Load**两阶段**提交的方式将数据**微批**写入Doris，具备exactly-once保障

### 离线数据写入：

- SparkLoad将Hive外表的OBS数据跨云导入Doris，并**支持insert overwrite语义**
- 对于小数据量场景，支持Stream Load导入



## Point 2 - 数据查询



### 数据查询：

- 统一查询入口：数据导入Doris后，用户可以在公司自研的BigQuery平台进行自助查询；
- UDF扩展：为满足业务的事件分析需求，研发Doris的UDF函数，如：分位数预聚合 UDAF
- 慢查询治理：慢查询自动打标，查询优化



Point 3 - 数据研发

交互查询

doris自带的web页面提供了管理功能，为了便于用户进行数据开发，将查询doris功能整合到了公司的bigquery平台，方便用户使用。

1 SELECT

2 biz\_date,

3 `city\_name` AS `区域`,

4 `biz\_type\_name` AS `业务线`,

5 SUM(`independ\_demand\_match\_rate`) AS `独立需求订单配对率`,

6 sum(matched\_gmv) as `配对GTV`

7 FROM

8 `hll\_gpt`.`dm\_app\_chatgpt\_kexin\_kpi\_city\_final\_v2\_1d\_in`

9 WHERE

10 `city\_name` = '全国'

11 AND `biz\_type\_name` = '货运小车'

12 AND `biz\_date` >= '2023-07-24'

13 AND `biz\_date` <= '2023-07-26'

14 GROUP BY

15 biz\_date,

16 `biz\_type\_name`,

17 `city\_name`

Command+Enter: 执行SQL, Command+F: 搜索、替换, Command+Z

▶ 查询

⬇ 查询并下载

⊙ 终止

格式Sql

保存模板

清除内容

运行日志

资源总数

资源概览

结果(3)

序号	biz_date	区域	业务线	独立需求订单配对率	配对GTV
1	2023-07-25	全国	货运小车		
2	2023-07-26	全国	货运小车		

任务调度

在公司自研的任务调度平台上，进行数据研发，并能够设定任务的调度时间（分钟级/小时级）。

数据时间	任务类型	周期	主题	状态	次	实际运行情况
2023-09-20 11:25:00	Doris SQL	5分钟	测试	运行	1/3	起: 2023-09-20 11:31:25 止: -- 运: 00:00:57
2023-09-20 11:20:00	Doris SQL	5分钟	测试	成功	1/3	起: 2023-09-20 11:25:21 止: 2023-09-20 11:26:36 运: 00:01:15
2023-09-20 11:15:00	Doris SQL	5分钟	测试	成功	1/3	起: 2023-09-20 11:21:22 止: 2023-09-20 11:22:35 运: 00:01:13
2023-09-20 11:10:00	Doris SQL	5分钟	测试	成功	1/3	起: 2023-09-20 11:15:23 止: 2023-09-20 11:16:36 运: 00:01:13



Point 4 - Doris 调优

维度	配置项	含义	默认值	修改值
查询	exec_mem_limit	单个查询的内存限制	2G	8G
查询	parallel_fragment_exec_instance_num	BE上执行实例的个数	2	4
查询	enable_sql_cache/enable_partition_cache	SQL级/分区级缓存	FALSE	TRUE
查询	enable_profile	查看任务的profile信息	FALSE	TRUE
导数	streaming_load_json_max_mb	控制单次streamLoad数据量	100	150
导数	compaction_task_num_per_disk	并发compaction数量	2	4
导数	number_tablet_writer_threads	tablet写线程数	16	32
导数	flush_thread_num_per_store	每个store用于刷新内存表的线程数	2	3

# Point 4 - 业务准入规范

CheckList

(重要, 填写后请打勾)

稳定性要求

☐ 可用性  $\geq xx.xx\%$  (无特殊需求可不填写)

导出

实时 (如无需实时导出, 则无需填写)

☐ 数据写入速度  $\leq x$  行/分钟, 约  $y$  Byte/行

☐ 延时: 毫秒/秒/分钟级?

离线 (如无需离线导出, 则无需填写)

☐ 导出方式: sparkload/brokerload/streamload

☐ 每天导入分区数: 一天一个分区/一天 $n$ 个分区

☐ 每天是否需覆盖旧分区数据: 是/否

存储

☐ 日存储数据量 (如为hive表, 可根据元初表详情存储量估算):  $\geq x$  G/天

☐ 数据保留天数: 1个月/2个月/3个月/半年 (若保留3个月及以上, 请说明理由)

查询

☐ 查询QPS  $\leq xx$

☐ 查询性能要求 P95  $\leq xx$  Sec

☐ 单次查询结果集最大行数  $\leq xx$  行 (若大于10w行, 请说明理由)

☐ 单次查询最大扫描分区数:  $<30$ 个 /  $<60$ 个 /  $<90$ 个 /  $>90$ 个 (若大于60个, 请说明理由)

特殊feature支持

☐ 是否需要精准去重, 非精准去重误差1%左右是否接受: 是/否, 接受/不接受

☐ 数据导入 (实时导入) 是否需要精准一次, 是否接受偶发的数据重复: 是/否, 接受/不接受

☐ 查询是否涉及join: 是/否

☐ 表字段是否涉及复杂结构? (Map/JSON/Struct等): 是/否

☐ 是否需要unique模型: 是/否

查询Pattern

(重要)

请尽量全面的列举常用的查询语法 (若暂时不确定查询pattern, 可用文字描述查询场景):

查询SQL Pattern 1: select xxx from

查询SQL Pattern 2: select xxx from

## 需求评估

- 快速理解业务需求, 判断Doris是否最适合业务场景

## 准入评审

- 参加大数据部门的需求准入评审
- 评估业务价值、投入产出比ROI



Point 4 - 使用规范

类型	关注1	关注2	关注3
建表	<p>分桶数建议值16或32，单个tablet 约1G</p> <p>反例:分桶设置太小，导致单个tablet达30G，执行compact很慢，集群吞吐变差</p>	<p>1. 表模型(优先使用Aggregate/Duplicate)</p> <p>2. 前缀索引(根据查询条件设置)</p> <p>3. 分区字段 (设置合理的生命周期)</p>	<p>高频写入表建议放在单独的数据库，避免事务数过多影响到同库的其他表</p>
flink写入	<p>切勿使用自己的jar，要使用flinksql</p> <p>反例：使用自己的jar，一条数据一个写入事务，导致集群吞吐变慢</p>	<p>数据无乱序(保证只有单个分区的数据)</p> <p>反例: 数据存在大量乱序，任务每次写入都涉及几十个分区，导致集群吞吐变慢</p>	<p>单个batch的发送数据量建议 Batch size &gt;= 100MB或者timeout = 2~ 5min，单表批次导入实例并发 &lt;= 2</p>
insert写入	<p>insert into执行成功后，还需等待版本发布后，数据才可见</p> <p>反例:执行成功后立即查询，查询结果为空</p>	<p>N/A</p>	<p>N/A</p>
删除	<p>避免频发删除，导致Compaction压力大，影响写入和查询性能</p> <p>反例:业务高峰期频繁执行delete语句，集群base compact频繁，吞吐变慢</p>	<p>Delete执行，数据可见性是异步</p>	<p>Drop语句后不用加force，这样出现误删除可在一段时间内找回</p>
修改	<p>一张表同时间只能执行一个alter</p> <p>反例: 执行修改列类型,会对数据做遍历，耗时久，这段时间再次执行则会报错</p>	<p>支持增删列、修改列，不支持修改列名</p>	<p>新增/删除数据前停写入任务，且等待修改状态finish后再写入，防止数据不一致</p>
查询	<p>严禁不带过滤条件查询全量数据，且建议业务加上limit兜底</p> <p>反例: 查询全表数据，打爆集群CPU和内存</p>	<p>Doris不适用高并发QPS场景</p>	<p>Doris兼容Mysql协议，业务查询时可带traceId，方便排查问题</p>



# 4 稳定性挑战应对实践

# 稳定性挑战

OLAP 引擎需要接入多个核心业务，是大数据的核心基础组件。业务对其稳定性要求高。

## 数据质量问题

画像不准确，用户洞察失真，**决策误导**；  
派券超发，**造成资损**

## 导数慢

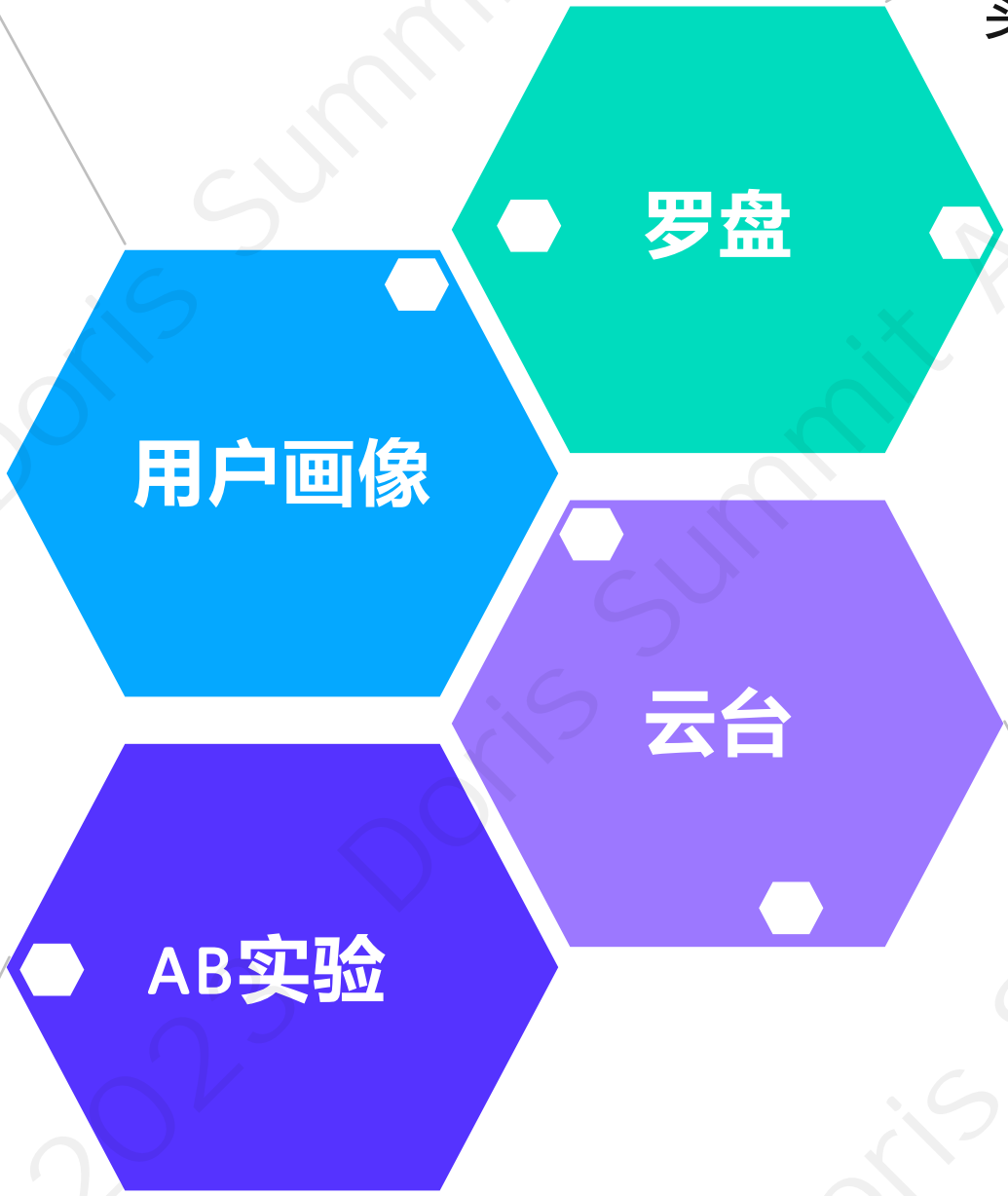
实时分析受阻，**诊断延误**，策略执行滞后

## 服务不可用

数据分析受阻，决策延误，**业务运营受影响**

## 查询问题

查询慢，用户体验差，满意度下降  
高并发大查询**打爆BE集群**



# 稳定性保障实践 - 数据质量问题

## 场景描述

业务使用SparkLoad导入Unique模型表，  
查询结果不稳定

## 原因

Unique模型表使用Sparkload导数时存在  
异常

## 规避办法

- 提前在Hive侧去重
- 将Unique模型改为Duplicate模型重建表

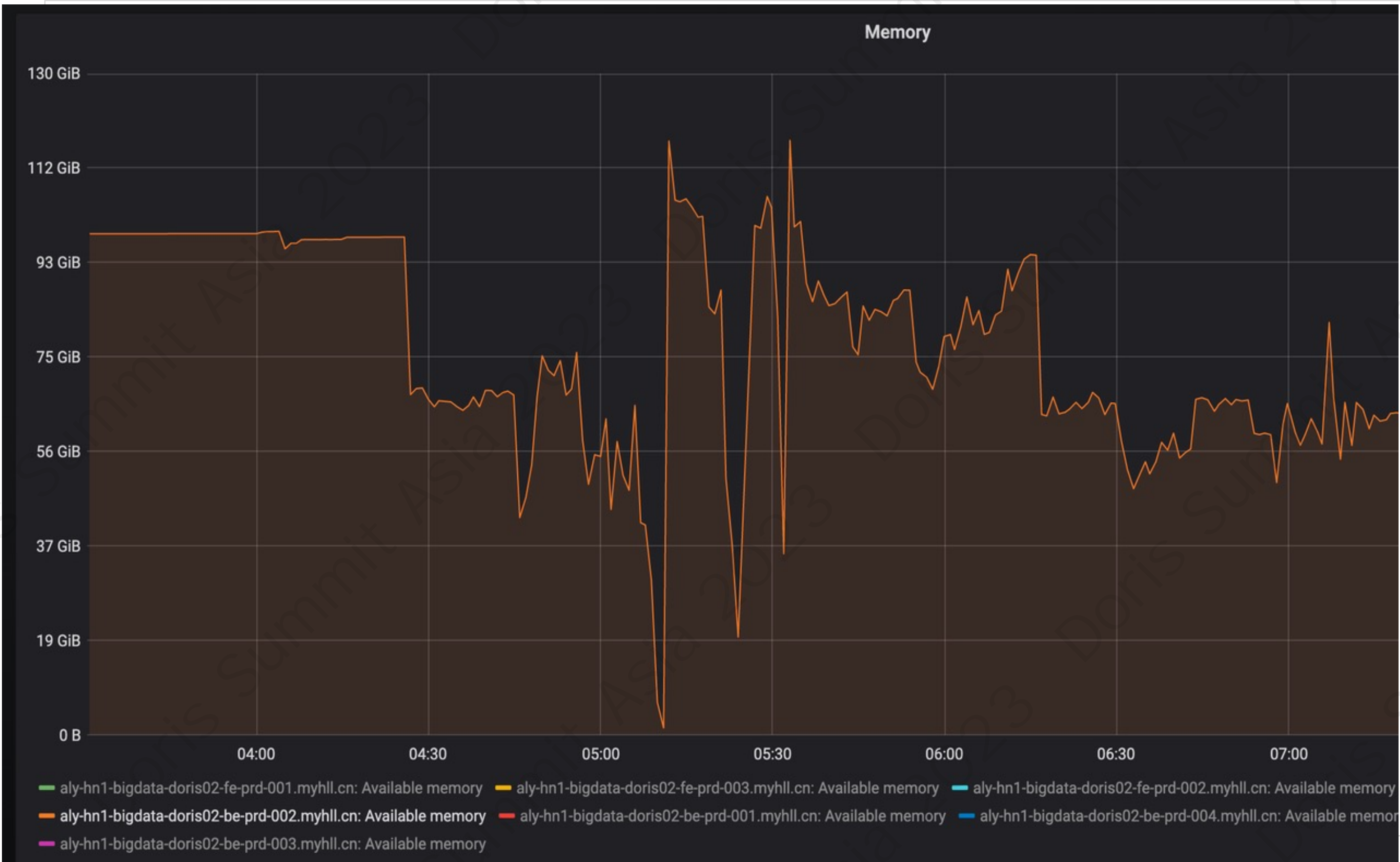


第一次查询

第二次查询



# 稳定性保障实践 - 版本升级问题



## 场景描述

凌晨时间段 broker load任务和insert任务重合时间段，BE内存出现OOM被kill导致任务报错

## 原因

升级1.2版本后的bitmap向量化读没有进行谓词下推，导致内存上涨

## 解决办法

1. 业务对SQL谓词下推的优化，如and和or的条件合并
2. 合入社区相应patch
3. 后续集群HA方案（因1.2无法直接回退1.1）

# 稳定性保障实践 - 业务变更问题

Results

Run successfully

SHOW PROC '/dbs/13032/1060207/partitions/9136516/9299229/9299550';

Execution Time: 1 ms

ReplicaId	BackendId	Version	VersionHash	LstSuccessVersion	LstSuccessVersionHash	LstFailedVersion	LstFailedVersionHash
9299551		350	1021618335750951880	350	1021618335750951880	351	0
9299552		350	1021618335750951880	350	1021618335750951880	351	0
9299553		350	1021618335750951880	350	1021618335750951880	351	0

## 场景描述

业务侧自行对Doris表进行新增字段,表数据未更新且在无法查询

## 原因

触发Doris版本1.0的bug，导致部分segment损坏，无法修复（该问题在 之后的版本已修复）

## 解决办法

- 1. 沉淀通过Sparkload快速恢复数据预案
- 2. 宣导用户使用规范、任务上线规范、发布变更规范



# 稳定性保障实践 - 查询性能问题

## 场景描述

场景:云台查询Doris间歇性报错(Thread pool is at capacity)

## 原因

用户提交大量查询以及一些大查询，导致fragment的rpc处理线程池满

## 解决办法

- 1. 加大查询缓存容量，增加缓存命中率
- 2. 查询超时由5min调整至3min
- 3. 增强大查询拦截能力





# 稳定性保障实践 - 导数性能问题

## 场景描述

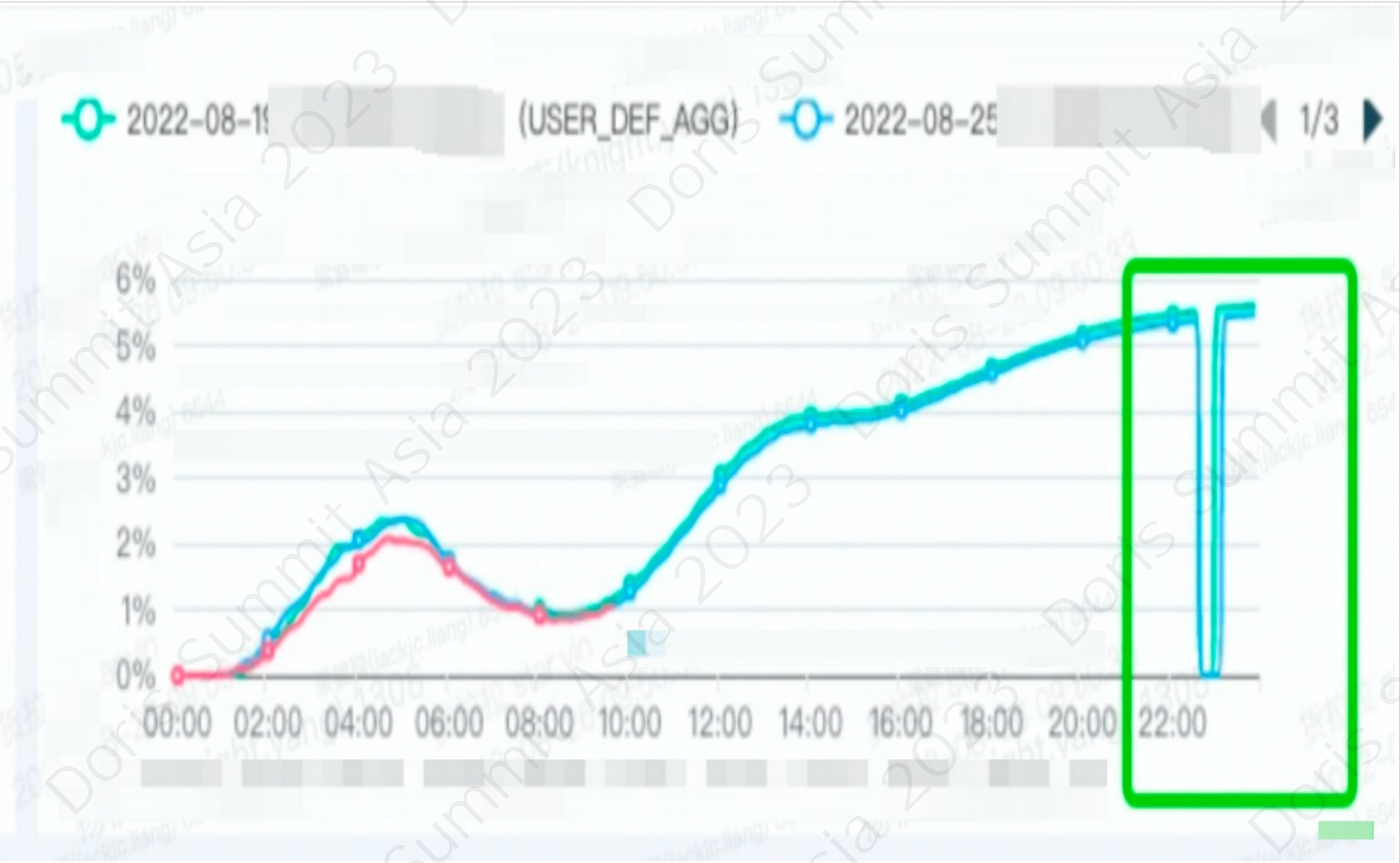
准实时场景下5分钟调度任务因多个任务执行超时，导致报表数据更新延迟并跌0

## 原因

新增的其他任务存在严重乱序，集群整体写入吞吐变慢，影响了准实时场景

## 解决办法

- 1. Doris任务及导入参数优化  
number\_tablet\_writer\_threads (16 -> 32)
- 2. 加强Doris变更规范管控与审批流程
- 3. 业务多租户隔离(进行中)



# 稳定性建设思路

少出事

- 稳定性案例: 业务变更问题、数据质量问题
- 稳定性能力: 容量规划、自动化能力、查询拦截能力、业务隔离、用户权限管控

快发现

- 稳定性案例: 导数问题、查询问题
- 稳定性能力: 发现能力

快恢复

- 稳定性案例: 导数问题、查询问题、版本升级问题
- 稳定性能力: 故障快恢复能力

# 容量规划

## 容量梳理

- 业务需求：如实时或者离线、数据写入速度与时延、表模型与分区方式、查询要求、存储要求、是否有特殊feature支持
- 数据量： $(\text{业务总数据量} / (1 - \text{数据量预留比例}40\%)) * \text{数据副本}3$
- 硬件资源：FE的配置1:4，如4C16G；BE的配置1:4，如32C128G
- 集群规模：参考业务需求、数据量情况来确认最终的集群规模

## 容量监控

- 机器指标：磁盘容量、CPU、内存
- 服务内部指标：连接数、核心线程池
- 导数及查询指标：任务数、延迟、状态
- 表级监控指标：表容量、tablet数量和大小

## 容量预警

- 高危/严重告警事件
- 关注业务需求
- 高峰期、拉货节保障
- 监控容量异动



# 自动化运维能力

## 背景

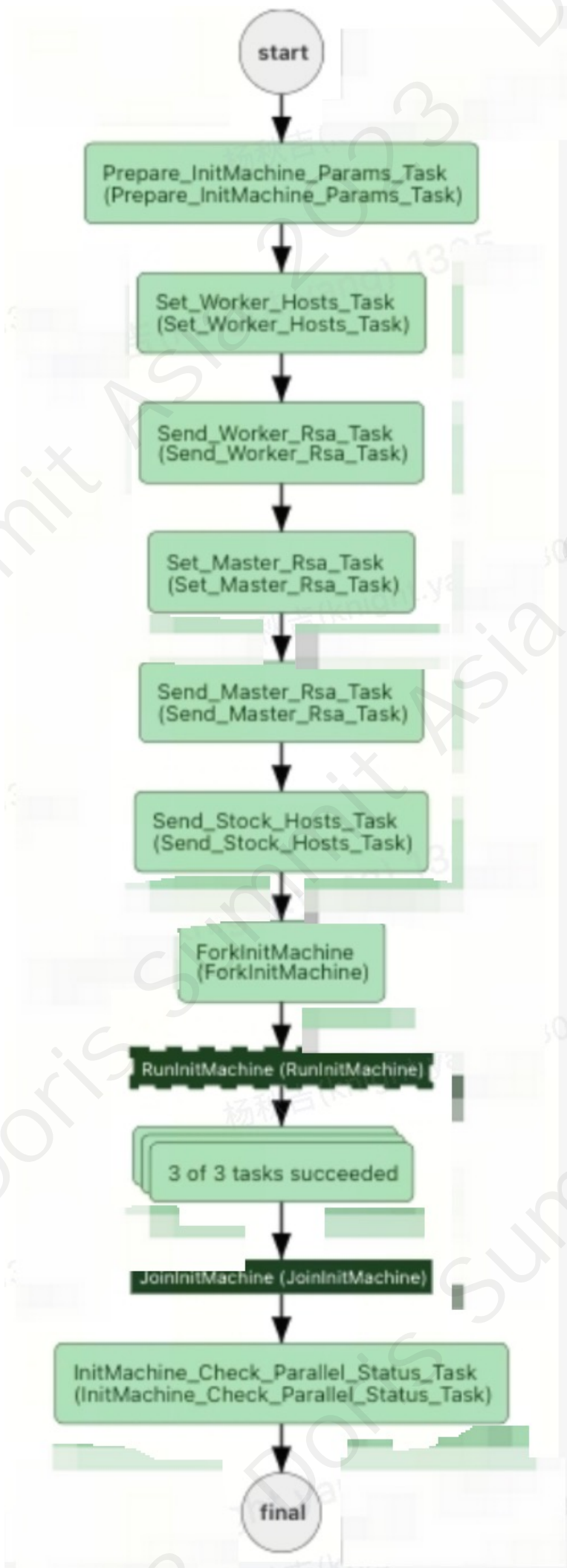
初期在构建大数据Doris集群时，我们以标准SOP指引下通过脚本手动操作为主，人为误操作或遗漏的可能，稳定性相对较差。

## 进展

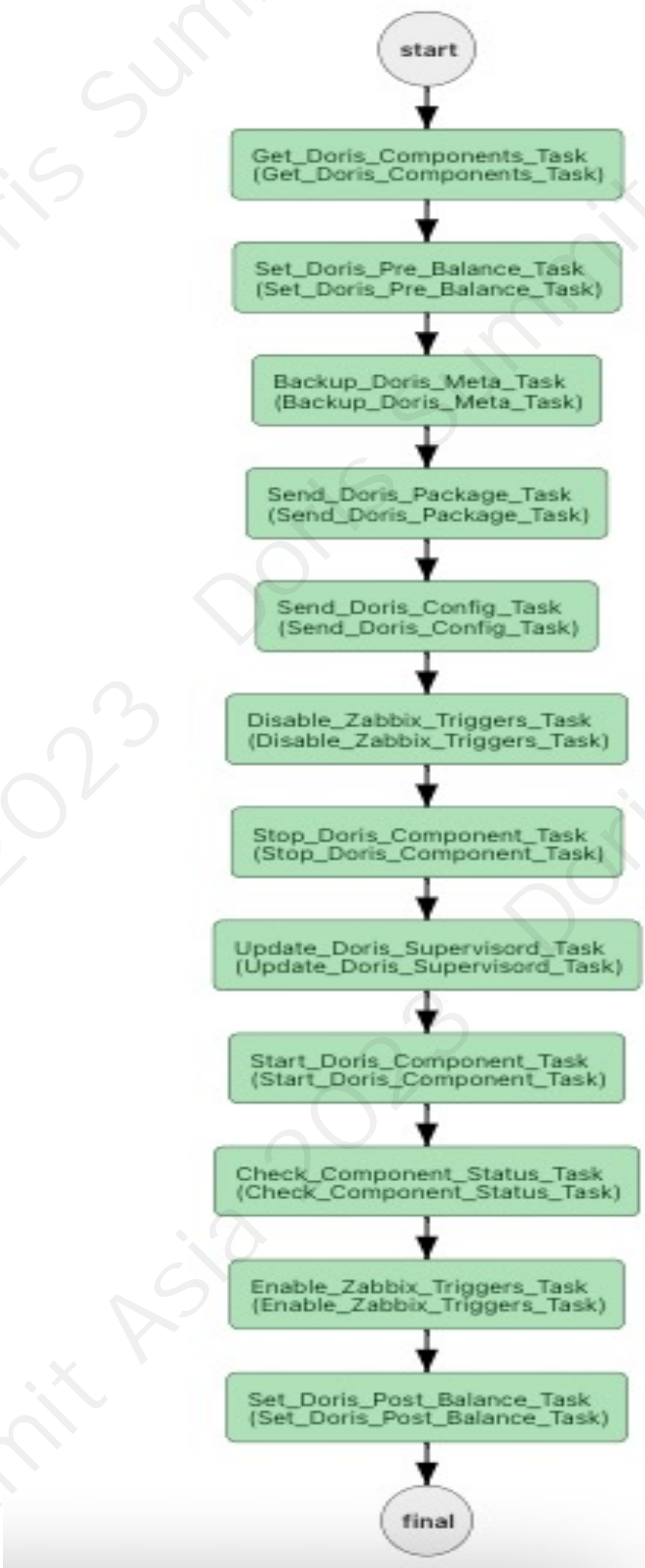
通过大数据自动化平台构建Doris自动化能力，底座基于Netflix Conductor、Ansible开发，已集成Doris部署、Doris扩容、Doris升级等工作流编排能力。

## 收益

- 1. 提升Doris组件服务稳定性
- 2. 提升运维人效

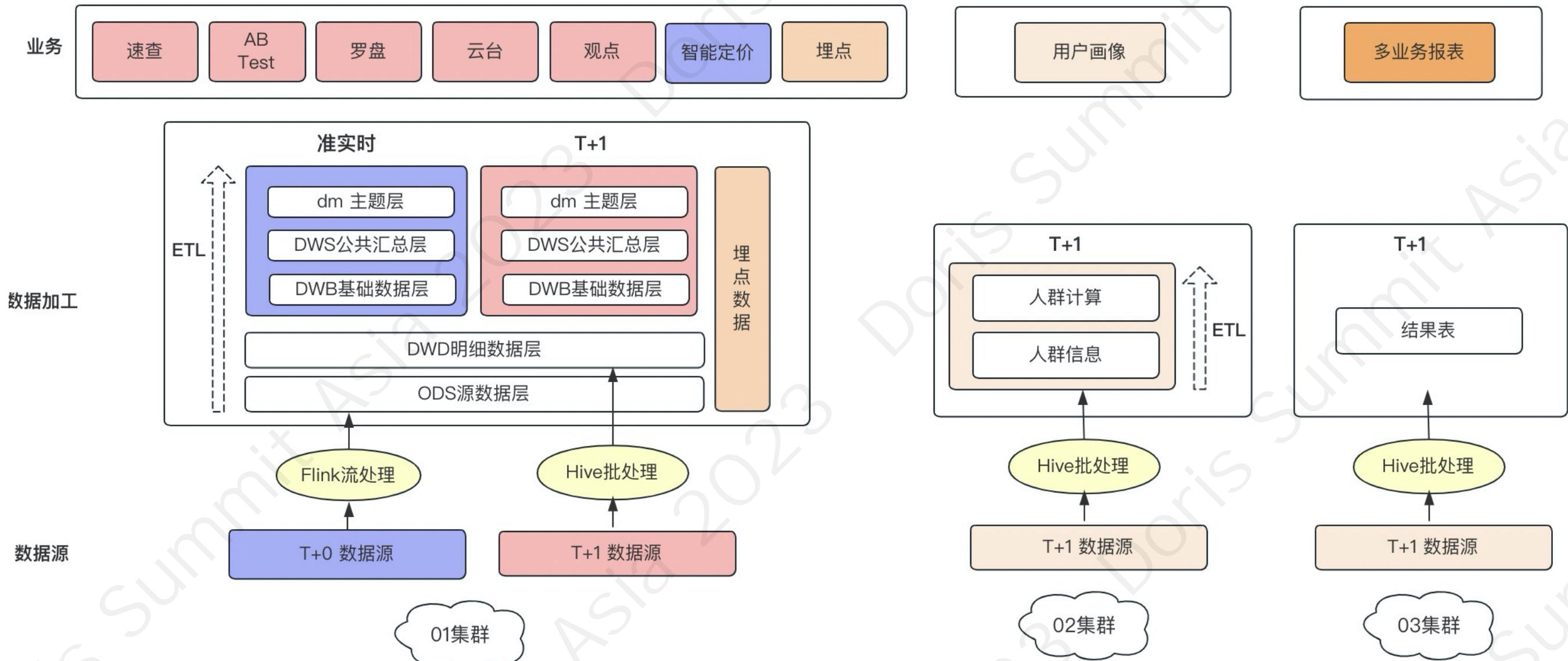


Doris 扩容 workflow



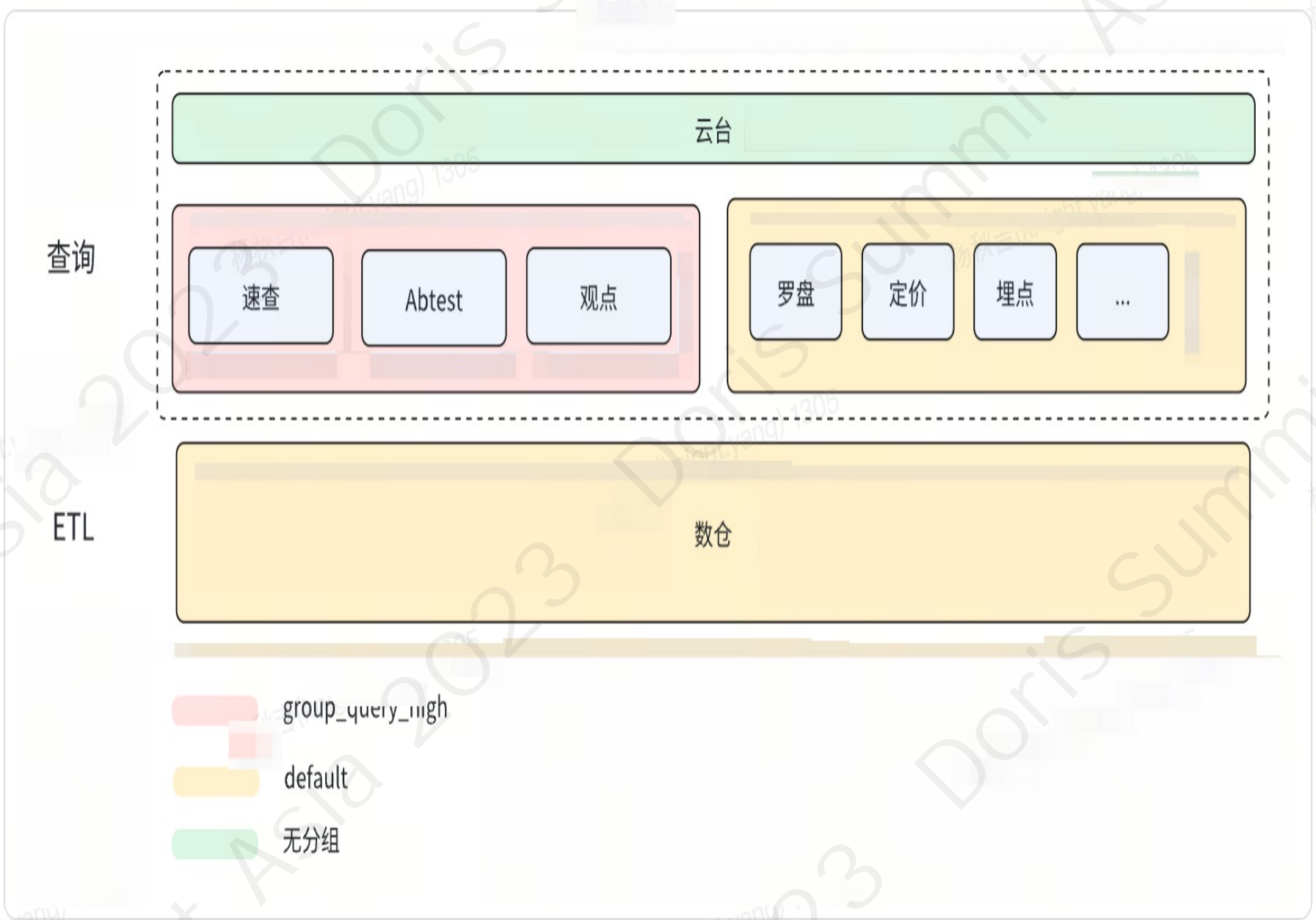
Doris 升级 workflow

业务隔离



集群间物理隔离

特征：高SLA、数据复用少、相互影响大



集群内多租户隔离

1. Doris自带多租户能力，支持存储和查询级别隔离
2. 按业务优先级划分
3. 优先保障查询可用性



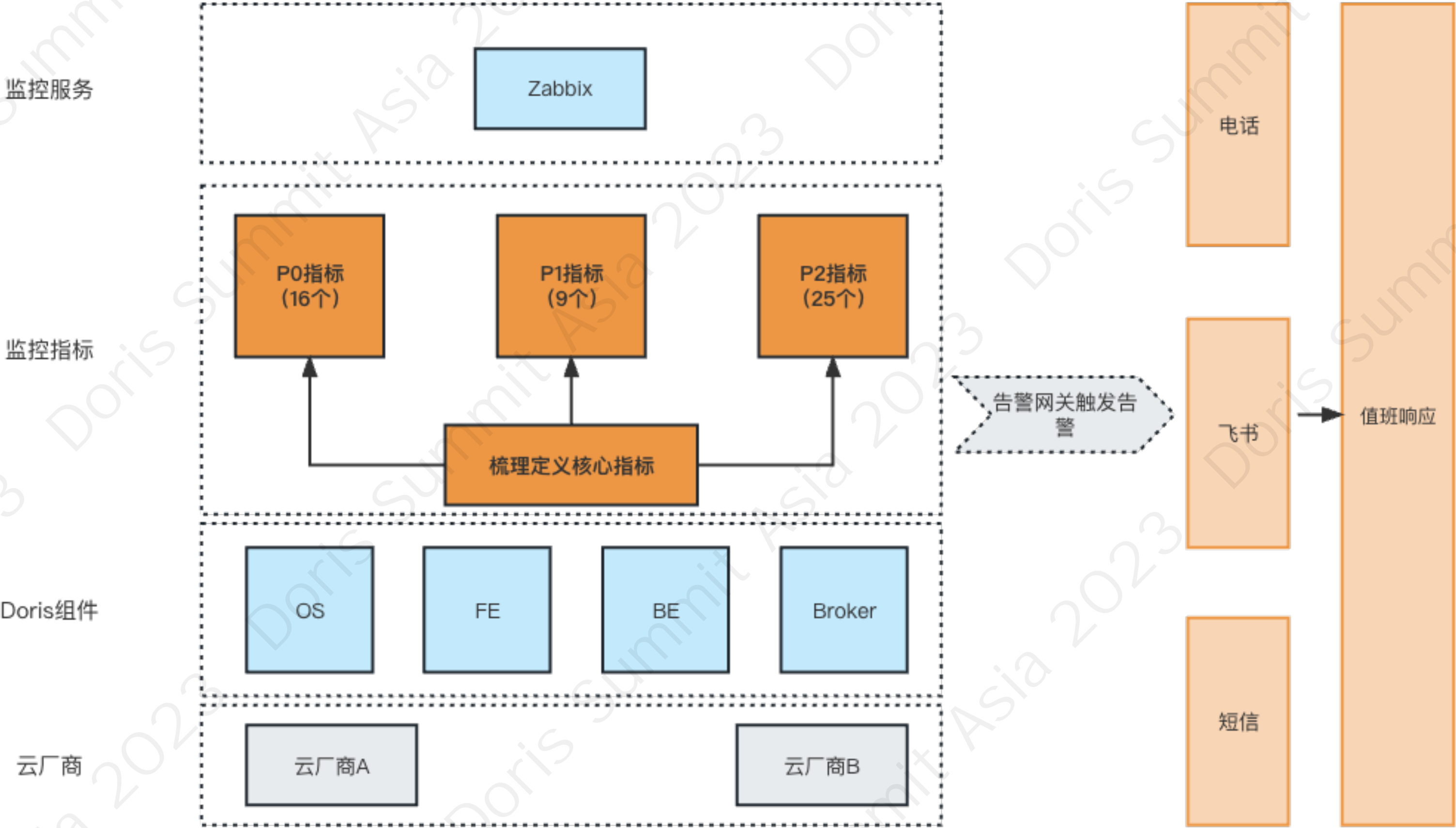
# 发现能力

## 目标

快发现：核心链路问题（主动发现）时间 <= 5min

## Doris监控告警系统

以Zabbix作为大数据基础架构组核心监控系统底座，对Doris服务进行监控和告警



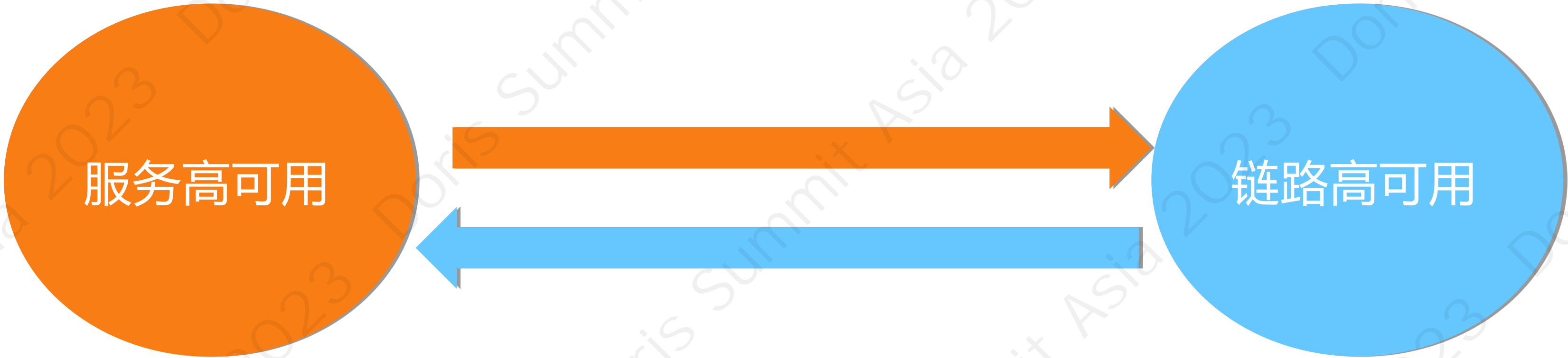


# 发现能力

- 1. 表级监控：监控表容量、状态
- 2. 任务监控：监控导数任务状态
- 3. 组件监控：服务指标(查询、导数)、进程、机器指标

指标分级	作用	告警级别	指标项
一级指标	发现和定位问题 (服务不可用)	Disaster	<ul style="list-style-type: none"><li>• 服务导数及查询(事务拒绝、事务失败、连接数、队列吞吐、RT、P99、P95)</li><li>• 服务内部状态(不健康tablet、进程、探活、JVM)</li><li>• 机器(内存、CPU、负载、磁盘、网络不可达)</li><li>• 任务状态(导数失败)</li></ul>
二级指标	定位问题	High	<ul style="list-style-type: none"><li>• 服务查询(QPS、查询请求)</li><li>• 服务内部状态(JVM)</li><li>• 机器(磁盘IO、网络读写)</li></ul>
三级指标	日常巡检、分析问题	Warnning	<ul style="list-style-type: none"><li>• 服务查询(表级监控)</li></ul>

# 高可用能力



- 1. FE：三台FE高可用部署
- 2. BE：数据三副本，四台及以上BE，避免一台宕机导致数据不可写
- 3. LB：使用负载均衡绑定三台FE，实现连接数均衡及读写高可用

- 1. 离线/准实时导数链路：  
Spark load/Broker load/Select insert into任务，通过离线调度任务平台进行调度，支持异常自动重试或者电话告警
- 2. 实时导数链路：  
Flink类型任务，通过自研实时任务平台进行调度，支持异常自动重试或者电话告警

# 其它保障能力



## 查询拦截能力

基于Doris查询黑名单功能配置规则  
异常SQL查杀能力



## 用户权限管控

基于Doris的RBAC  
和公司UAC打通（进行中）



## 故障快恢复能力

分区数据的快速恢复能力  
Tablet状态恢复能力



## 双集群HA能力（进行中）

业务侧双写



# 5 未来展望

## 未来展望

在过去的两年时间里，Doris已经在货拉拉内部得到广泛应用，支持了大数据内部的众多业务，逐步下线了Druid，形成一套以Doris为核心的数据生态。同时，在使用过程中也遇到了导入、查询等相关问题，这也推动我们深入了解Doris。

我们计划：

1. 支撑更多业务场景，跟随社区稳定版本升级迭代；
2. 上线存算分离能力，使计算、存储节点弹性；
3. 上线集群HA能力，解决跨版本升级不可回退的问题；
4. 深入参与社区，为社区贡献力量。

非常感谢社区的大力支持，同时我们也会密切关注社区，为社区贡献力量。



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](https://doris.apache.org)
- Apache Doris GitHub: [github.com/apache/doris/](https://github.com/apache/doris/)

## 获取更多峰会资料:

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