

# Apache Doris 在哪吒港航智慧科技数据智能中的应用与实践

邓宇超

上港集团哪吒科技 数据平台负责人



# 目录

1. 哪吒科技业务介绍

2. 哪吒数据智能架构演进

3. 实践场景

4. 收益与展望

# 01 哪吒港航智慧科技业务介绍

# 1-1 公司业务介绍

哪吒科技作为港航供应链数字化转型全解决方案供应商，致力于**智慧港口、智慧航运、智慧供应链**等领域数字化转型，产品落地众多海内外重要港口，全球范围内合作码头超 60 家，在船舶周转、库场管理等港航业务供应链持续提效。

The image is a composite of four panels illustrating the company's business process. The first panel shows a person writing on a document with the text '咨询服务' (Consulting Services) and '顶层设计 细节设计' (Top-level design, Detailed design). The second panel shows a person working on server cables with the text '项目实施' (Project Implementation) and '项目管理 软件部署 硬件集成' (Project management, Software deployment, Hardware integration). The third panel shows a server room with the text '运维服务' (Maintenance Services) and '运维中心' (Operations Center). The fourth panel shows a dashboard with the text '成熟产品' (Mature Products) and 'SMART系列' (SMART Series). Below the panels are two labels: '全生命周期' (Full Life Cycle) and '全解决方案' (Full Solution).

工班	工班	工班	工班
300	1	0	5
倒作业价值 (unit)	南工干线 (艘)	西工支线 (艘)	作业干线 (艘)
工班 小时	工班 小时	工班 小时	工班 小时
17.6	24.2	13.9	3562
18.9	26.1	17	1275
吊吊周转率 (天)	千点桥吊台时量 (unit/小时)	支墩桥吊台时量 (unit/小时)	装卸总量 (teu)

工班	工班	工班	工班
308	194	409	255
集卡进场车次 小时峰值	外集卡进场车次 小时峰值	外集卡作业车次 小时峰值	外集卡作业车次 小时峰值

# 1-2 公司核心产品



## 乾坤圈S 物流

通用仓库

重箱堆场

冷链仓库

空箱堆场

金属期货仓库

查验堆场

集团化仓库

智能堆场

理货信息系统



## 风火轮M 平台

车队预约

理货管控

港口防疫

危险品平台

船舶调度

安防监控



## 金砖A 研发

智能运维

智慧引航



## 火尖枪R 智能

数字孪生

虚拟现实

智能理货

智能道口

远控

F5G超远控

增强现实



## 混天绫T 码头

C-TOS

B-TOS

R-TOS

I-TOS

M-TOS

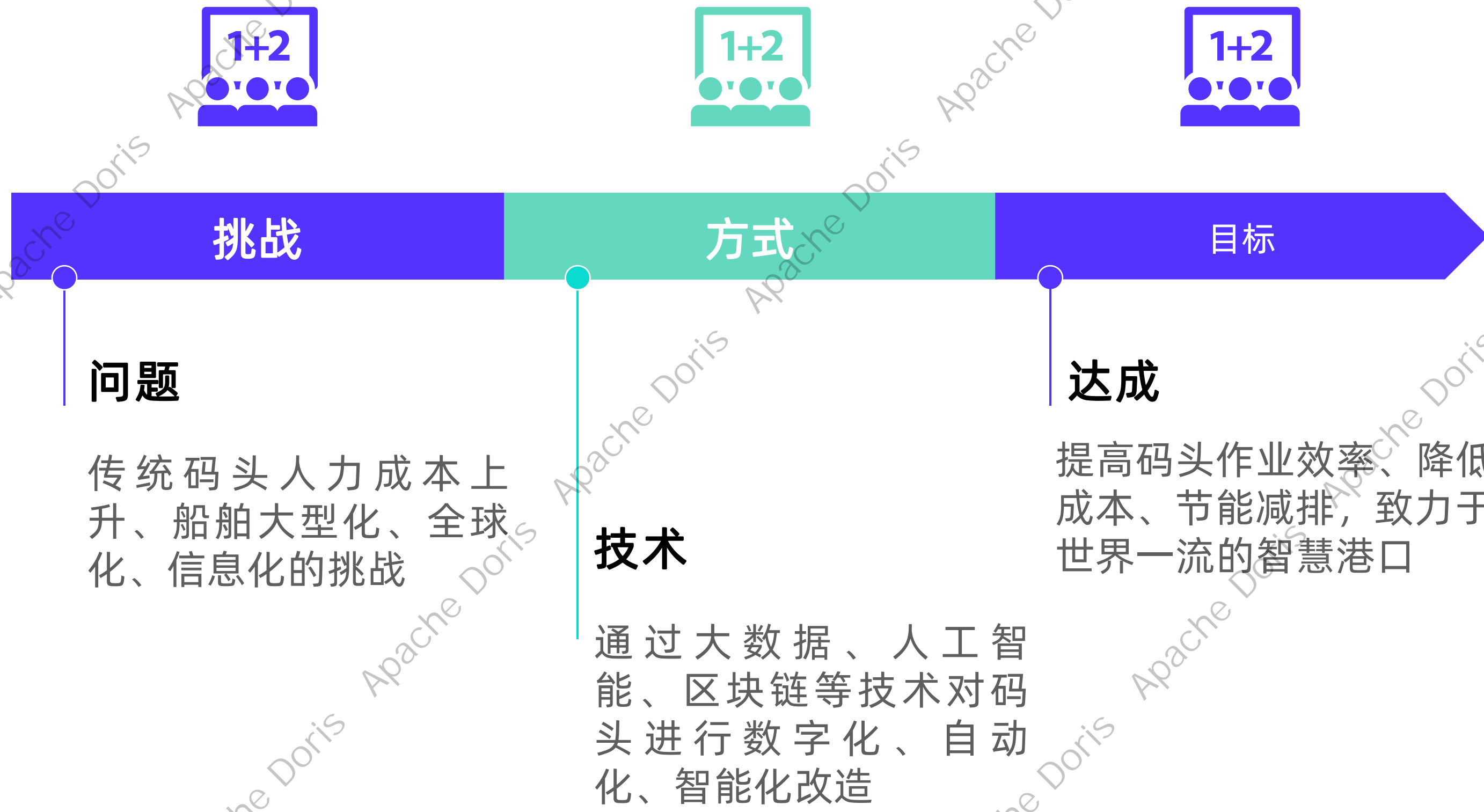
G-TOS

# 1-3 产品服务/智慧港口全解决方案

金融贸易便利化	支付网关	单一窗口	GSBN	电子商务	金融保险					
海运物流协同化	统一线上受理	电子装箱单	电子EIR	电子D/O	集卡预约					
	集装箱货信息服务				在线客服					
港域调度智能化	船舶调度	集拼	中转箱	环境监测	危险品监管	安防监控	交通管理	冷链监管		
	数据治理/数据中心/数据标准化中台									
码头生产数字化	应用	智能理货	远控/超远控	智能道口	ITOS/CTOS/BTOS/GTOS/RTOS	物流	车队	仓储	代理	理货
	基建	服务器	无线终端	网络通讯	智能机械	IoT设备				

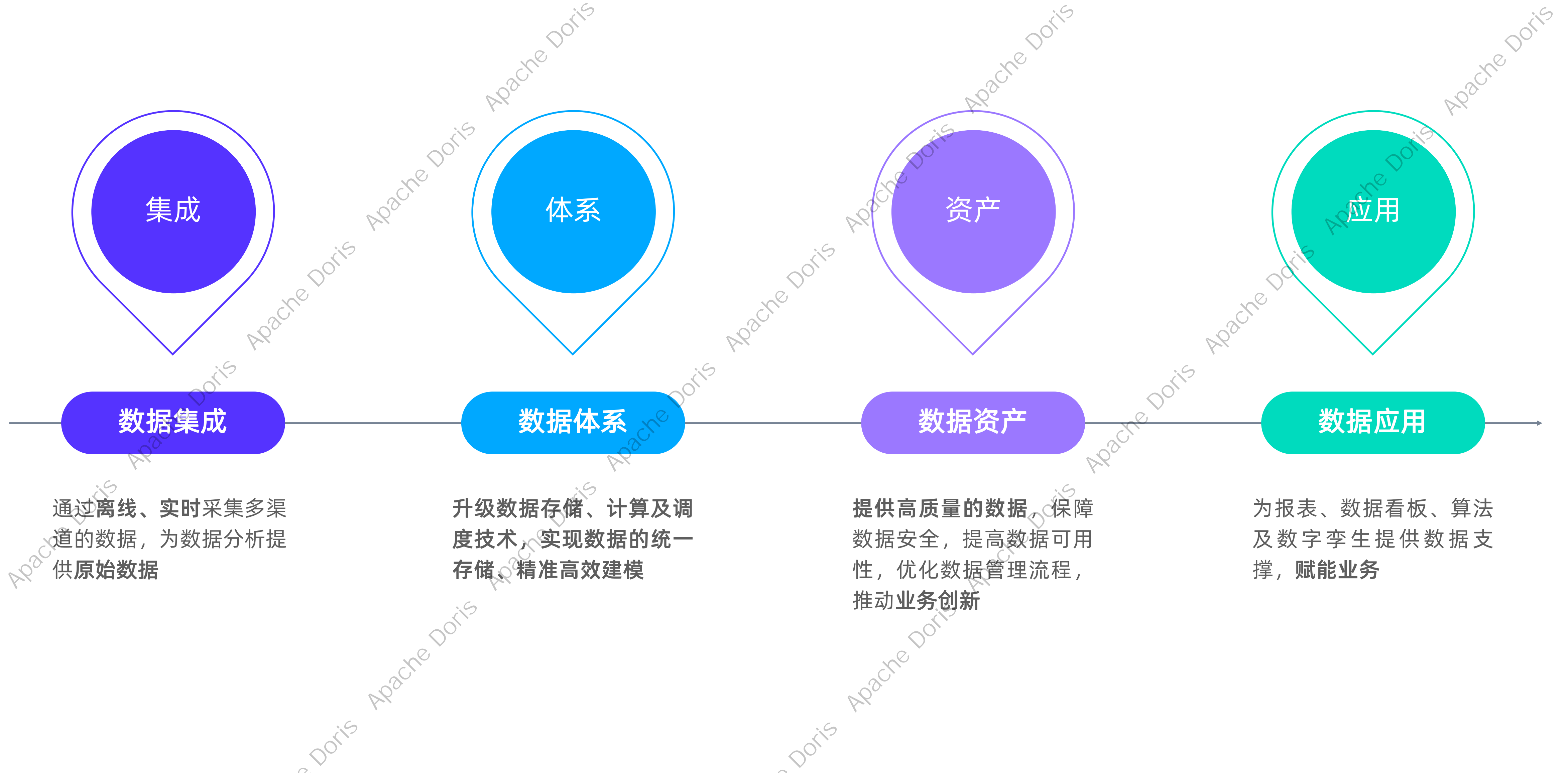
## 02 哪吒数据智能架构演进

# 2-1 数据智能项目背景

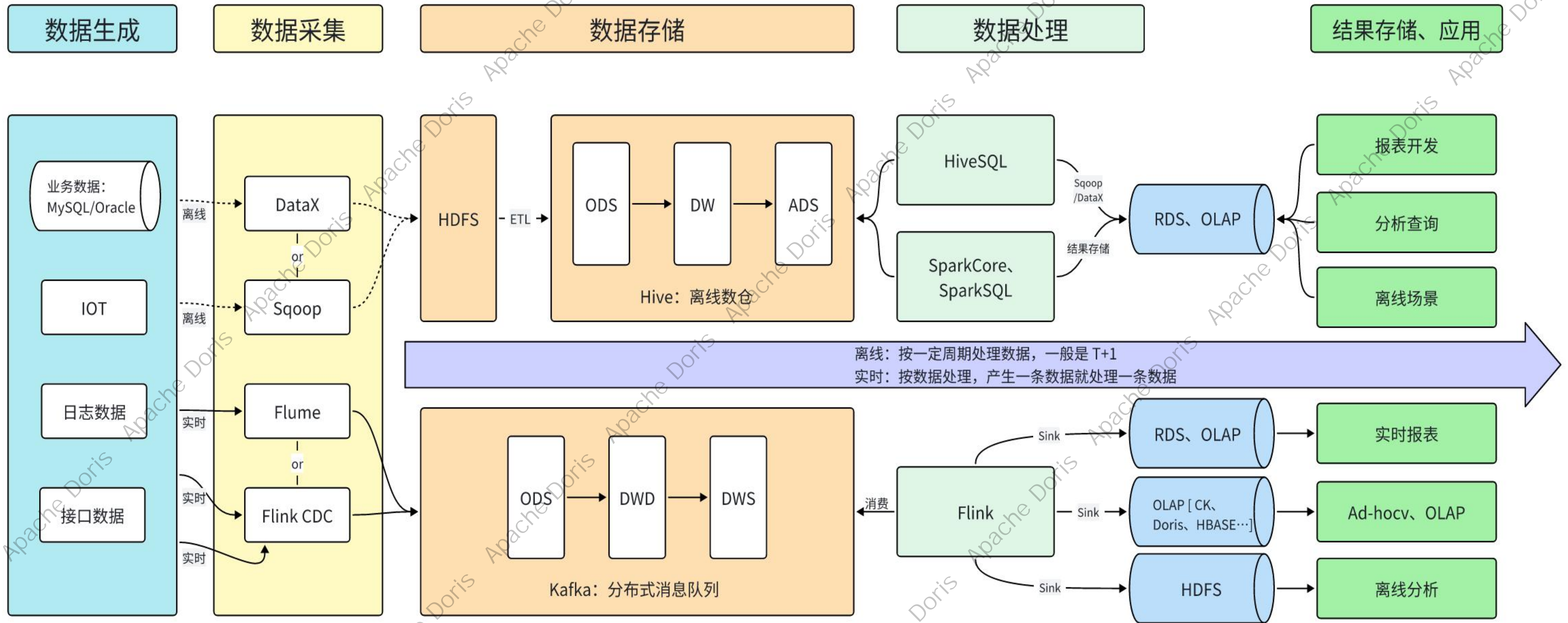




## 2-2 解决方案



# 2-3 数据智能架构 1.0



## 2-4 1.0 架构痛点与需求

### 架构痛点

- Hadoop 体系架构繁重，运维成本高
- 数据时效性低，不能满足数据近实时/实时分析需求
- 离线任务调度周期较长，任务数据回溯成本较高
- 研发学习成本高

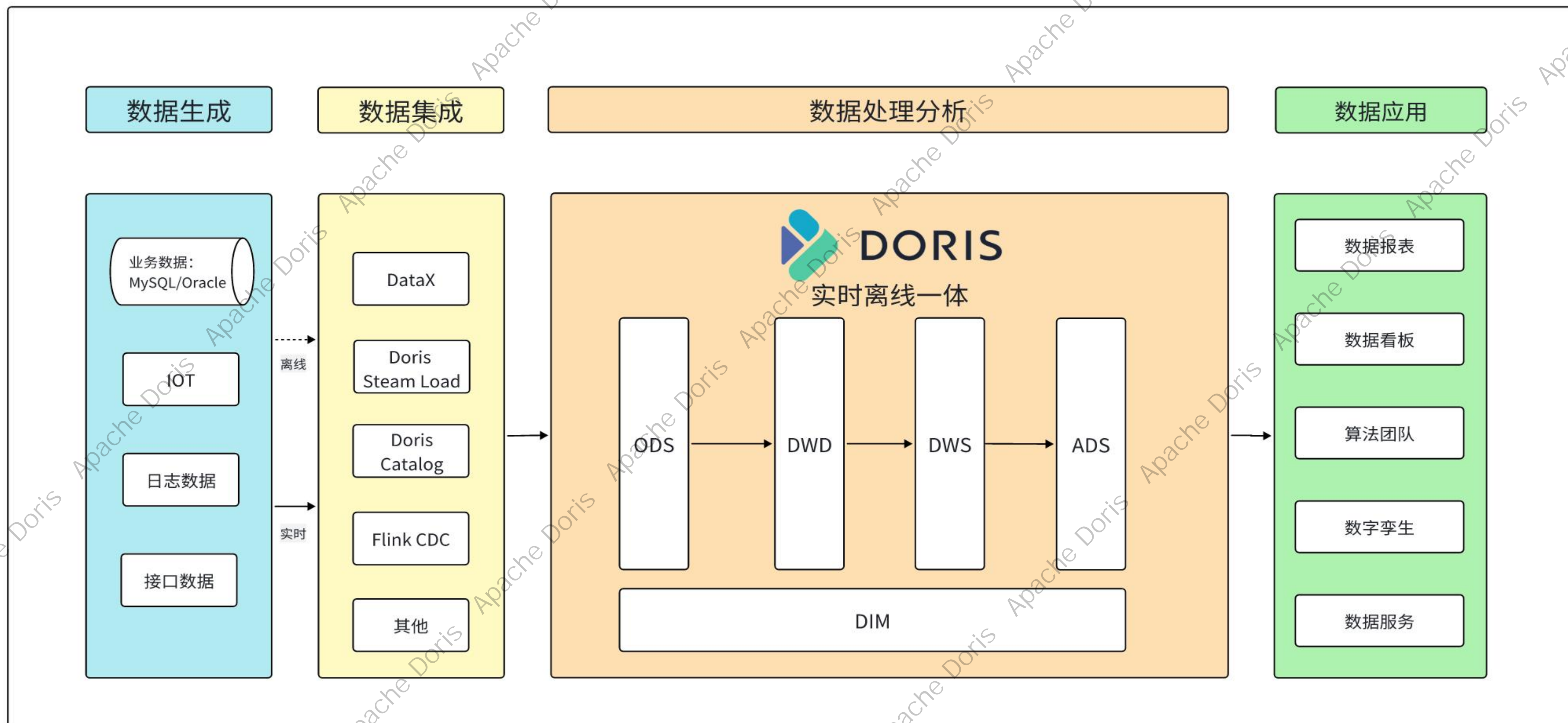
### 目标需求

- 架构简单，低运维成本
- 实时高效，支持实时数据采集、分析、查询
- 提升离线任务执行效率，降低数据回溯成本
- 简化开发流程，降低研发学习成本

## 2-5 选型因素

关键指标	Apache Doris
性能	在数据处理和查询上，可以达到10 亿级高并发点查毫秒级响应、亿级多表关联查询秒级响应，查询性能大幅提升
数据导入方式	支持 Stream Load、Bocker Load、Routine Load 等多种数据导入方式，满足不同业务需求场景
运维部署	不依赖第三方组件，FE/ BE扩缩容简单，支持数据自动平衡，数据迁移简单
成本	兼容 MySQL 协议，入手 Doris 时的学习成本相对较低，对开发同事十分友好，客观上降低了人力成本

## 2-6 数据智能架构 2.0



## 2-7 集群规模

当前总集群数

10+

总节点数

100+

平均每日查询  
总数据量

1亿+

最大集群  
存储数据总量

100TB+

# 03 实践场景

# 3-1 数据导入场景



## 实时导入

- 业务库 Binlog
- 服务日志等

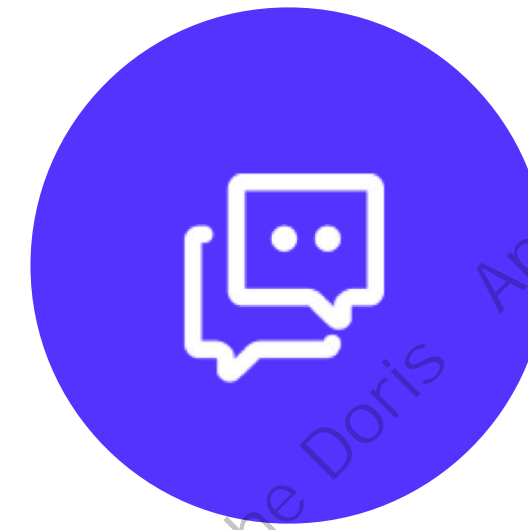
Flink Doris Connector  
sink.enable.batch-mode = true  
借助 Uniq 模型做到幂等，不依赖于 Flink ck  
性能 TPS **100w+**



## 离线导入

- 船舶作业数据
- 码头作业数据等

JDBC Catalog  
多表多并发导入  
性能达到 TPS **500w+**



## HTTP 写入

- 后端接口数据

Stream Load  
通过 HTTP 协议写入数据  
多线程写入性能 TPS **30w+**



## 3-2 数据导入：问题and经验分享（一）

### 问题现象

问题：Doris-StreamLoad json 格式数据写入部分数据丢失。

现象：后端调用 StreamLoad 接口，写入 json 数据，到所有字段为 key 列的 unique 的表中，发现 StreamLoad 任务成功，且上游日志显示写入成功，doris 表中部分数据缺失。

如：同一个 taskid，有 3 次 put 请求，执行 3 次 StreamLoad 事务成功，返回 success 信息，但表中仅 1 条数据

### 原因

vertical compaction 分组的逻辑在只有 key 列，没有 value 列的时候有问题，会导致 key column group 的最后一列不是 delete sign。现在的代码逻辑，认为 key group 的最后一列是 delete sign，这导致了 vertical compaction 的时候，错误地删除了数据。

### 解决方案

方案1：调整表模型，避免使用全字段为 key 列，从而避免版本 bug。

方案2：版本升级至 Doris-2.0.8。

## 3-3 数据导入：问题and经验分享（二）

### 问题现象

问题：通过 JDBC Catalog 导入 doris 数据，oracle-catalog 创建 success，select 查不到表

现象：

# 出现以下报错：找不到表

```
select * from oracle-catalog.dhname.bas_checkpoint limit 10;
```

### 原因

部署 Doris 时，设置了大小写不敏感，即 FE 参数的 `lower_case_table_names` 设置为 1 或 2 时，导致在查询 doris 上游数据表时，select 表名全部被转化为小写，而 doris 同步的上游元数据，为大写表名，造成无法查到上游 oracle 原始大写表的情况。

### 解决方案

创建 catalog 时，加入参数：`"lower_case_table_names" = "true"`

3. 当 FE 参数的 `lower_case_table_names` 设置为 1 或 2 时，JDBC Catalog 的 `lower_case_table_names` 参数必须设置为 `true`。如果 FE 参数的 `lower_case_table_names` 设置为 0，则 JDBC Catalog 的参数可以为 `true` 或 `false`，默认为 `false`。这确保了 Doris 在处理内部和外部表配置时的一致性和可预测性。

在 Doris 2.0.3 之前的版本，仅对 Oracle 数据库有效，在查询时，会将所有的库名和表名转换为大写，再去查询 Oracle。

# 3-4 高可用测试：问题and经验分享

## 问题现象

问题：Flink 任务写入 Doris，间接性出现 tablet 异常，version 版本缺失，导致 tablet 无法正常 compaction。

现象：

```
last full failure time: null,
last cumulative success time: "2023-12-12 17:56:24.274",
last base success time: "2023-12-12 17:56:24.274",
last full success time: "1970-01-01 08:00:00.000",
rowsets:
[0-476] 2 DATA NONOVERLAPPING 020000000003af7102340d77eccb0524cdbe2e290762f058b 280.44 MB,
[477-286376] 1 DATA NONOVERLAPPING 020000000003af7112340d77eccb0524cdbe2e290762f058b 17.74 MB,
[286377-476484] 1 DATA NONOVERLAPPING 020000000009e5ae02340d77eccb0524cdbe2e290762f058b 2.61 MB,
[476485-476960] 1 DATA NONOVERLAPPING 0200000000000e2246a2dd84e0b4ba41678109b011cd80 458.30 KB,
[476961-476965] 0 DATA OVERLAPPING 020000000000009a2246a2dd84e0b4ba41678109b011cd80 0,
[476966-476966] 0 DATA OVERLAPPING 020000000000010d2246a2dd84e0b4ba41678109b011cd80 0,
[476967-476967] 0 DATA OVERLAPPING 02000000000001112246a2dd84e0b4ba41678109b011cd80 0,
[476968-476968] 0 DATA OVERLAPPING 020000000000014e2246a2dd84e0b4ba41678109b011cd80 0,
[476969-476969] 0 DATA OVERLAPPING 02000000000001cc2246a2dd84e0b4ba41678109b011cd80 0,
[476970-476970] 1 DATA NONOVERLAPPING 02000000000002072246a2dd84e0b4ba41678109b011cd80 73.59 KB,
[476971-476971] 0 DATA OVERLAPPING 02000000000002492246a2dd84e0b4ba41678109b011cd80 0,
[476972-476972] 0 DATA OVERLAPPING 020000000000026d2246a2dd84e0b4ba41678109b011cd80 0,
[476973-476973] 0 DATA OVERLAPPING 020000000000028a2246a2dd84e0b4ba41678109b011cd80 0,
[476974-476974] 0 DATA OVERLAPPING 02000000000002c82246a2dd84e0b4ba41678109b011cd80 0,
[476975-476975] 0 DATA OVERLAPPING 02000000000003152246a2dd84e0b4ba41678109b011cd80 0,
[476976-476976] 0 DATA OVERLAPPING 02000000000003382246a2dd84e0b4ba41678109b011cd80 0,
[476977-476977] 1 DATA NONOVERLAPPING 020000000000034f2246a2dd84e0b4ba41678109b011cd80 76.50 KB,
[476978-476978] 1 DATA NONOVERLAPPING 020000000000036c2246a2dd84e0b4ba41678109b011cd80 76.50 KB,
[476979-476979] 0 DATA OVERLAPPING 02000000000003902246a2dd84e0b4ba41678109b011cd80 0,
[476980-476980] 0 DATA OVERLAPPING 02000000000003ad2246a2dd84e0b4ba41678109b011cd80 0,
[476981-476981] 0 DATA OVERLAPPING 02000000000003d92246a2dd84e0b4ba41678109b011cd80 0,
[476982-476982] 0 DATA OVERLAPPING 020000000000043f2246a2dd84e0b4ba41678109b011cd80 0,
[476983-476983] 0 DATA OVERLAPPING 02000000000004902246a2dd84e0b4ba41678109b011cd80 0,
[476984-476984] 1 DATA NONOVERLAPPING 02000000000004e62246a2dd84e0b4ba41678109b011cd80 71.97 KB,
"missing_rowsets": [
  "476961-476963"
],
```

## 原因

BE 做高可用测试，导致集群有大量的 tablet 需要修复，副本修复和副本均衡任务在同时进行，导致副本节点直接做迁移和合并的时候，出现异常。

## 解决方案

Doris 已在 2.0.3 版本解决。

## 3-5 数据处理：问题and经验分享

### 问题现象

问题：Doris 函数 substr() 引起 BE 宕机，版本 Doris-2.0.5。

现象：

Doris-1.2.4 升级到 2.0.5 后执行相同 SQL，BE 发生宕机，并稳定复现。

### 原因

substr(VARCHAR content, INT start, INT length)

求子字符串，返回第一个参数描述的字符串中从 start 开始长度为 len 的部分字符串。首字母的下标为 1。

SQL 中使用了 substr 函数，当参数 3 的 value 值为负值，会触发 Doris-2.0.5 中的 bug，导致 BE 宕机。

SQL: `select substring('1aa',1,-1);`

### 解决方案

升级版本至 Doris-2.0.6: <https://github.com/apache/doris/pull/31316>

## 3-6-1 道口车辆实时监控场景

### 业务场景

对进出港口的车辆实时监控，记录车流信息，进出港口的车辆信息、车辆行为路径分析、货物目标溯源。

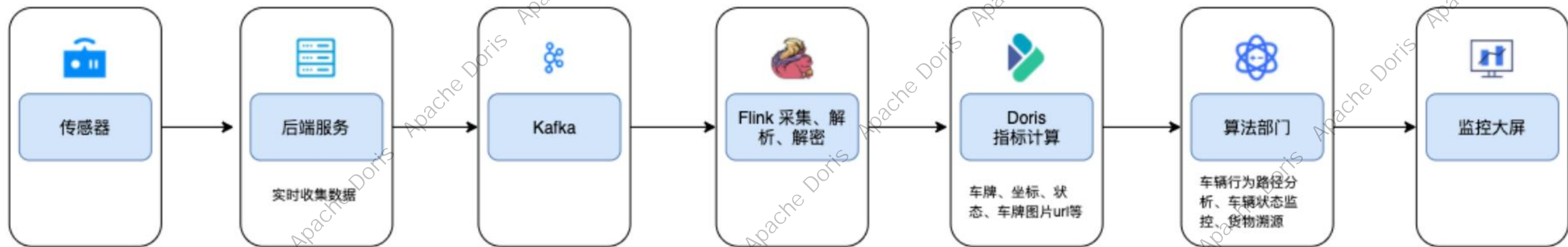
### 遇到的挑战

数据链路长，百亿级的数据量，多表关联分析，实时性要求高。

### 解决方案

采用 Flink 实时采集车辆数据，存储到 Doris 进行指标计算，针对大数据量慢 SQL 查询的问题，将磁盘 Raid5 替换成裸存储优化写入及查询性能。

## 3-6-2 道口车辆实时监控场景架构



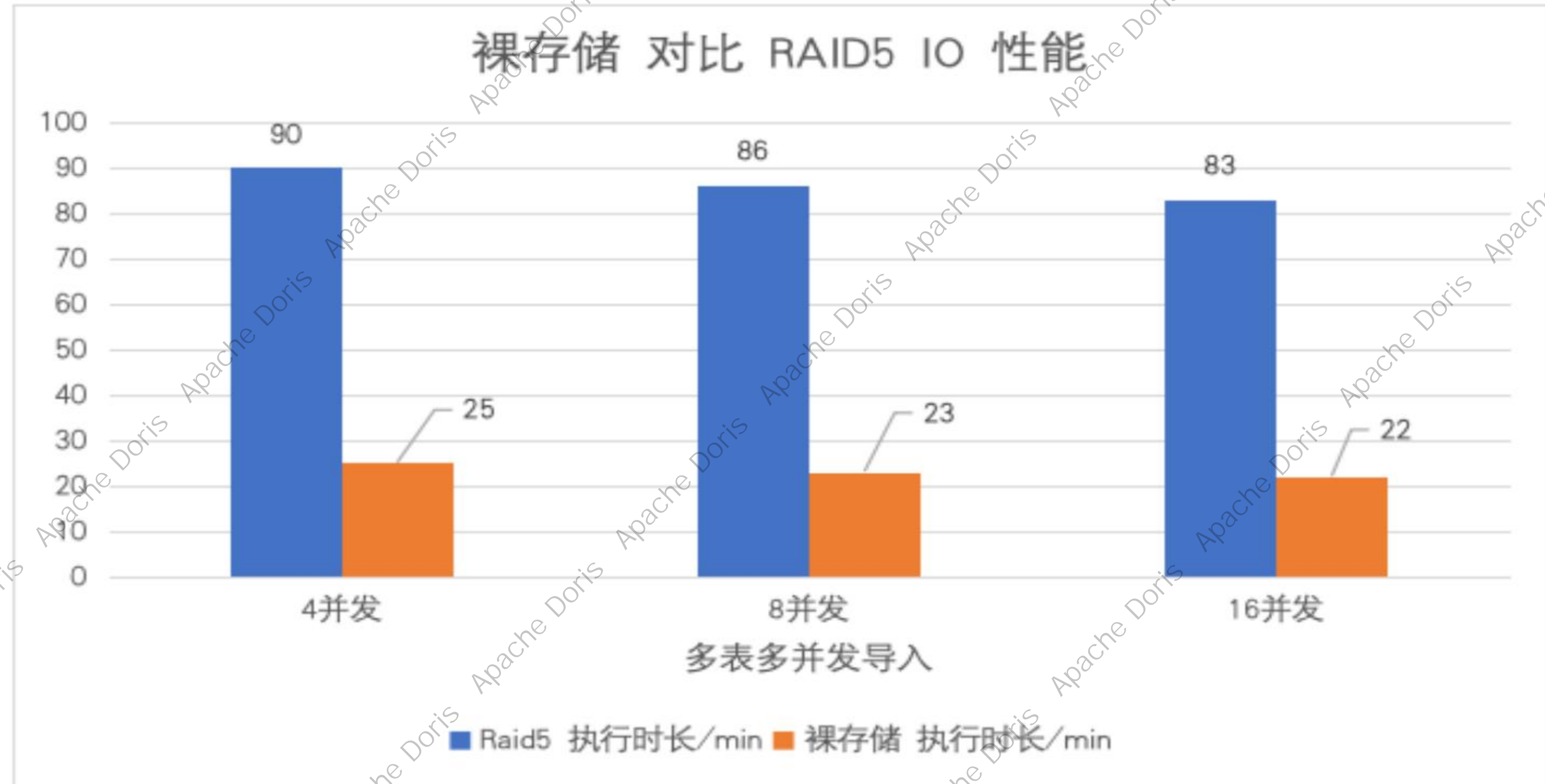
### 方案收益

经过压测裸存储相对于 Raid5, 磁盘 IO 有三倍的性能提升, 解决了大表 Join 性能及慢查询的问题, 查询延迟在秒级别, 满足车辆自动化实时监控需求。

## 3-6-3 DataX 导入 Doris 压测脚本示例

```
"job": {  
  "content": [  
    {  
      "reader": {  
        "name": "xxx",  
        "parameter": {  
          "column": ["xxx_fgid",  
                    "xxx_insertdt",  
                    "xxx_insertuser",  
                    "xxx_lstupddt",  
                    "xxx_lstupuser",  
                    "xxx_dngapply",  
                    "xxx_preapplytm",  
                    "xxx_dngremark"],  
          "splitPk": "xxx",  
          "connection": [  
            {  
              "jdbcUrl": ["jdbc:oracle:thin:@//****:****/xxx"],  
              "table": ["xxx"]  
            }  
          ],  
        },  
      },  
    },  
  ],  
  }.....
```

### 3-6-4 Doris 导入压测结果





## 3-6-5 Doris 查询压测脚本示例

```
select xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx,  
  xxx
```

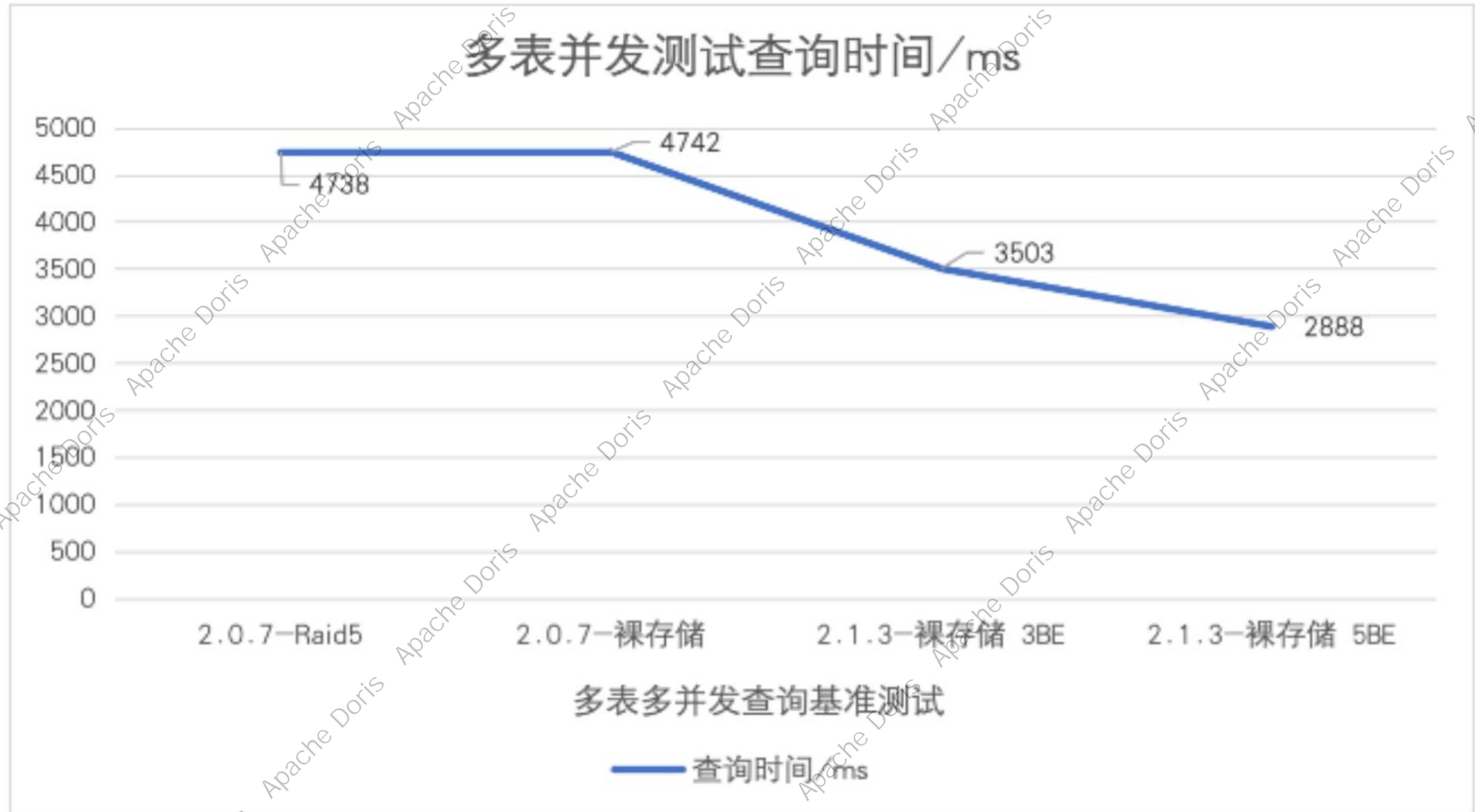
```
from xxx MOR
```

```
  inner join xxx IYC on MOR.xxx = IYC.xxx
```

```
  inner join [broadcast] xxx PCS
```

```
on IYC.xxx = PCS.xxx ...
```

# 3-6-6 Doris 查询压测结果



# 04 收益与展望

# 4-1 收益总结

## 性能提升

亿级多表关联查询秒级响应，查询性能大幅提升，满足低延迟查询需求，查询性能提升 80%

## 降低成本

采用 Doris 精简架构，降低了平台的运维成本、硬件成本、人员开发成本，降低因技术栈兼容性带来的风险

## 提升容错能力

数据写入支持事务，确保数据操作的原子性、一致性、隔离性和持久性，支持复杂的数据操作，提高了系统的容错能力，数据的高并发多用户访问，容错能力提升 50%。

## 推动业务创新

提高码头作业效率，优化码头运营，助推港航业务全链条整体数智化升级



## 4-2 展望

- 深入 Doris 在智慧港口中的应用，提供高效稳健的数据平台；
- 使用 Doris 在日志和时序场景中的运用；
- 利用 Doris 基于 Workload Group 的负载隔离能力。

# Thanks !

