

Apache Doris 在 Hadoop 生态圈中的角色定位

肖培根（Bacon）- 某家居行业公司 大数据主管

目录

01 自我介绍

02 现状分析

03 Doris平台建设

04 Doris平台应用

01

自我介绍

分享嘉宾



肖培根 (Bacon)

大数据主管

02

现状分析

中台价值

◆以公司私有云为基础，建设数据中台+系统前台的数据应用模式，实现数据统一、复用，加速创新数据应用落地

打破原有数据孤岛

- ①**数据拉通**：将所有数据汇聚到数据中台，应用系统从数据中台取数，打破数据孤岛
- ②**统一指标**：拉通各业务环节，明确指标owner，统一数据口径，提供统一指标服务，消除数据二义性
- ③**数据质量**：连通全域数据，对数据进行常态化、周期性的质量管控与治理

指标口径统一、数据质量提升

快速支撑业务应用

- ①**自助取数**：围绕分析、取数等场景，面向业务人员建设简单易用的自助分析、自助取数工具
- ②**自助消费**：建设低门槛数据自助消费工具集，实现简单数据使用场景自助服务

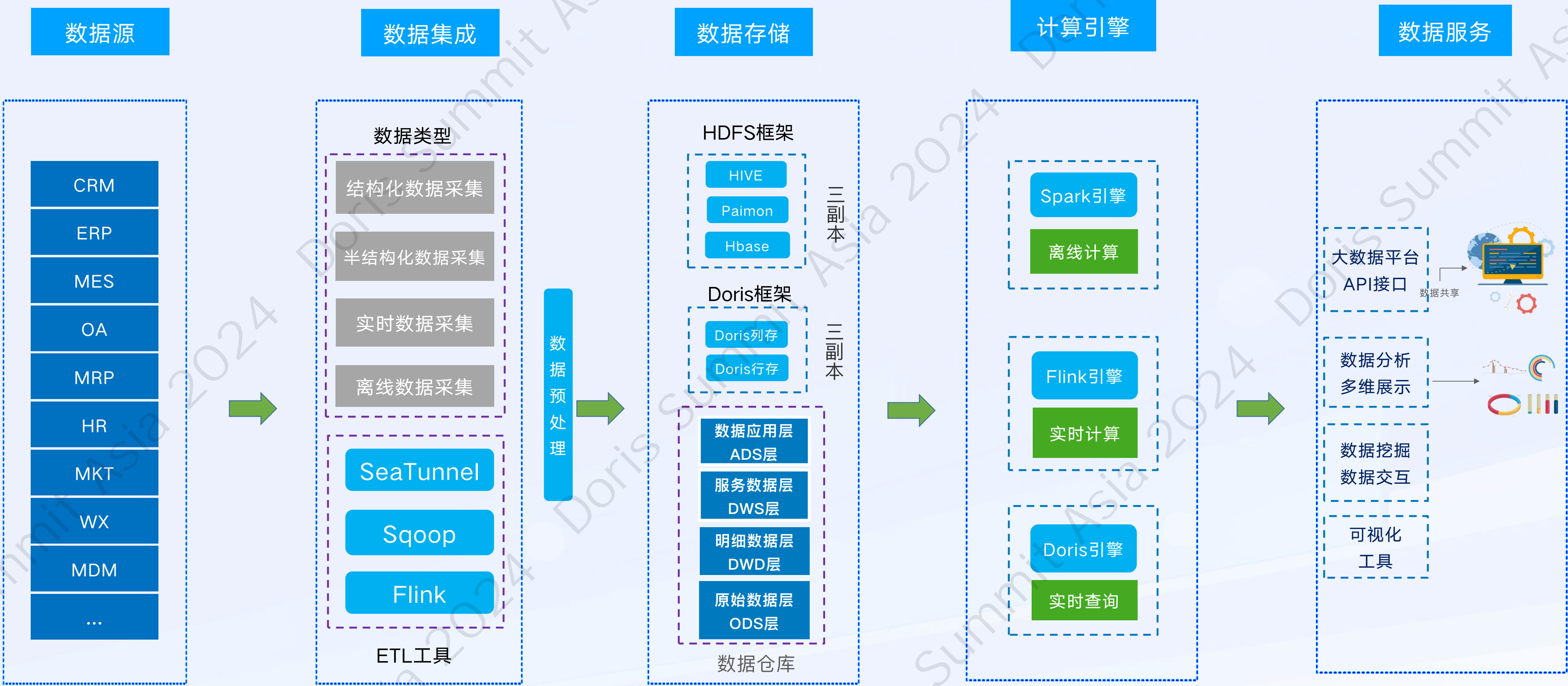
沉淀数据资产、高效取数服务

挖掘数据内在价值

- ①**数据服务**：提供便捷、快速的数据服务能力，支持数据资产场景化能力的快速输出
- ②**业务赋能**：根据业务场景形成可快速使用的数据模型，挖掘数据价值，为业务进行赋能

指标口径统一、数据质量提升

中台架构



03

Doris 平台建设

Doris 平台建设

资源规划

集群搭建

数据备份

集群升级

查询测试

文档编写

3.1-3.30

4.1-4.30

5.1-5.21

5.22-5.31

6.1-6.11

6.15-6.30

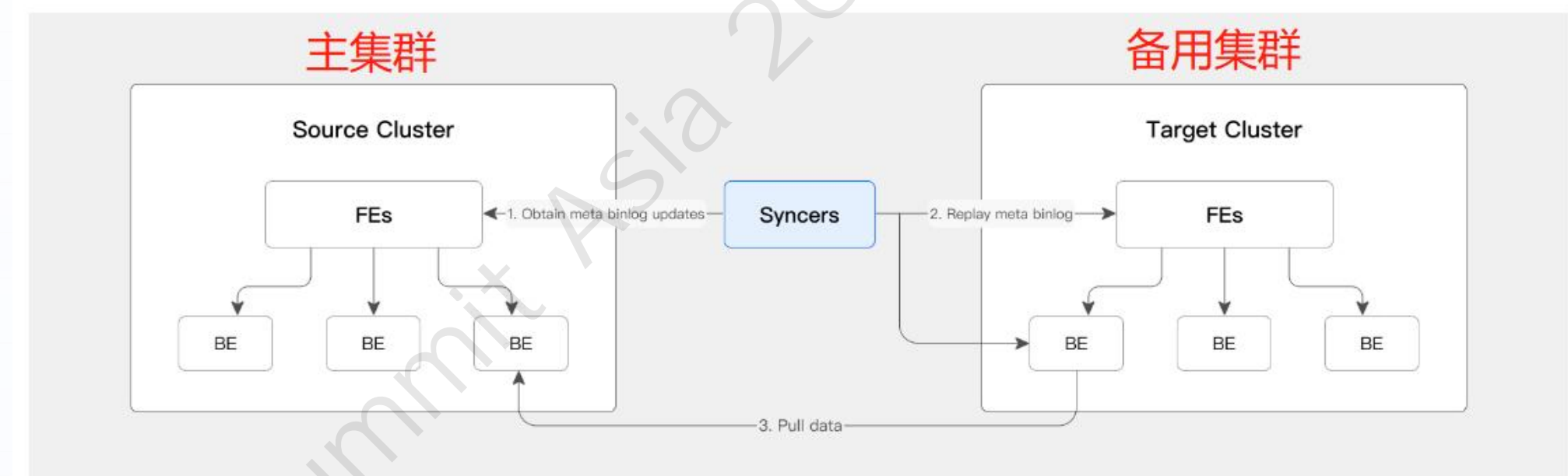
完成情况

- (1) 搭建 2 套 Doris 集群，采用 3 FE + 3 BE 组合；
- (2) Doris 资源隔离和权限控制；
- (3) 搭建 CCR 跨集群数据同步，主集群实时同步到副集群；
- (4) 编写备份脚本，定期备份主集群数据到 Hadoop；
- (5) 配置 Nginx 负载均衡和高可用；
- (6) 增加 FE/BE 故障告警到企业微信；

CCR跨级群数据同步

用于容灾备份、读写分离、集团与公司间数据传输和隔离升级等场景。

架构说明



CCR 工具主要依赖一个轻量级进程：Syncers。Syncers 会从源集群获取 binlog，直接将元数据应用于目标集群，通知目标集群从源集群拉取数据。从而实现全量和增量迁移。

Doris 平台资源

1.1_{TB}

内存

使用率50%

7.96_{TB}

硬盘

使用率30%

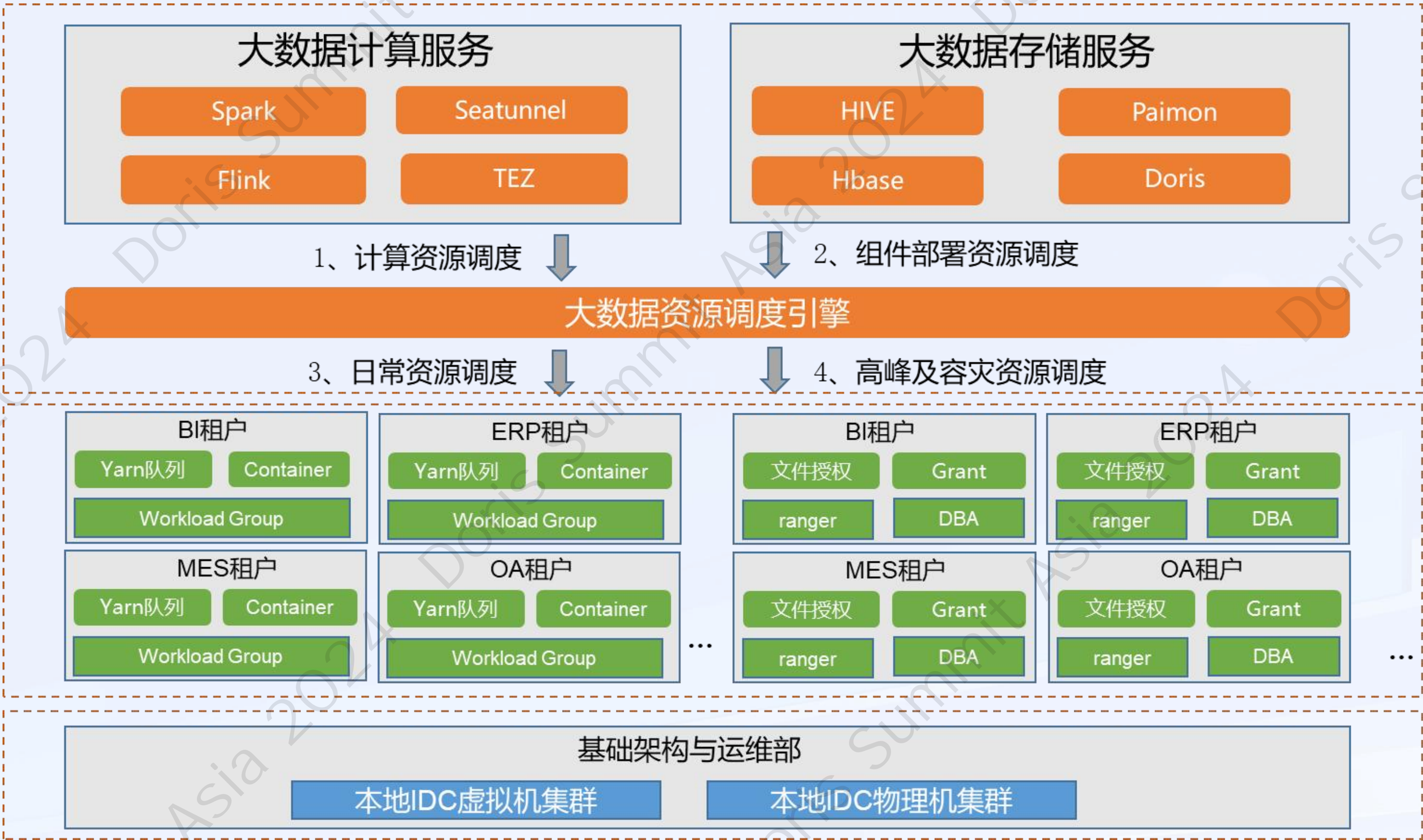
288_{Core(s)}

CPU

Intel(R) Xeon(R) Gold 6336Y CPU @
2.40GHz

平台资源隔离

- ◆ 建设目标：弹性大数据服务组件、容器化、存储与计算分离、资源弹性伸缩。
- ◆ Workload Group：负责授予给特定的角色和用户，管理 Doris 集群所使用的 CPU/内存/IO 资源用量。



04

Doris 平台应用

Apache Doris 的定位

◆ Apache Doris 的定位：满足报表分析、即席查询、统一数仓构建、湖仓一体等使用场景。

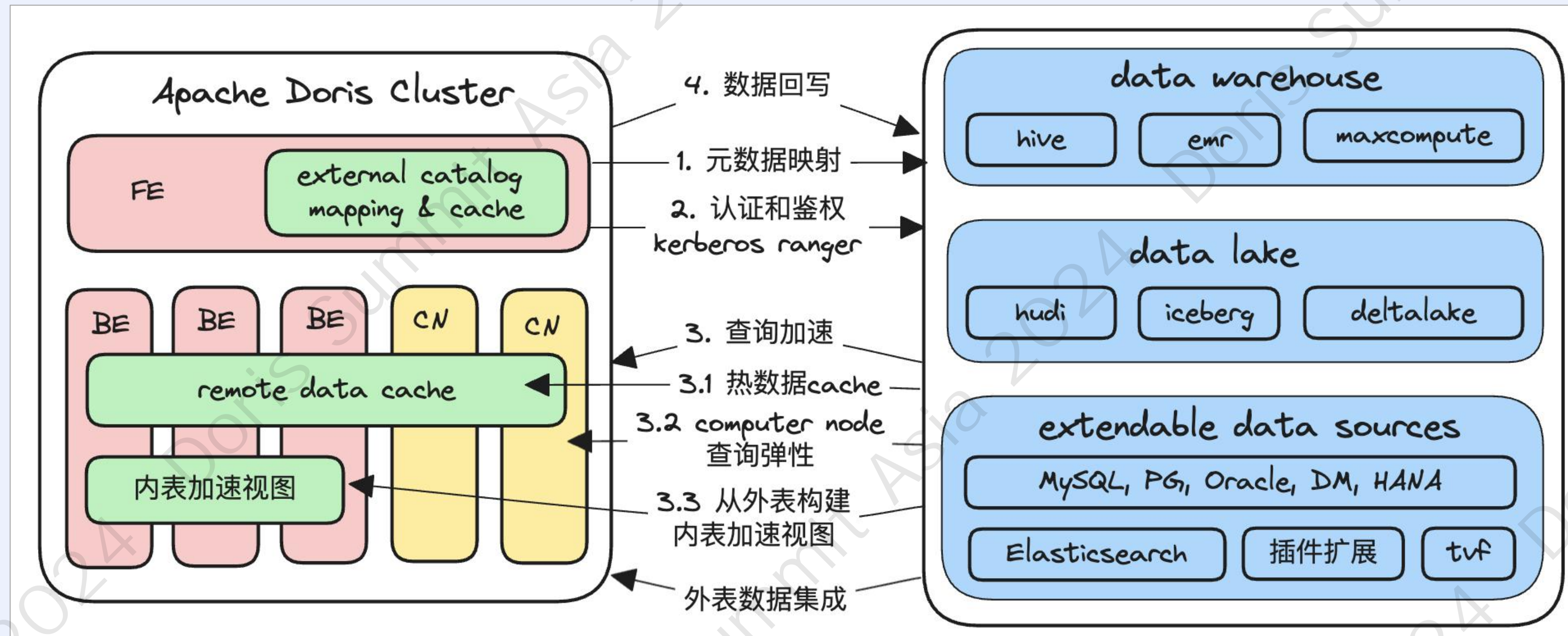
Doris 功能研究

- ◆ Doris 和 Paimon 结合
- ◆ Catalog 统一元数据
- ◆ 冷热数据分层
- ◆ Flink Doris Connector
- ◆ Spark Doris Connector
- ◆ BE OOM 分析和内存管理
- ◆ OPEN API 服务
- ◆ 资源管理
- ◆ 数据备份和迁移
- ◆ 湖仓一体研究
- ◆ SQL 优化



图：Doris 定位

湖仓一体实践



■ 便捷的元数据和数据打通

- 元数据映射、Cache 和 自动刷新
 - 支持大部分湖仓格式 和 meta store
 - 支持对接各类主流数据库
- 已测试有：MySQL、Oracle、SQL Serve、ES
Hive、Paimon、HANA

■ 分析加速

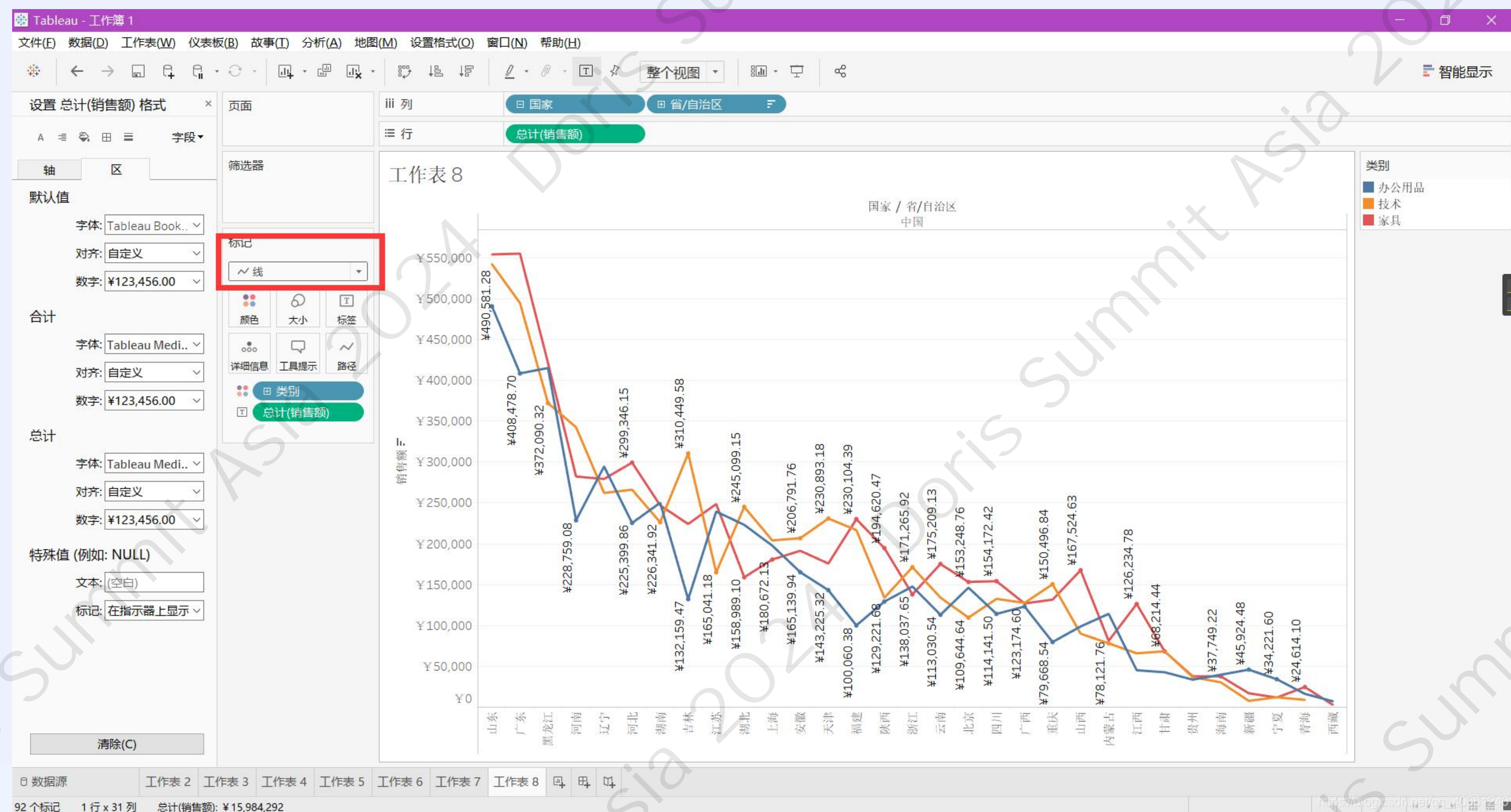
- 利用 Doris 高效的分析引擎加速
- 热数据 Cache 到本地
- 支持弹性计算节点，实现计算弹性加速
- 外表处理结果可写入内表，形成加速视图

■ 主要应用场景

- 湖仓查询加速
- 数据导入和集成
- 统一查询网关
- ETL / ELT 加速

Tableau 自助分析推广

- ◆Doris 数据提取：Catalog 方式同步数据，T+1 模式
- ◆Doris 库表：最大的表有 6 亿多记录，总数据超 12 亿
- ◆业务场景：用户流量分析、客户画像、物料统计分析等
- ◆实现价值：自助取数和自助消费，面向业务人员建设自助分析工具，实现简单数据使用场景自助服务



数据质量保障

- ◆任务告警
- ◆数据备份
- ◆资源隔离
- ◆资源监控
- ◆任务监控
- ◆参数优化
- ◆自动重启
- ◆故障转移
- ◆负载均衡

FlinkCDC 实时同步 Doris

SQLserver数据库端操作

step1: 数据库开启CDC功能

root登录服务器执行命令启用SqlServeragent后重启

数据库

/opt/mssql/bin/mssql-conf set

sqlagent.enabled true

step2: 选择要进行 CDC 跟踪的表

EXEC sys.sp_cdc_enable_table

@source_schema = 'dbo',

@source_name = 'orders',

@role_name = 'cdc_role';

step3: 检查开启了CDC的表

select name,is_tracked_by_cdc

from cdc_test.sys.tables where

is_tracked_by_cdc ='1'

	name	is_tracked_by_cdc
1	orders	1
2	C_Dealer	1
3	C_Dealer1	1

Flink端操作

1、Download flink-sql-connector-sqlserver-cdc-3.1.0.jar and put it under <FLINK_HOME>/lib/

2、启动flink client ./sql-client.sh embedded

SET execution.checkpointing.interval = 10s; 一定要设置否则无法同步

3、创建源表t_source_sqlserver, 使用SQL Server Change Data Capture (CDC)连接器从SQL Server数据库读取数据

```
CREATE TABLE t_source_sqlserver (  
  id INT,  
  order_date DATE,  
  purchaser INT,  
  quantity INT,  
  product_id INT,  
  PRIMARY KEY (id) NOT ENFORCED  
) WITH (
```

'connector' = 'sqlserver-cdc', -- 使用SQL Server CDC连接器

'hostname' = 'xx.xx.xx.xx', -- SQL Server主机名

'port' = '30027',-- SQL Server端口

'username' = '',-- SQL Server用户名

'password' = '', -- SQL Server密码

'database-name' = 'cdc_test', -- 数据库名称

'table-name' = 'dbo.orders'-- 要捕获更改的表名

);

4、创建目标表doris, 使用JDBC连接器将数据写入doris数据库

```
CREATE TABLE doris_test_sink (  
  id INT,  
  order_date DATE,  
  purchaser INT,  
  quantity INT,  
  product_id INT,  
  PRIMARY KEY (id) NOT ENFORCED  
) WITH (
```

WITH (

'connector' = 'doris',

'fenodes' = 'xx.xx.xx.xx :8030',

'table.identifier' = 'xxx.orders',

'username' = 'root',

'password' = '',

'sink.properties.format' = 'json',

'sink.properties.read_json_by_line' = 'true',

'sink.enable-delete' = 'true',

'sink.label-prefix' = 'doris_label');

-- 从t_source_sqlserver表中选择数据, 并将其插入到doris_test_sink表中

5、INSERT INTO doris_test_sink SELECT * FROM t_source_sqlserver;

操作验证

验证新增:

id	order_date	purchaser	quantity	product_id
1	2024-01-01	2	2	2
2	2024-01-01	3	3	3
3	2024-01-01	5	5	5
4	2024-01-01	5	5	5
5	2024-01-01	6	6	6
6	2024-01-01	7	7	7
7	2024-01-01	8	8	8
8	2024-01-01	9	9	9
9	2024-01-01	11	11	11
10	2024-01-01	10	10	10

验证修改:

id	order_date	purchaser	quantity	product_id
1	2024-01-01	2	2	2
2	2024-01-01	3	3	3
3	2024-01-01	5	5	5
4	2024-01-01	5	5	5
5	2024-01-01	6	6	6
6	2024-01-01	7	7	7
7	2024-01-01	8	8	8
8	2024-01-01	9	9	9
9	2024-01-01	11	11	11
10	2024-02-01	100	100	100

验证删除:

id	order_date	purchaser	quantity	product_id
1	2024-01-01	2	2	2
2	2024-01-01	3	3	3
3	2024-01-01	5	5	5
4	2024-01-01	5	5	5
5	2024-01-01	6	6	6
6	2024-01-01	7	7	7
7	2024-01-01	8	8	8
8	2024-01-01	9	9	9
9	2024-01-01	11	11	11
10	2024-02-01	100	100	100

Java UDF 使用

- ◆ **实现原理：** Doris 支持使用 JAVA 编写 UDF、UDAF 和 UDTF，对于用户来讲，Hive UDF 可以直接迁移至 Doris。
- ◆ **实现价值：** 通过使用UDF函数，可以实现复杂的数据处理、计算和转换，提高查询的灵活性和效率。

编写 UDF

- 使用 Java 代码编写 UDF，UDF 的主入口必须为 evaluate 函数。

```
/**
 * 数据加密
 * @param str
 * @return
 */
1 usage
private static String encrypt(String str) {
    String retStr = str;
    if (str != null && str.trim() != "") {
        DorisEncrypt.initKey();
        for (String key : encryptMap.keySet()) {
            retStr = retStr.replace(key, encryptMap.get(key));
        }
        return DatatypeConverter.printHexBinary(retStr.getBytes());
    } else {
        return "";
    }
}
```

创建 UDF

- 实现的 jar 包存放在远程服务端，让每个 FE 和 BE 节点都能获取到 jar 包。

```
CREATE FUNCTION DorisEncrypt(String)
RETURNS String PROPERTIES (
```

```
"file"="file:///data/software/doris/doris/udf/hive-function-1.0-SNAPSHOT.jar",
"symbol"="com.suofy.udf.DorisEncrypt",
"always_nullable"="true",
"type"="JAVA_UDF"
);
```

使用 UDF

- 当链接 session 位于数据内部，直接使用 UDF 名字会在当前 DB 内部查找对应的 UDF。否则用户需要显示的指定 UDF 的数据库名字，例：dbName.funcName

```
select id,DorisEncrypt(id) from sqlserver_dbo.c_dealer cd
```

id	dorisencrypt(cast(id as TEXT))
5	5E68
35	2A6B5E68
37	2A6B2466
41	2A6A293B
66	25672567

Doris 平台资源监控



Doris Manager

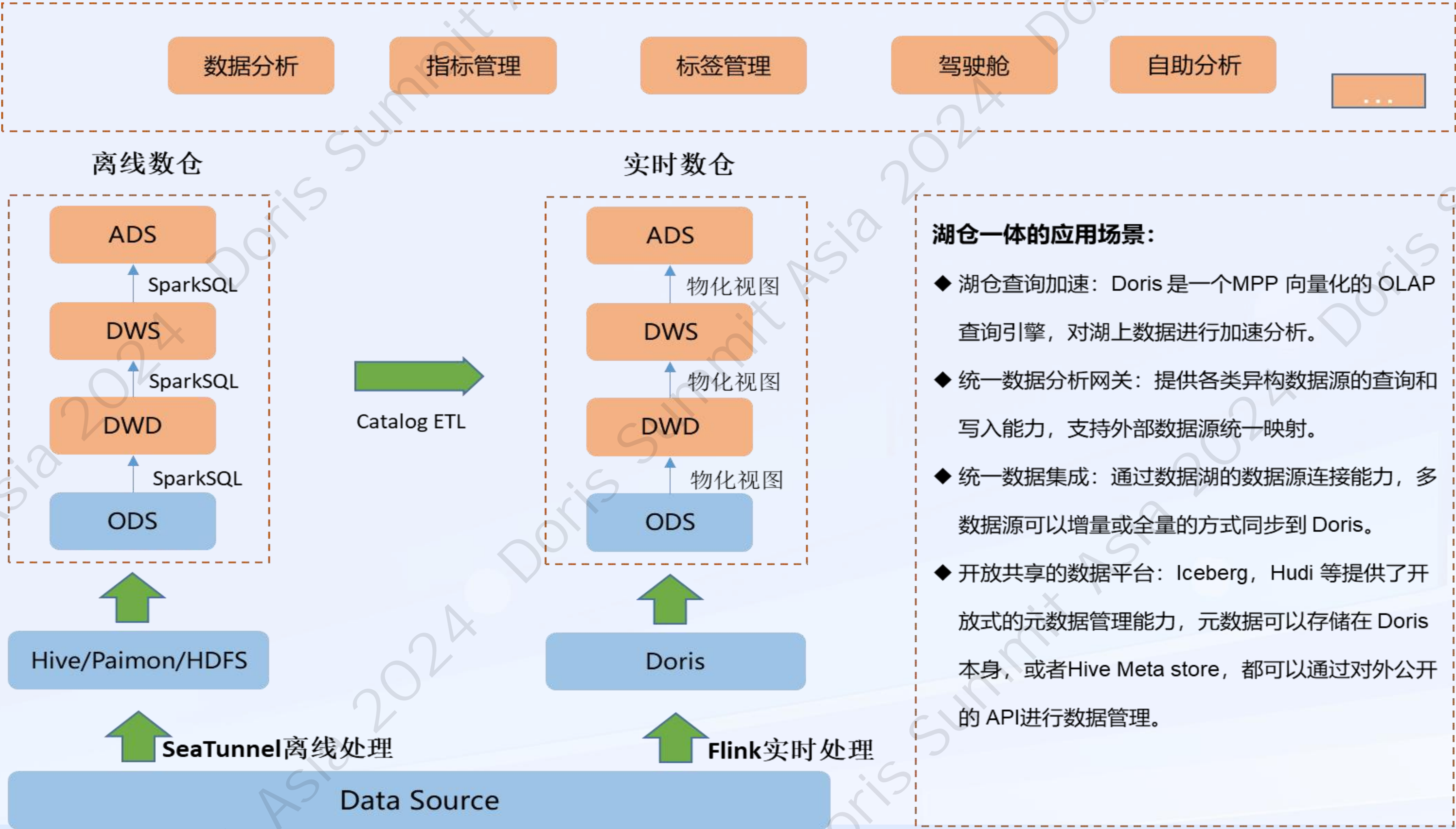
- ◆ 可视化运维管理（Cluster Manager for Apache Doris，简称 Manager），能够快速拉起一套 Doris/SelectDB 集群，集成了监控告警、集群巡检和 WebUI 等模块，提供了集群启停、扩缩容和升级等常规运维操作可视化的快捷入口。

Zabbix + Grafana

- ◆ 通过 Zabbix 进行监控，并通过 Grafana 将 Zabbix 监控到的数据进行可视化展示，更加方便了解主机以及虚拟机的运行状况。

基于 Apache Doris 的新一代数仓平台

- ◆ 离线数仓：Apache Spark + Apache Hive，Spark 是一个统一的大规模数据处理分析引擎，基于内存计算，Hive 负责结构化数据的存储管理。
- ◆ 实时数仓：Apache Doris，是一款基于 MPP 架构的高性能、实时的分析型数据库，海量数据查询秒级响应时间，支持高吞吐的复杂分析场景。



Thanks for Watching!