StarRocks

STARROCKS SUMMIT2023 | Evolution 极速进化，融合“新”生

微信基于 StarRocks 的湖仓一体实践

冯吕

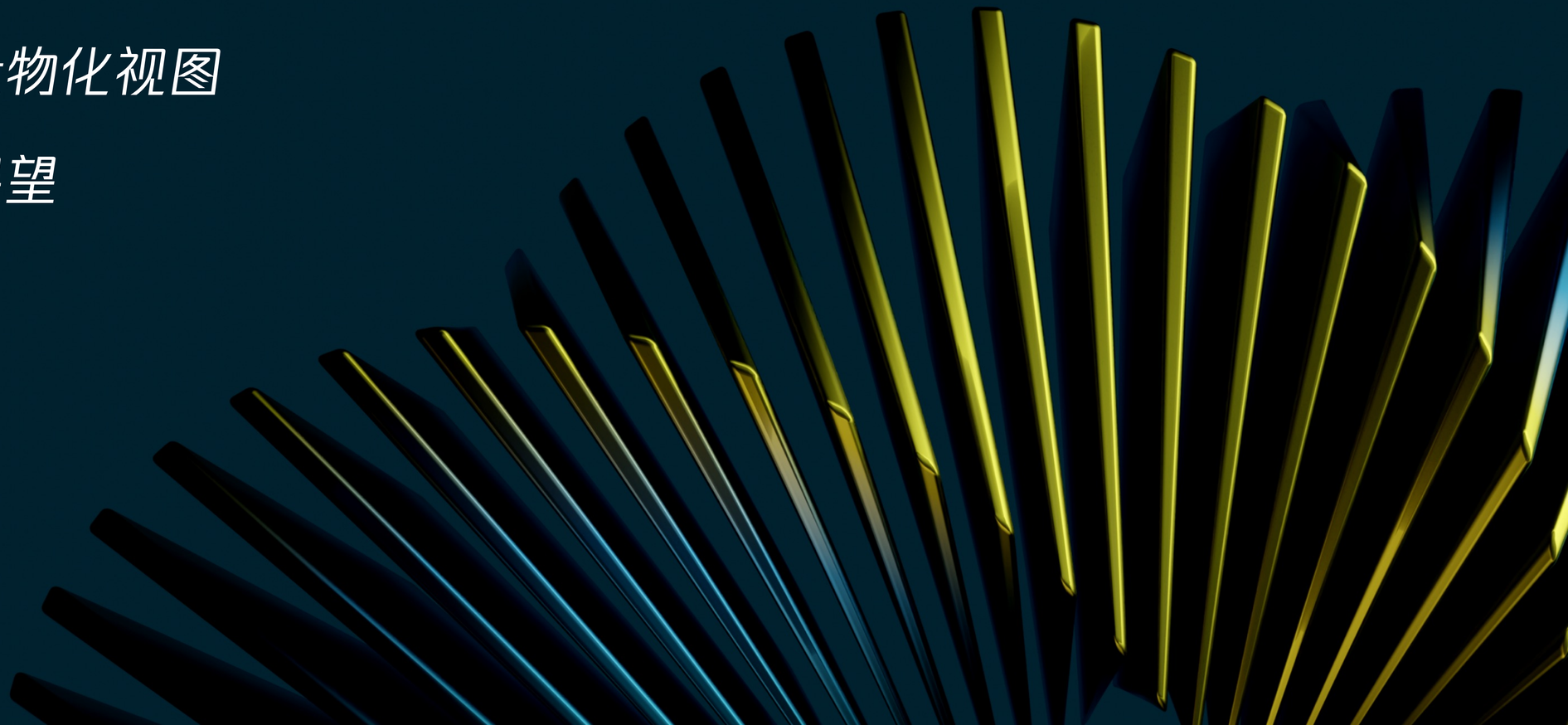
腾讯微信 OLAP内核研发工程师

01 背景

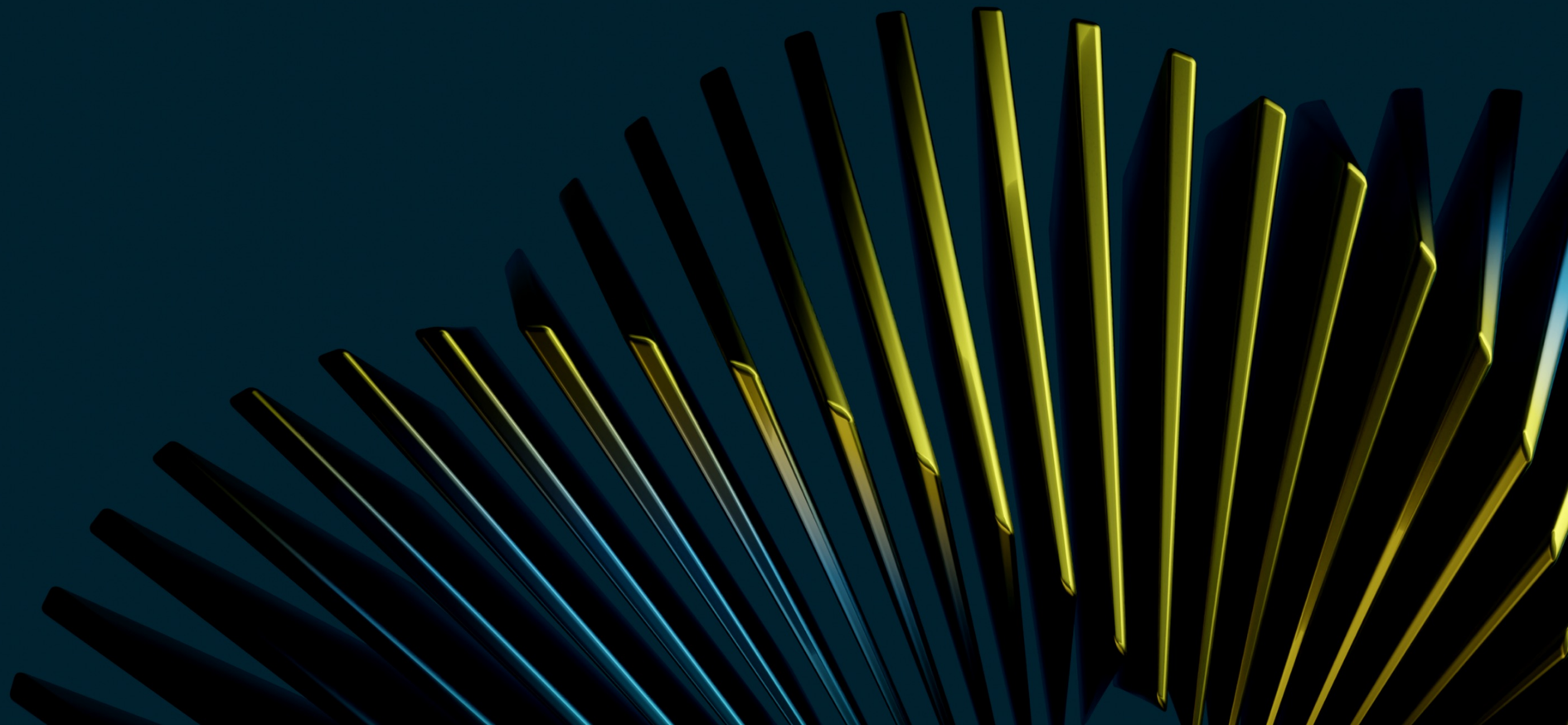
02 湖仓一体架构

03 实时增量物化视图

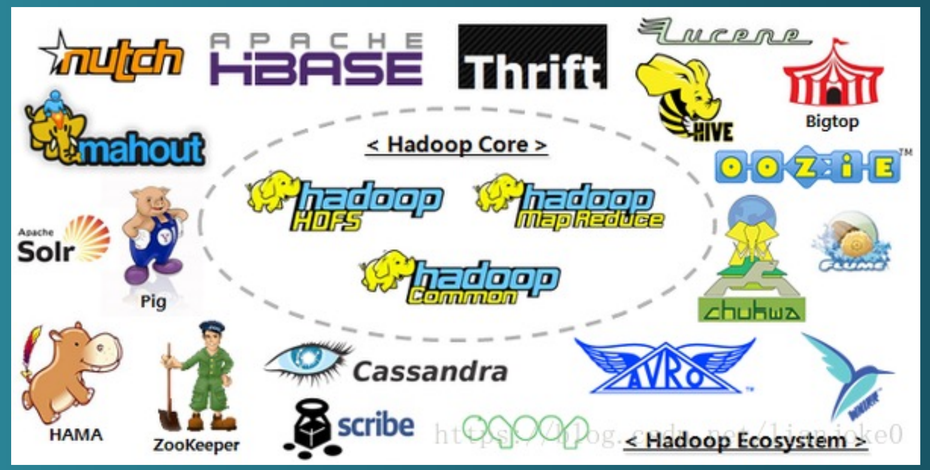
04 总结与展望



01 背景



背景：微信亚秒级实时数仓



- Hadoop生态

- 慢：查询慢，延迟高，开发慢
- 流批分离
- 架构臃肿

视频号等推荐系统对个性化体验的强烈诉求，催生了“亚秒级”分析系统的诞生

基于ClickHouse内核的亚秒级实时数仓

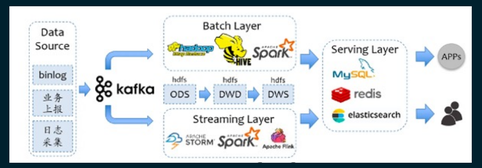
- **亚秒级响应**：亿行数据亚秒返回，万亿行数据秒级返回，支持A/B实验平台，BI分析等复杂指标场景，可追踪明细
- 海量数据低延迟：百亿级吞吐下秒级接入时延，满足实时指标统计、异常检测需求
- 云原生架构下：批流一体，T+1，实时分析模型统一，简单易运维
- 高可用：数据接入精确一次保障，单集群99.99% SLA

业界发展趋势 —— 湖仓一体&现代OLAP

数据分析技术栈演进历程

MySQL

实时



Lambda

Kappa

DataLake

极速

Hive	小时+
SparkSQL	小时-
Presto\ Druid\Kylin	分钟+

OLAP(ClickHouse)

流批一体

亚秒级响应

趋势

统一

Modern OLAP\湖仓一体

微信面临的难题

易用性

海量数据

- 单表日增万亿，100T
- 单次查询扫描 > 10亿/100亿
- 50+维度，100+指标

极速

- 查询计算耗时 T90 < 5s
- 数据低时效（秒/分钟级）

统一

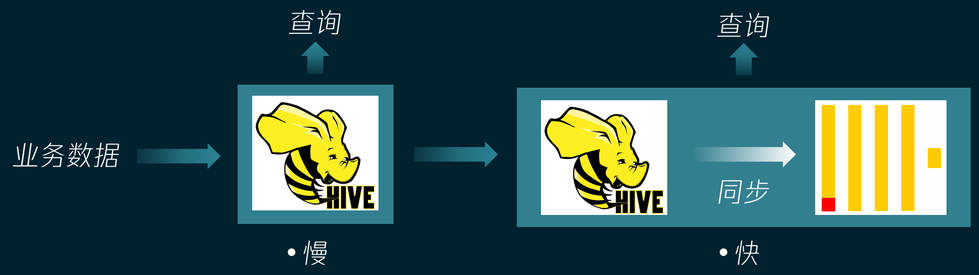
- 流批一体 [Flink]
- ALL in One [Spark]
- One Source [湖存储]

“统一”—— 极速湖仓下的诉求

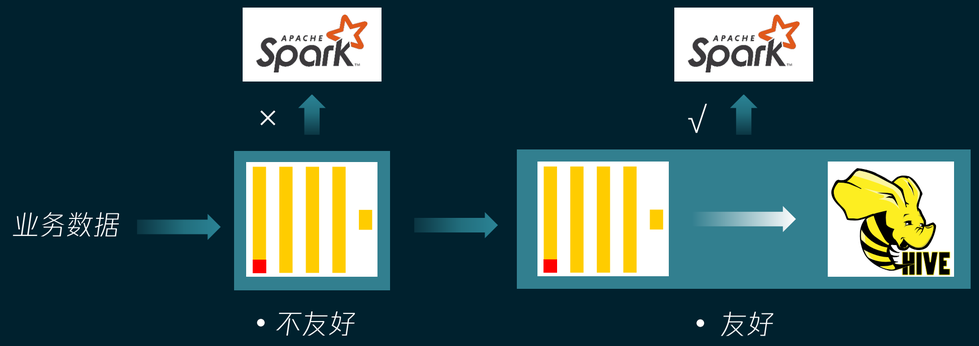
问题 通用大数据计算引擎处理OLAP数仓体验割裂，存储多份

目标 湖仓一体化，解决通用大数据处理“存储统一”问题

湖困境 离线流- Hadoop数据即席查询需要先导出OLAP再查询加工，资源浪费，工序繁琐



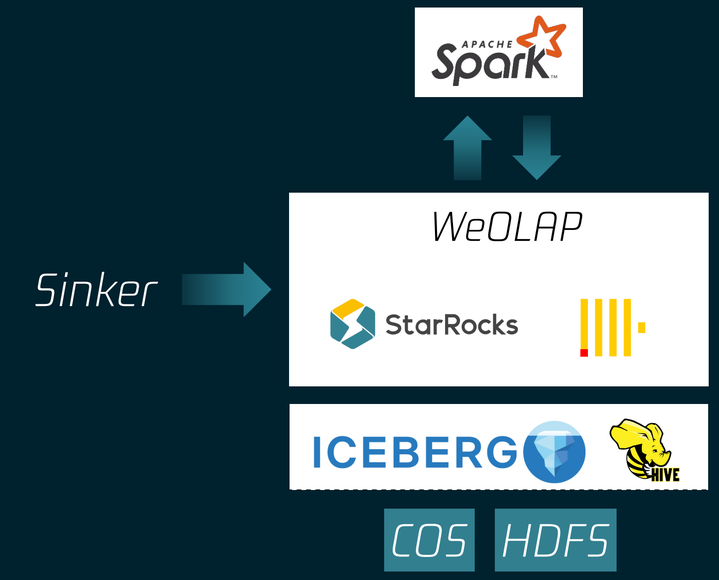
仓困境 实时流-通用大数据计算引擎处理OLAP实时数仓生态不够友好



问题：
- 两份存储
- 口径不一致
- 额外步骤
- 难标准化

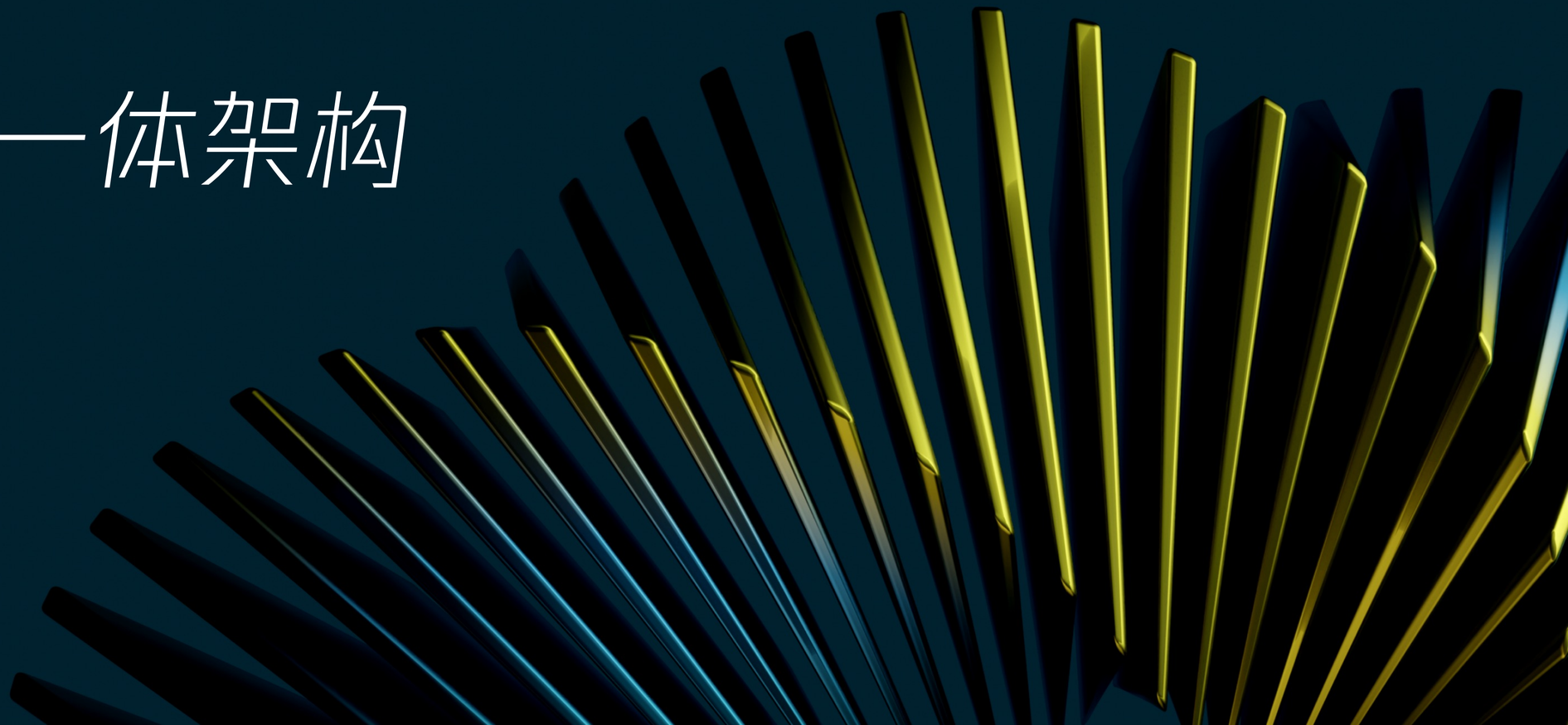
- 存算分离
- 极速查询
- 存储统一
- 统一接口
- Meta统一

Single source of truth
极速、统一、友好



湖仓一体化愿景：面向SQL，用户不再感知底层架构

02 湖仓一体架构

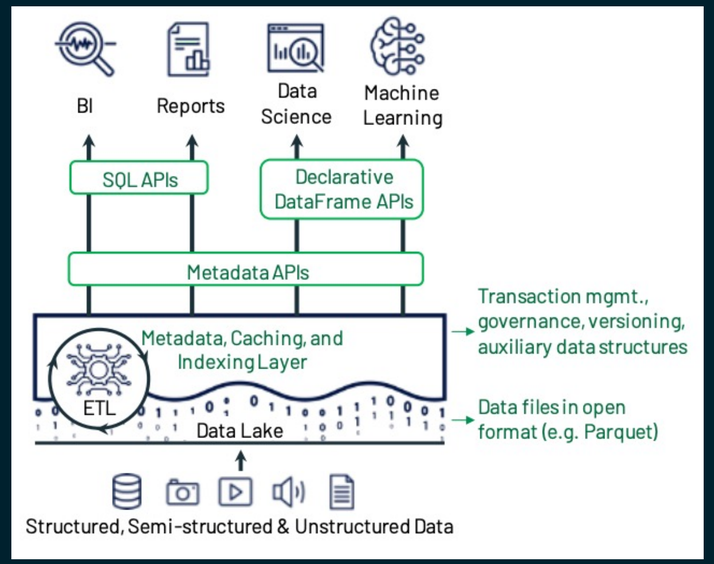


技术路线1——湖上建仓

数据湖基础上实现数仓的功能，代替传统数仓，构建LakeHouse

- 传统Hadoop系统引入Delta lake、Hudi、Iceberg或Hive 3.0的ACID等更新技术
- 引入Presto、Impala等SQL on Hadoop查询引擎
- 引入Hive Meta Store进行统一元数据和权限管理
- 引入对象存储作为底层存储等数仓特性，形成湖仓一体

业内代表(Databricks)



公司内演变方案

时效：小时\天级

查询：分钟

presto

APACHE Spark

业务数据

HIVE

时效：分钟

查询：秒~分钟

StarRocks

秒级〔80%〕

APACHE Spark

分钟级\海量

业务数据

ICEBERG

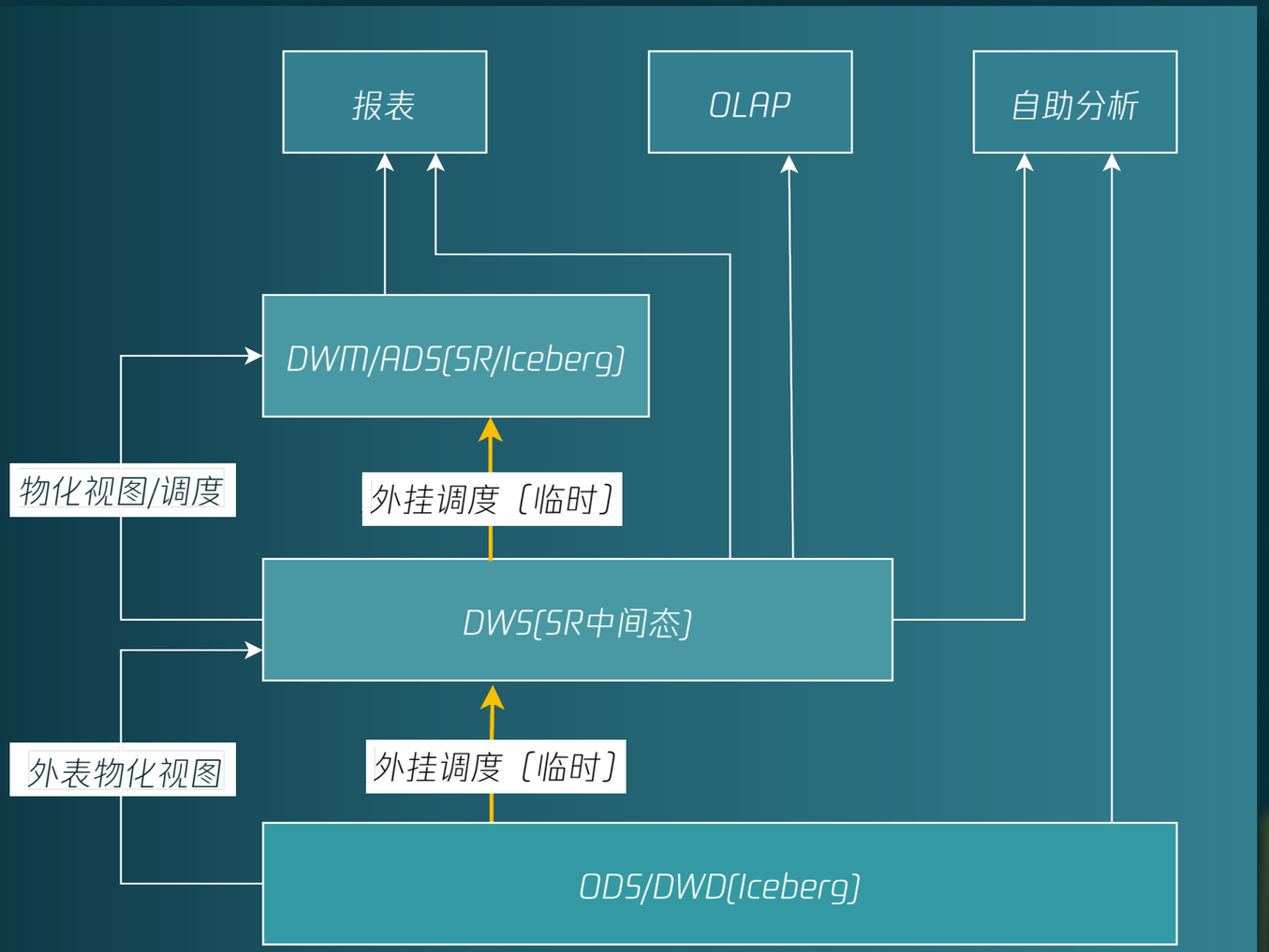
HIVE

技术路线1——湖上建仓

优点：
- 成本低
- 简单
- Hadoop兼容更好

缺点：
- 分钟延迟，5～9分钟以上
- 需要通过缓存等手段对ODS、DWD层的查询进行加速

适用场景：
- 湖为主，更适合于离线分析为主的场景



技术路线2 —— 仓湖融合（冷存下沉）

数仓中加入跨源融合联邦查询的功能，内部存储打通，不经过ETL直接分析数据湖

例如： aws redshift 的 spectrum, Google bigquery 的 gcp cloud storage， hive数仓的 Storage Handler技术等

优点

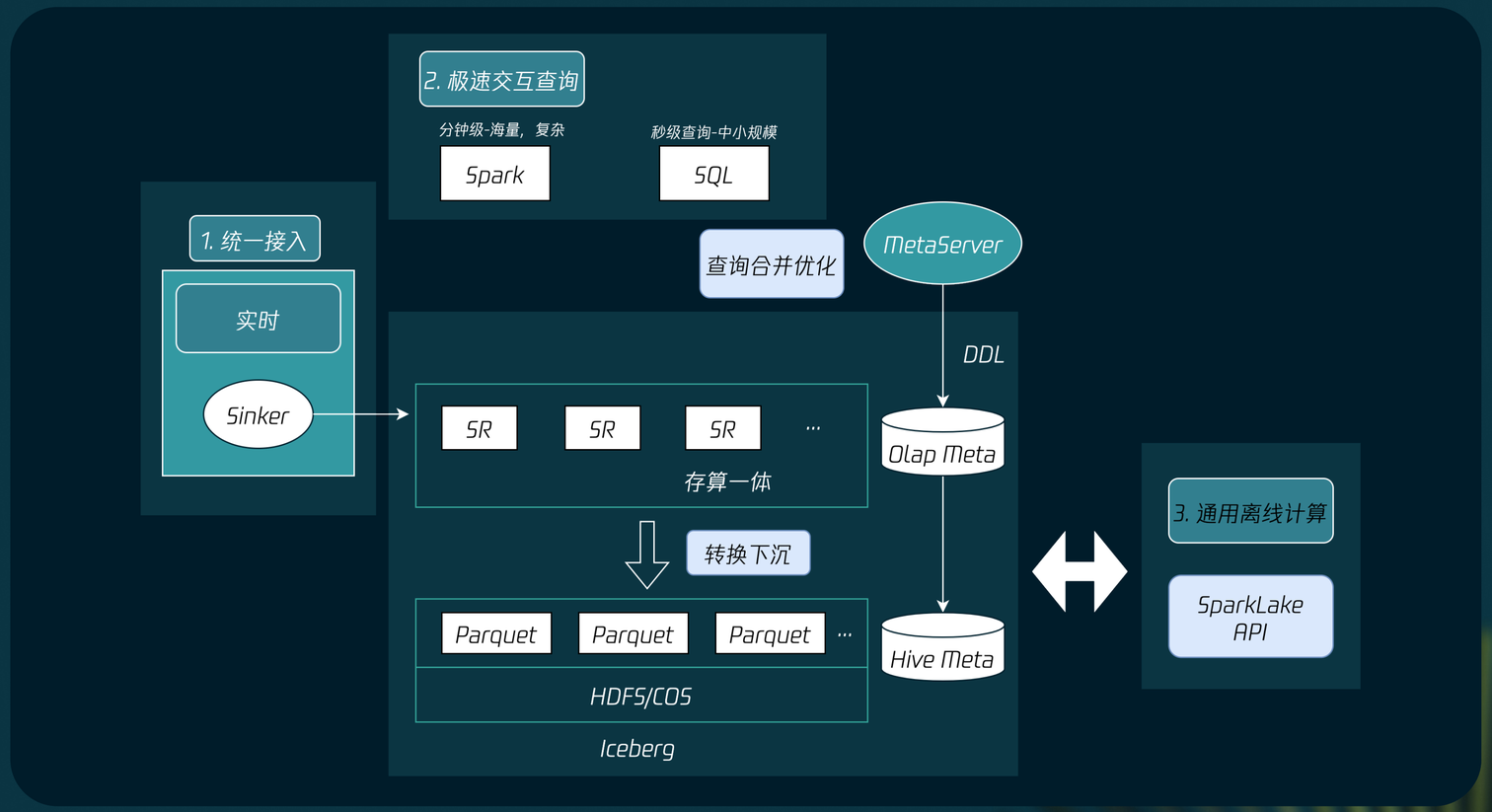
- 数据实时性高（秒～2分钟内）
- DWD查询响应更快

缺点

- 成本高（热数据TTL）
- Hadoop兼容不如湖上建仓

适用场景

- 仓为主，更适合于实时分析为主的场景



湖上建仓 VS 仓下沉湖

各有优缺点

	湖上建仓	仓下沉湖
数据时效性	5~10分钟	秒～2分钟内
DWD查询响应	秒	亚秒
ADS查询响应	亚秒	亚秒
Hadoop兼容	优	一般

成本OR 性能/时效，微信业务场景两种路线都有需求

湖仓一体方案

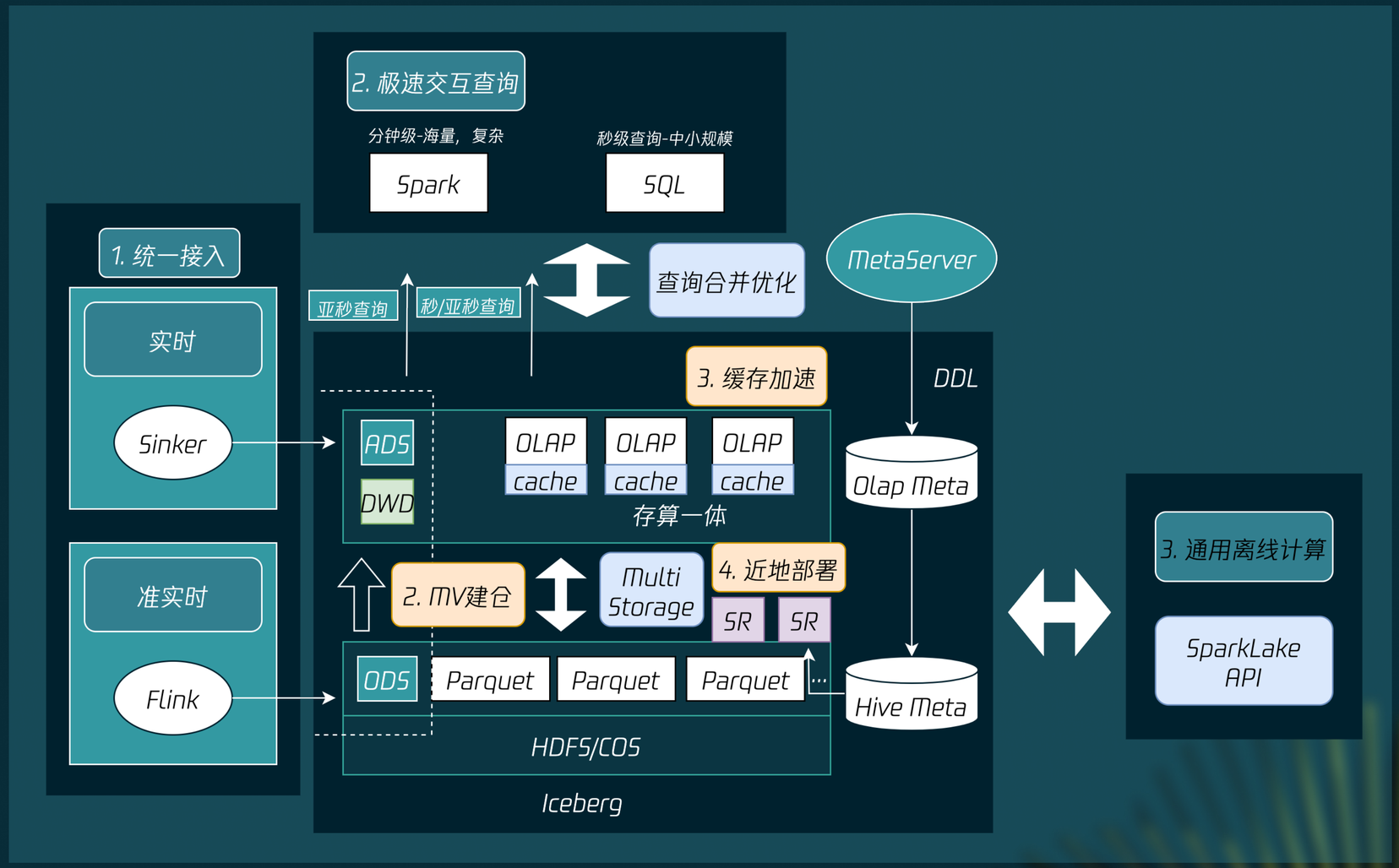
同时支持实时接入和
准实时接入

用户可选择

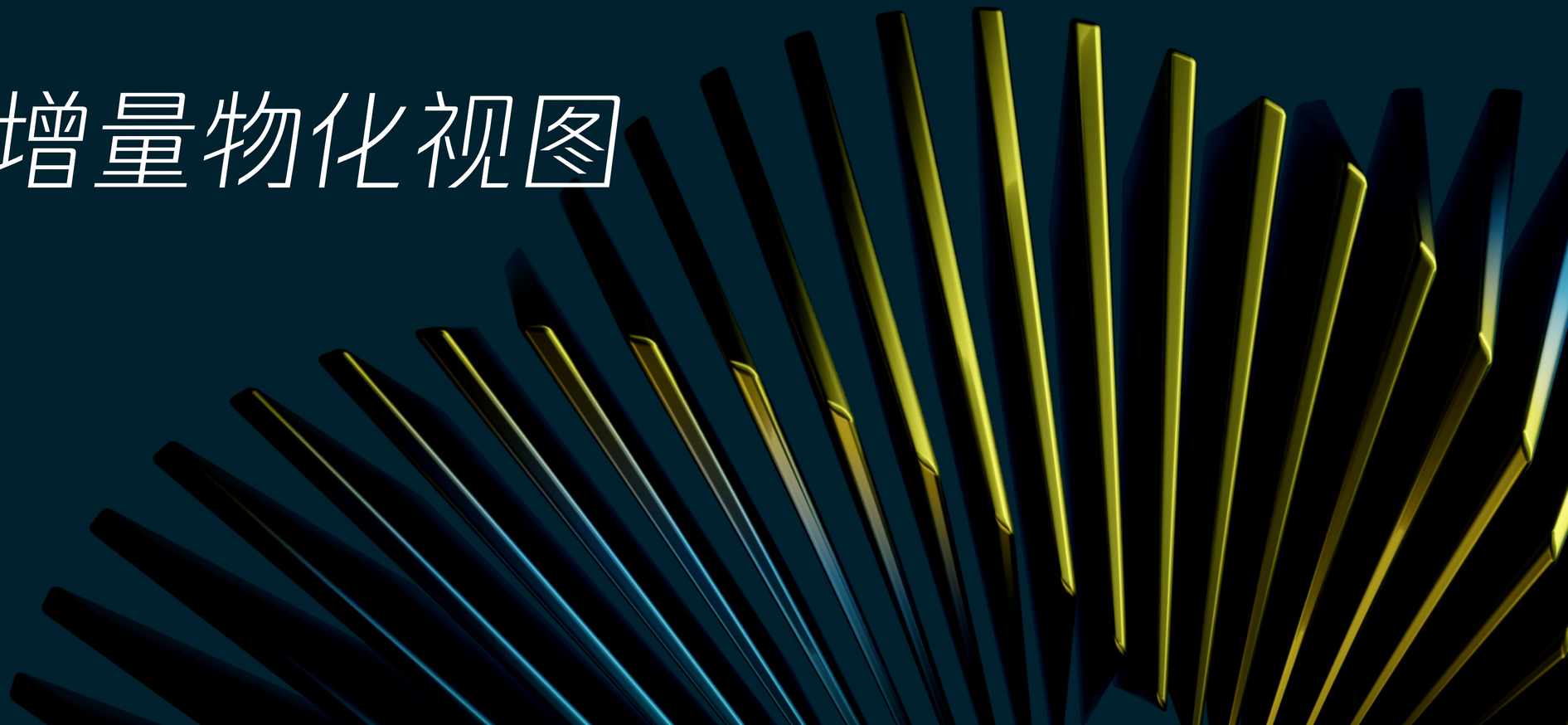
- 成本OR性能/时效
- 实时下沉/离线建仓
  可切换

场景

- 实时/离线接入
- 极速BI分析
- 通用离线计算



03 实时增量物化视图



实时场景物化视图建仓

对于实时性、性能要求高的业务场景，需要通过实时增量物化视图进行加速

微信实时物化视图特点

- ✓ 大规模，单表数据量大：增量更新，不能全量刷新
- ✓ 实时性要求高：同步刷新
- ✓ 多表指标拼接：多个基础表的指标拼接到同一个物化视图
- ✓ 物化视图写入时进行高性能维表关联
- ✓ ...

基于 StarRocks 的物化视图功能，我们与社区共同在实时物化视图上进行了新的探索，以适应更加复杂的分析场景。

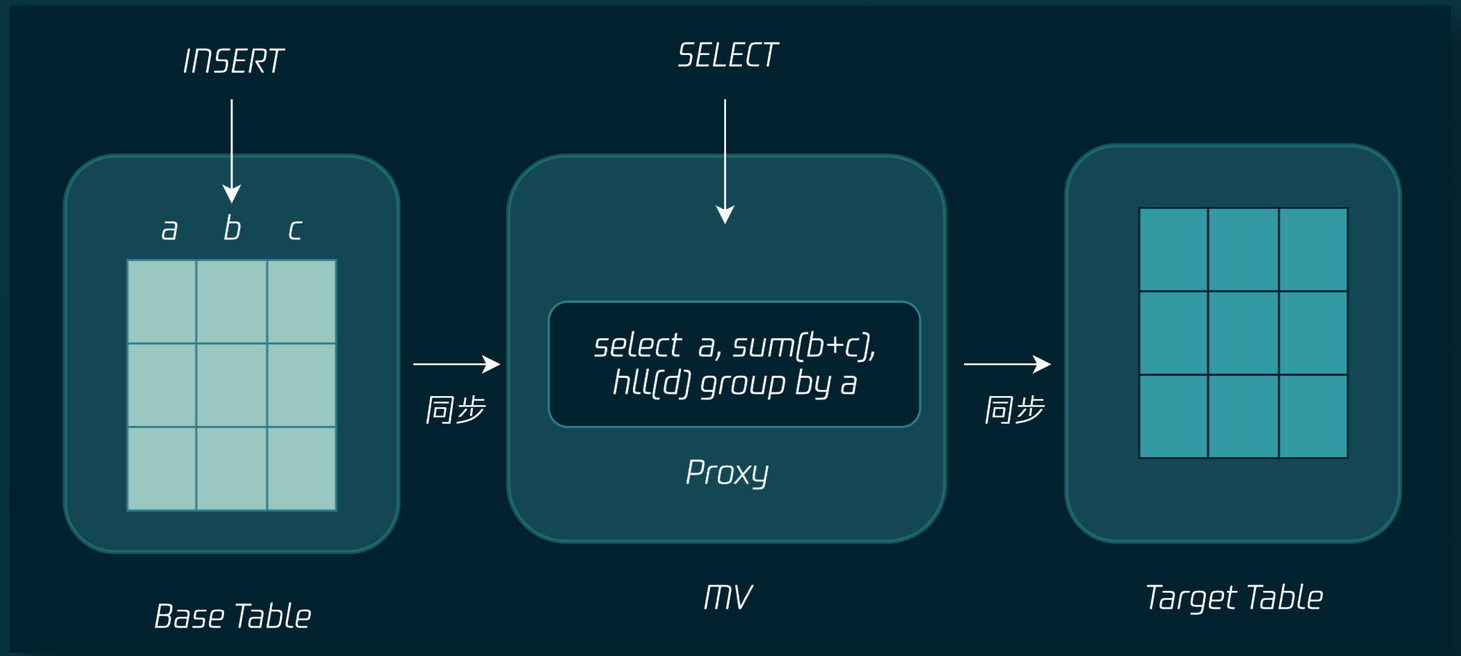
实时增量物化视图设计

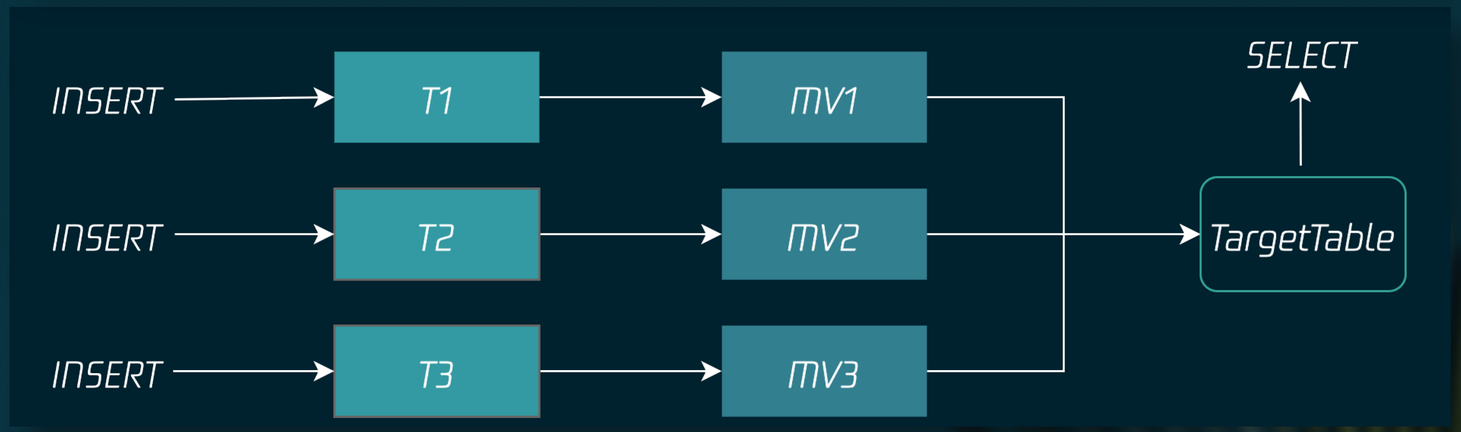
基础表与目标表解耦

提供不同存储周期，基础表ODS保留3～7天，目标表DWS保留半年到一年

物化视图计算逻辑与计算结果解耦，业务无需关心计算过程，提高易用性，同时上下游业务逻辑解耦

易于实现多个基础表协议关联、指标拼接，多个基础表物化视图写同一目标表





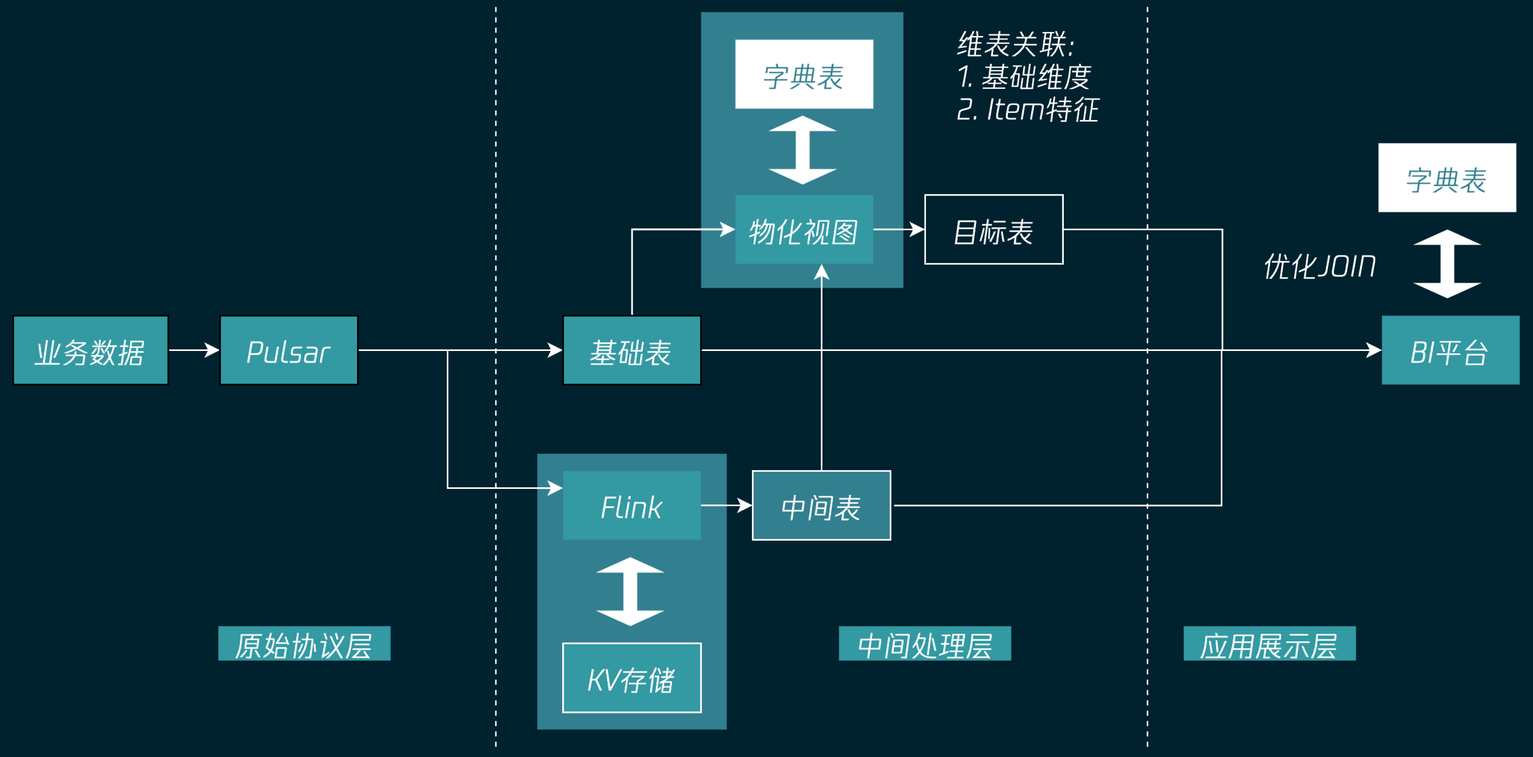
实时增量物化视图设计

基于全局字典的维表关联

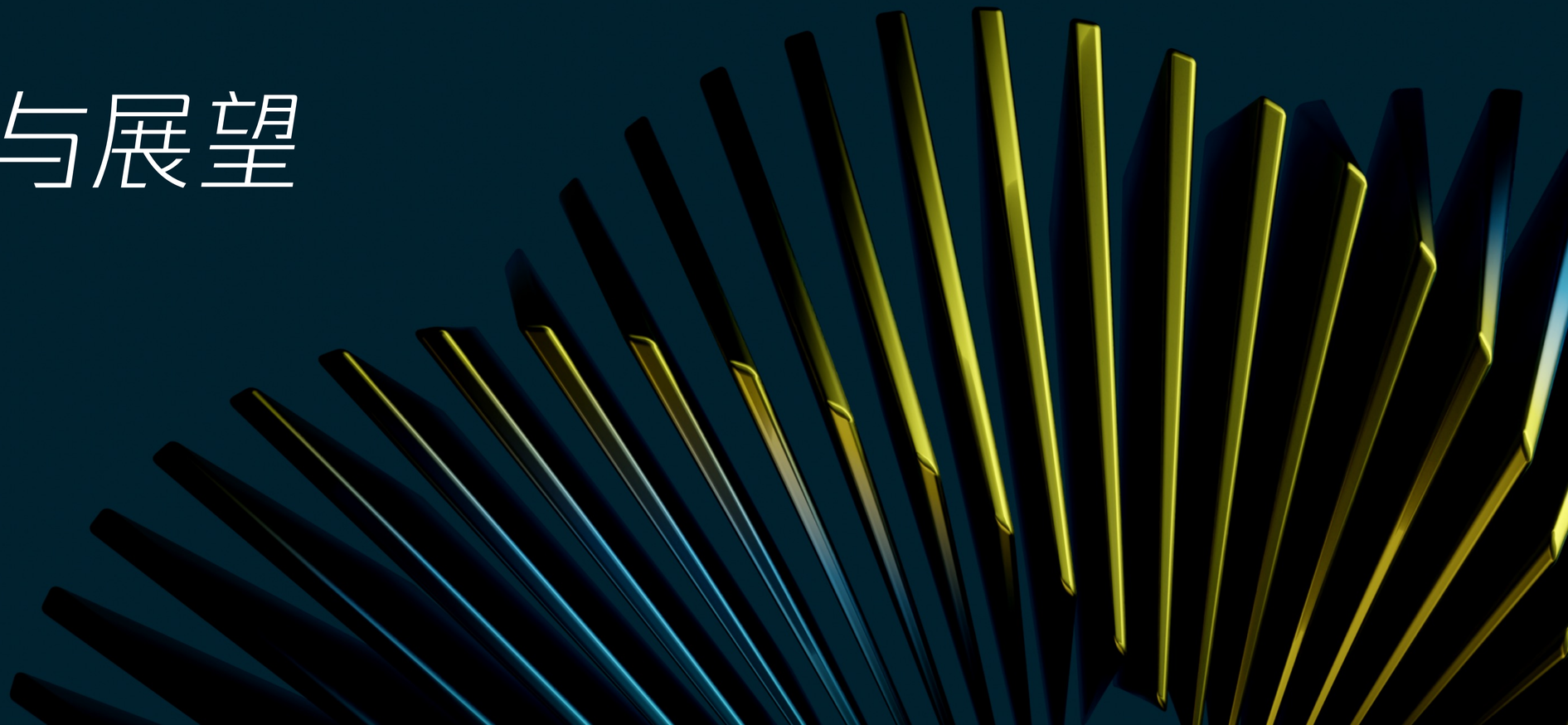
替代Flink，BI场景，流式计算通用问题

基于高性能字典表，在数据库内部完成维度表关联

查询时，也可以通过字典优化JOIN性能



04 总结与展望



StarRocks 目前在微信的线上使用情况

在微信多个业务中上线使用

- 视频号直播、微信键盘、微信读书、公众号等

规模	■ 集群规模数百台机器 ■ 数据接入量近千亿
收益举例	■ 直播业务场景，通过湖上建仓方案改造，运维任务数减半，存储成本降低65%以上，离线任务产出时间缩短两小时
定位	■ ClickHouse：实时数仓，亚秒，实时，高QPS，OLAP为主 ■ StarRocks：湖仓一体，秒，准实时，低QPS，离线加工为主 目前我们仍在探索更大的湖仓一体边界，在离线与实时融合的道路上不断努力

未来展望

期待的社区新功能

- Stream MV + 外表物化视图：更好的湖上建仓方案

理想中的湖仓一体

- 面向SQL，用户不再感知底层架构
- 接入/查询体验统一，存储统一；
- 秒级/分钟级延迟架构体验统一，亚秒/分钟级分析统一；
- 单个/百QPS体验统一；
- SQL交互标准统一；

StarRocks

STARROCKS SUMMIT2023 | Evolution
极速进化，融合“新”生

感谢聆听！

Thanks for listening !