云原生数据仓库
Apache Doris 特性解读与未来规划

陈明雨
Apache Doris PPMC
百度

GOTC

01 • 认识 Doris

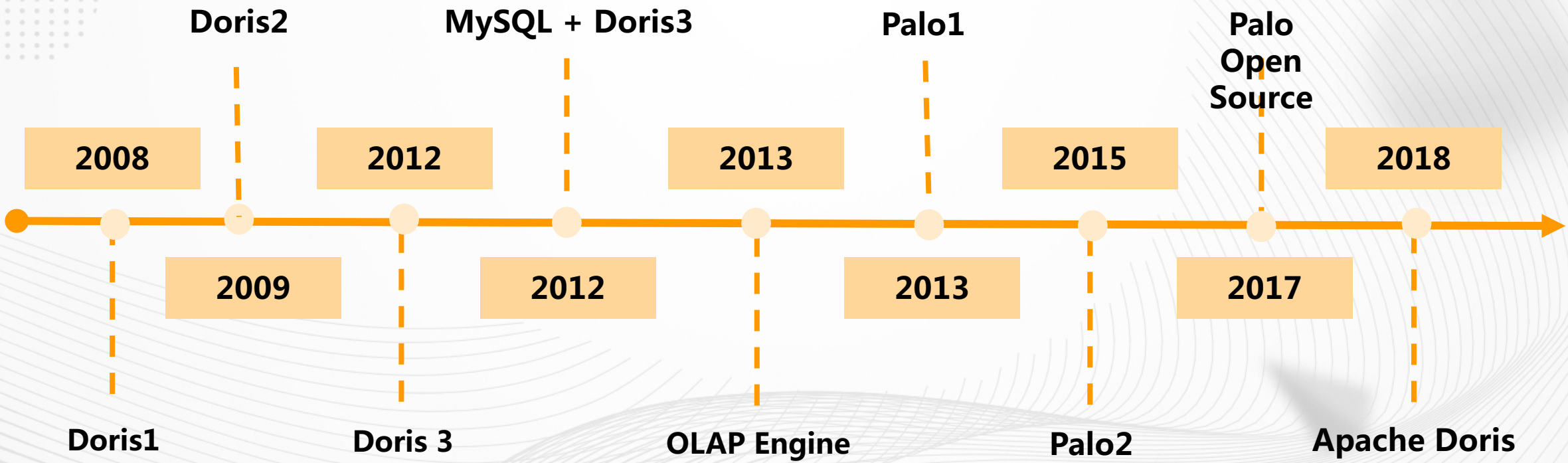
02 • 为什么要选择 Doris

03 • Doris 的进行时和未来式

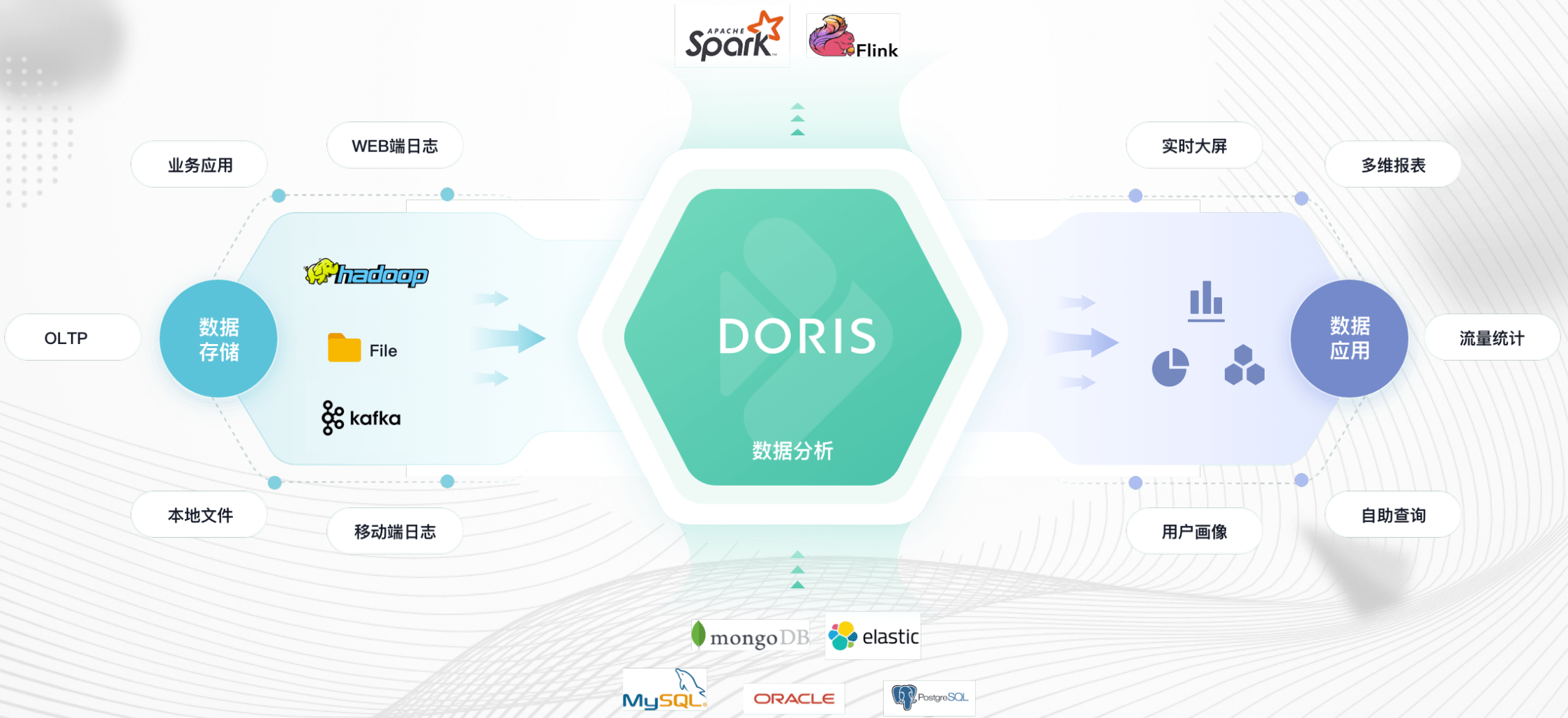
GOTC

01

认识Doris


Doris2
MySQL + Doris3
Palo1
Palo Open Source
2008
2012
2013
2015
2018
2009
2012
2013
2017
Doris1
Doris 3
OLAP Engine
Palo2
Apache Doris

Doris在数据流中的定位



GOTC

02

为什么选择Doris

GOTC

当我们选择一款分析型数据库时，我们在选择什么？

GOTC

1
足够
简单

2
足够
高效

3
功能
丰富

4
开源
开放

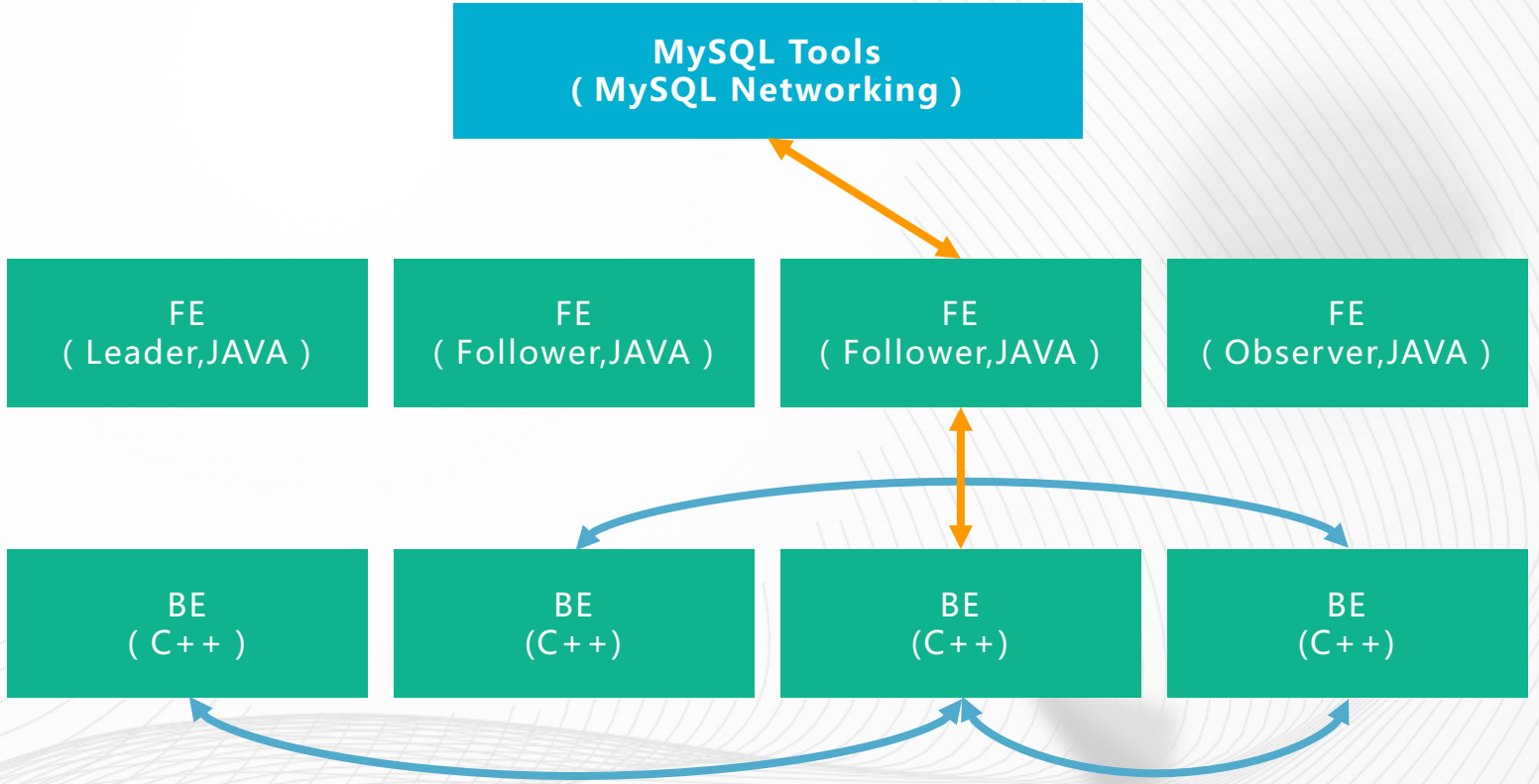
GOTC

1 足够简单

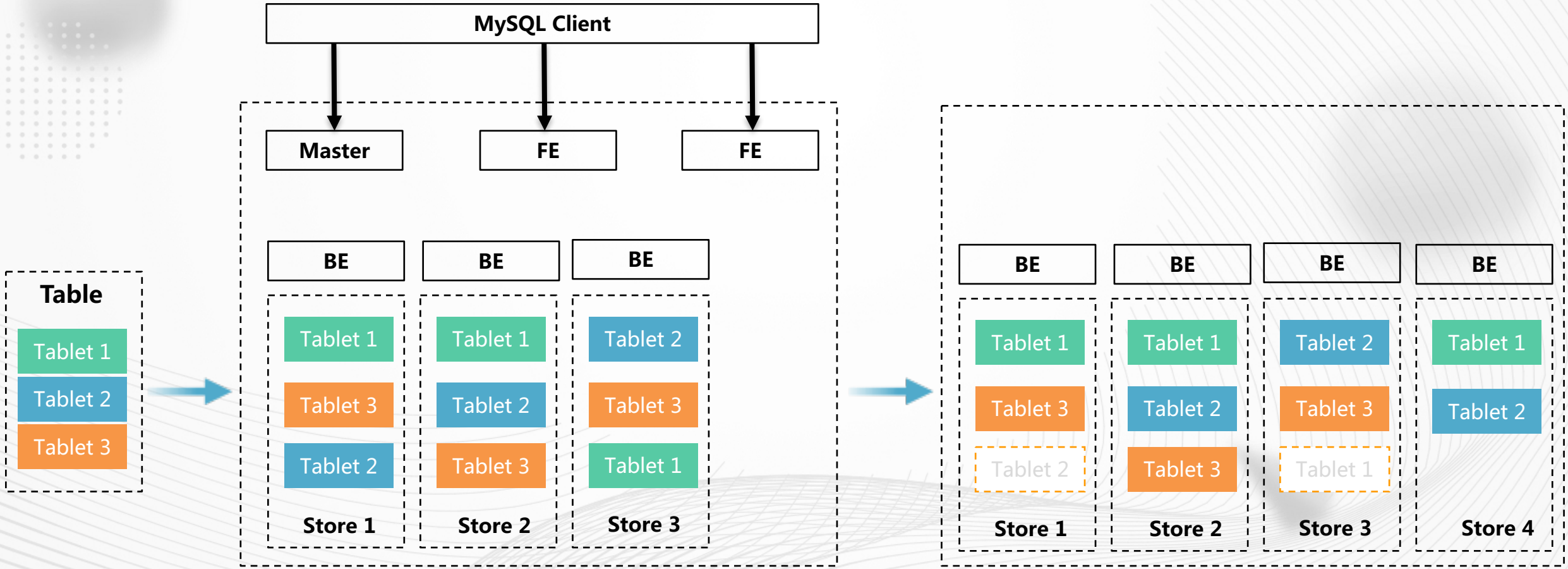
- 架构图都没看懂，怎么运维系统？
- 副本坏了要手动修复？扩容集群要1人周？
- 我不管，我只会写SQL！
- 每次升级都要给用户发通告？
- 。。。

整体架构简单，产品易用

- 高度兼容MySQL协议
- 主从架构，不依赖任何其他组件
- FE负责解析/生成/调度查询计划
- BE负责执行查询计划、数据存储
- 任何节点都可线性扩展



多副本存储，自动数据迁移、副本均衡



GOTC

标准 SQL 支持

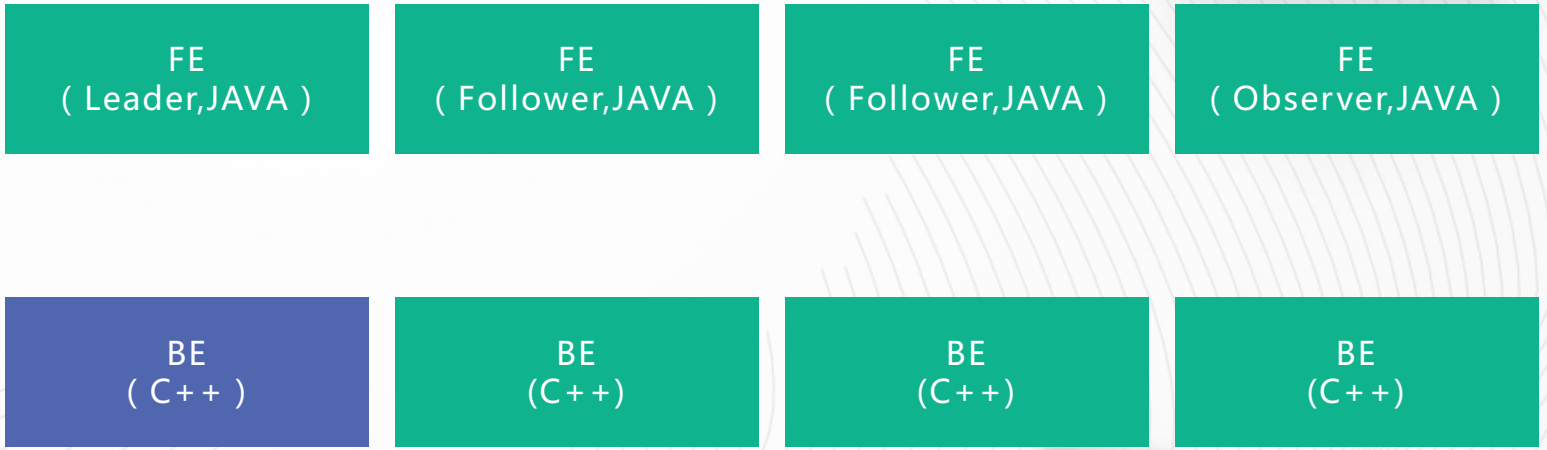
- 单表聚合、排序、过滤
- 多表关联、子查询、窗口函数

```sql
select
    l_returnflag,
    l_linestatus,
    sum(l_quantity) as sum_qty,
    sum(l_extendedprice) as sum_base_price,
    sum(l_extendedprice * (1 - l_discount)) as sum_disc_price,
    sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) as sum_charge,
    avg(l_quantity) as avg_qty,
    avg(l_extendedprice) as avg_price,
    avg(l_discount) as avg_disc,
    count(*) as count_order
from
    lineitem
where
    l_shipdate <= date '1998-12-01' - interval '93 day'
group by
    l_returnflag,
    l_linestatus
order by
    l_returnflag,
    l_linestatus;
```

GOTC

灰度升级

- 分布式系统升级，并没有那么可怕
- 向前兼容，滚动升级
- 查询自动重试，减少业务感知



2 足够高效

存储引擎

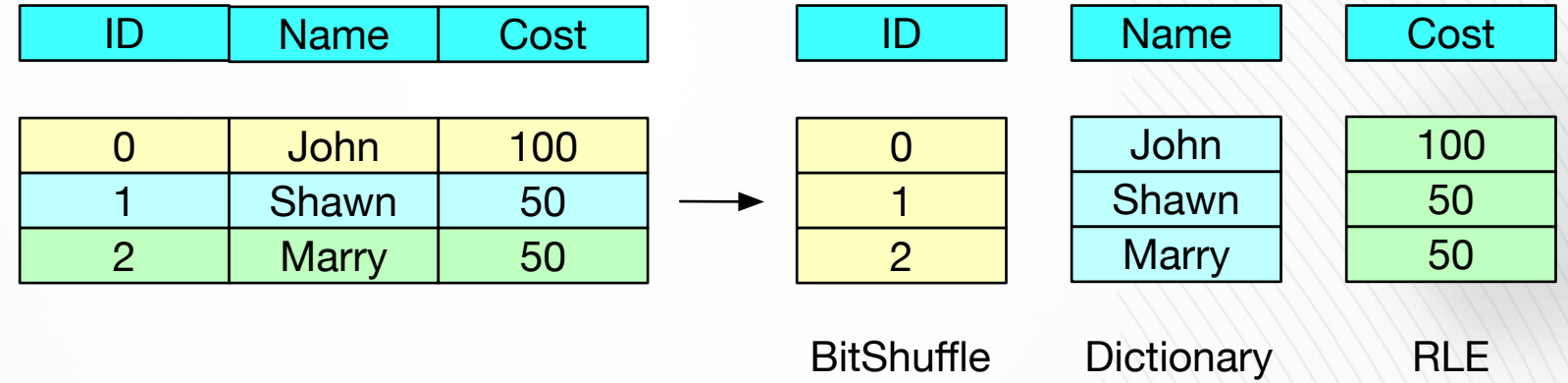
查询引擎

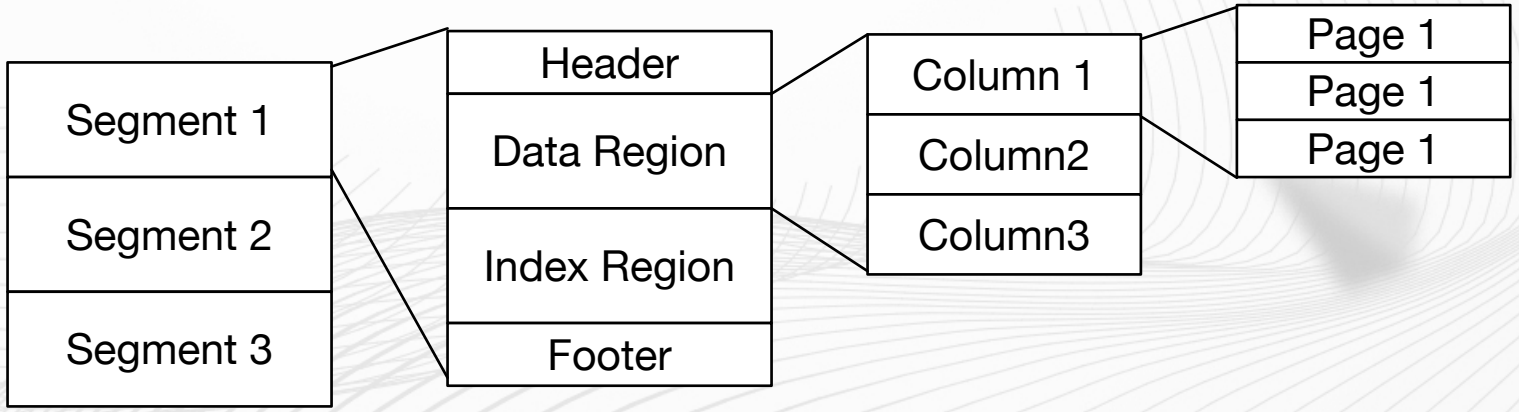
优化器

存储引擎

存储结构

- 多种编码方式
- 自适应编码
- 按需读取

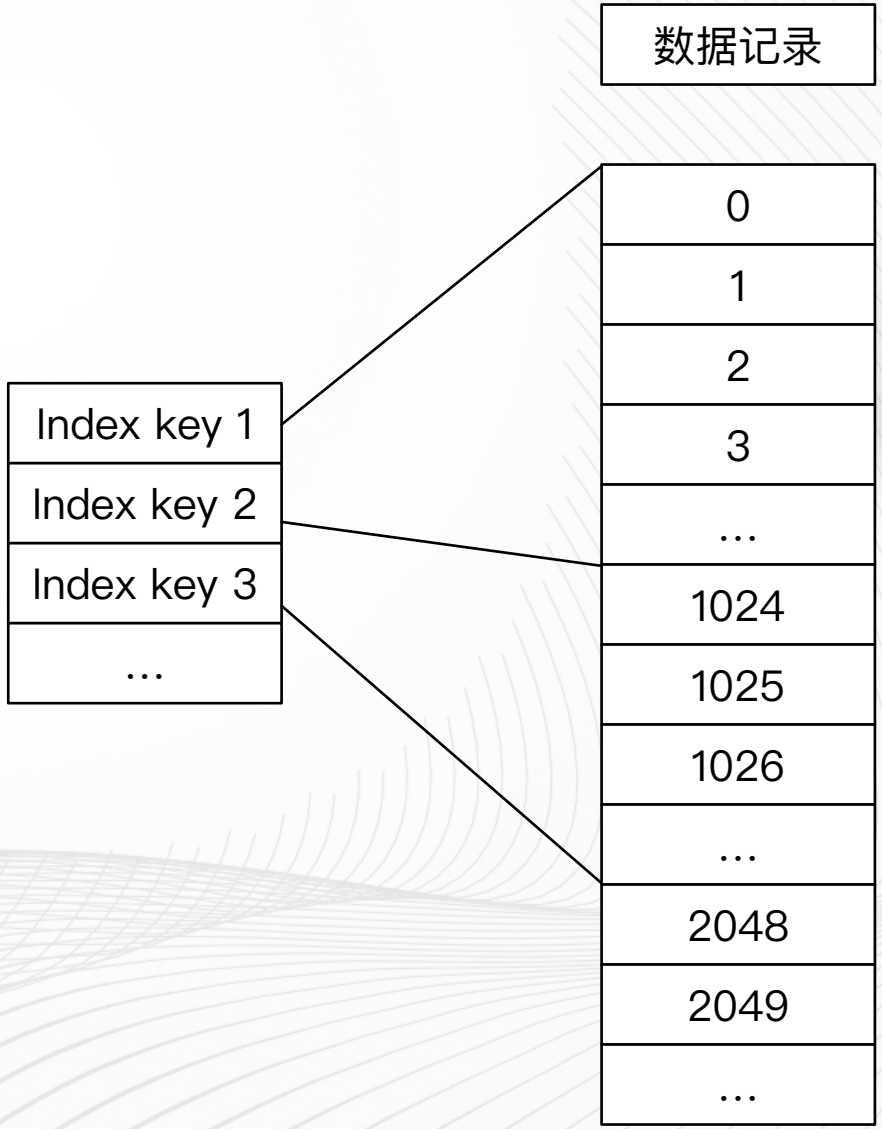




存储引擎

索引结构

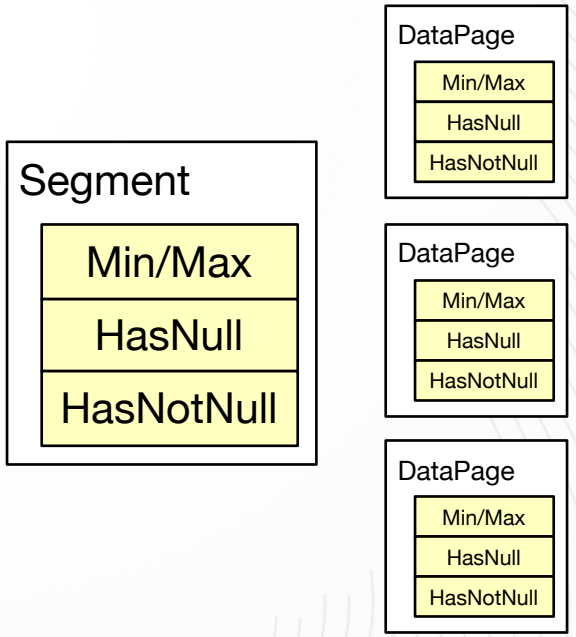
- 前缀稀疏索引：基于排序列查询的快速过滤
- Min Max 索引
- Bloom Filter
- 倒排索引



存储引擎

索引结构

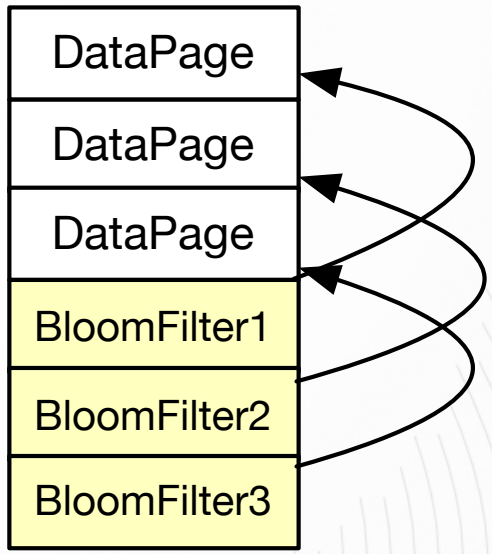
- 前缀稀疏索引
- **Min Max 索引：Segment 和 Page 级别快速过滤**
- Bloom Filter
- 倒排索引



存储引擎

索引结构

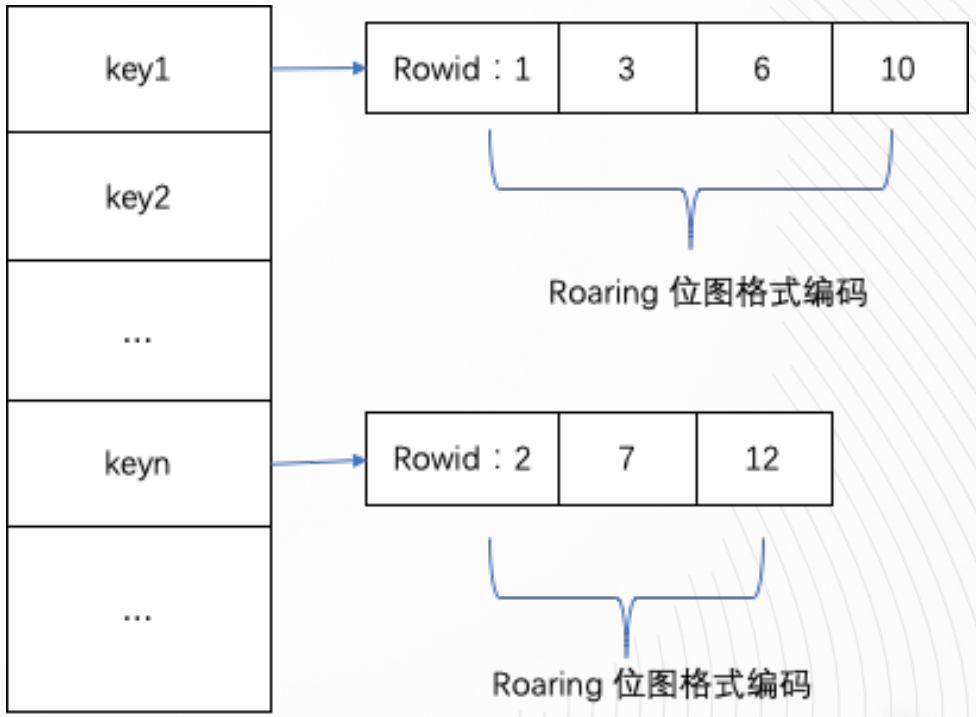
- 前缀稀疏索引
- Min Max 索引
- **Bloom Filter：Page 级别的快速过滤**
- 倒排索引



存储引擎

索引结构

- 前缀稀疏索引
- Min Max 索引
- Bloom Filter
- **倒排索引：基于 Bitmap 快速精确查找**

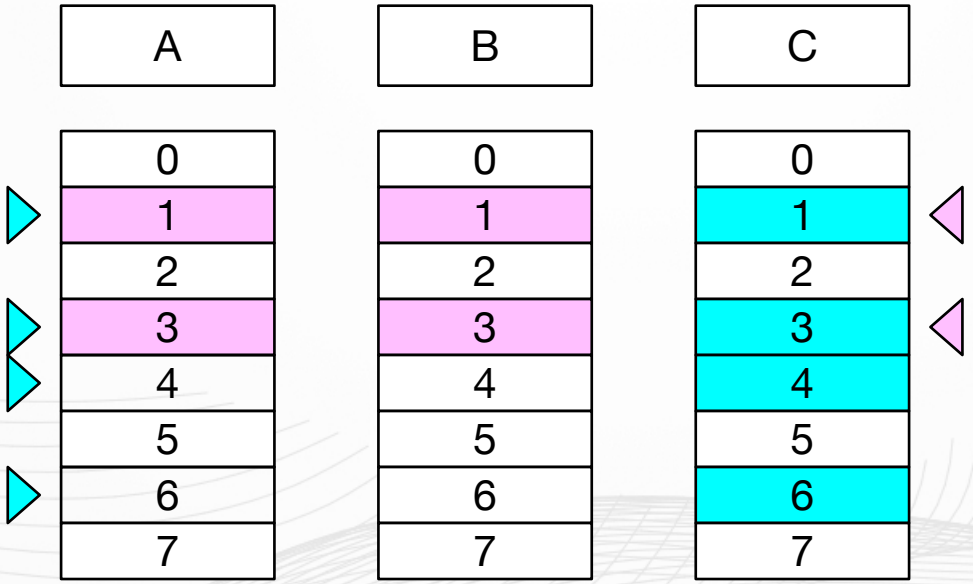


存储引擎

延迟物化

- 先过滤、再取数
- 减少 IO

SELECT A, B, C from tbl where A=2 and C=3

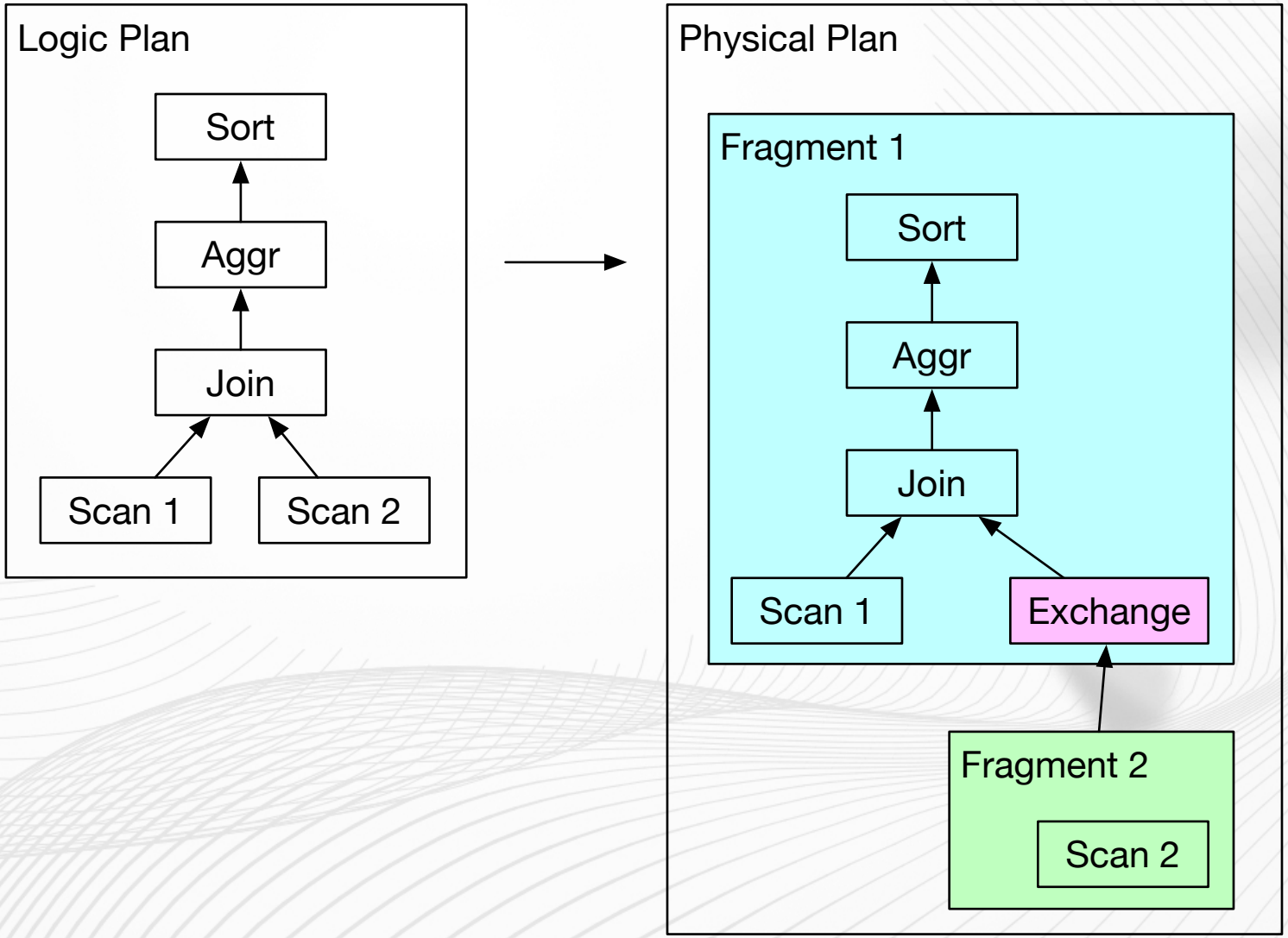


1. 根据A列索引定位到行号
2. 读取C列数据
3. 根据C列过滤条件过滤数据
4. 读取A、B两列数据

查询引擎

并发执行模型

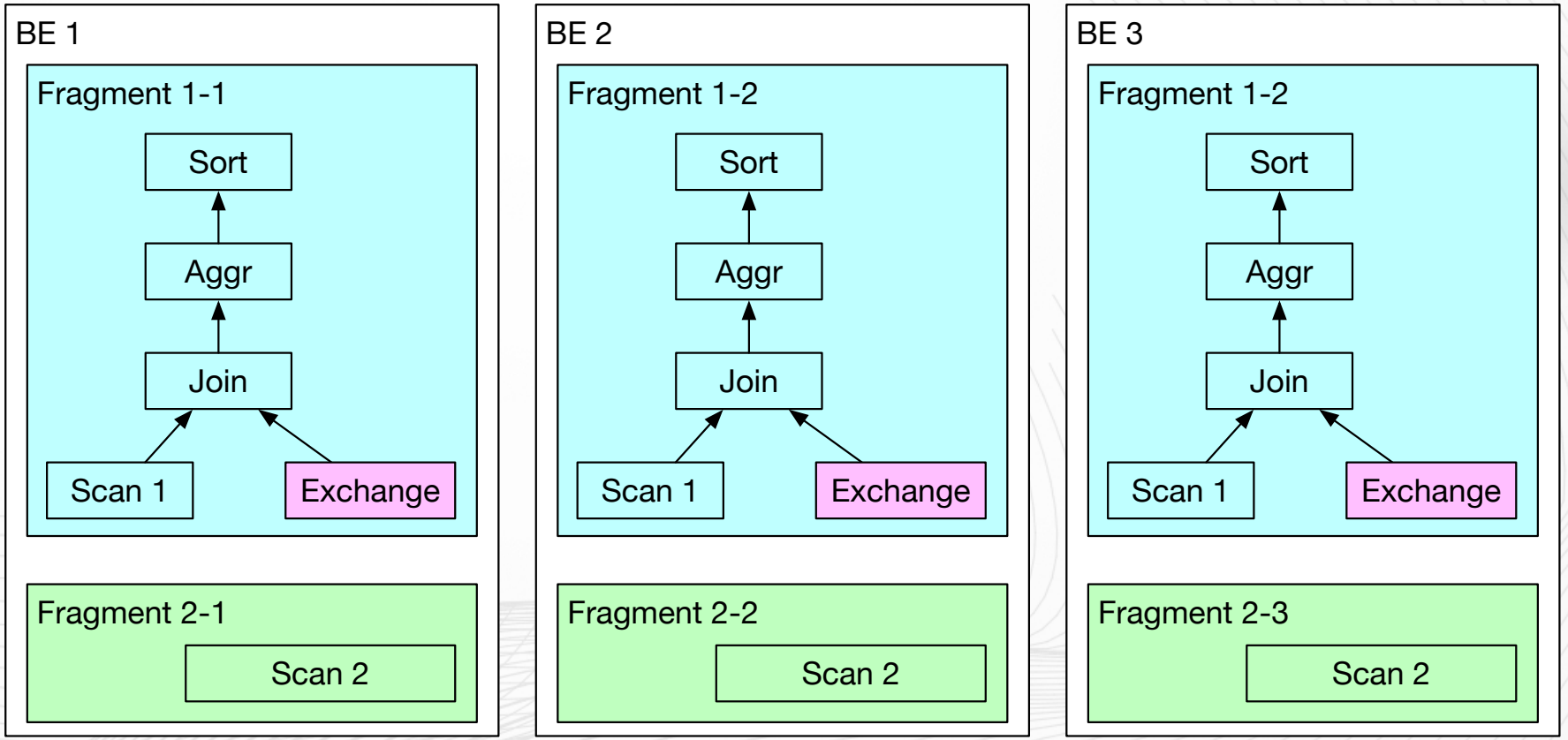
- 真正的 MPP
- 火山模型



查询引擎

并发执行模型

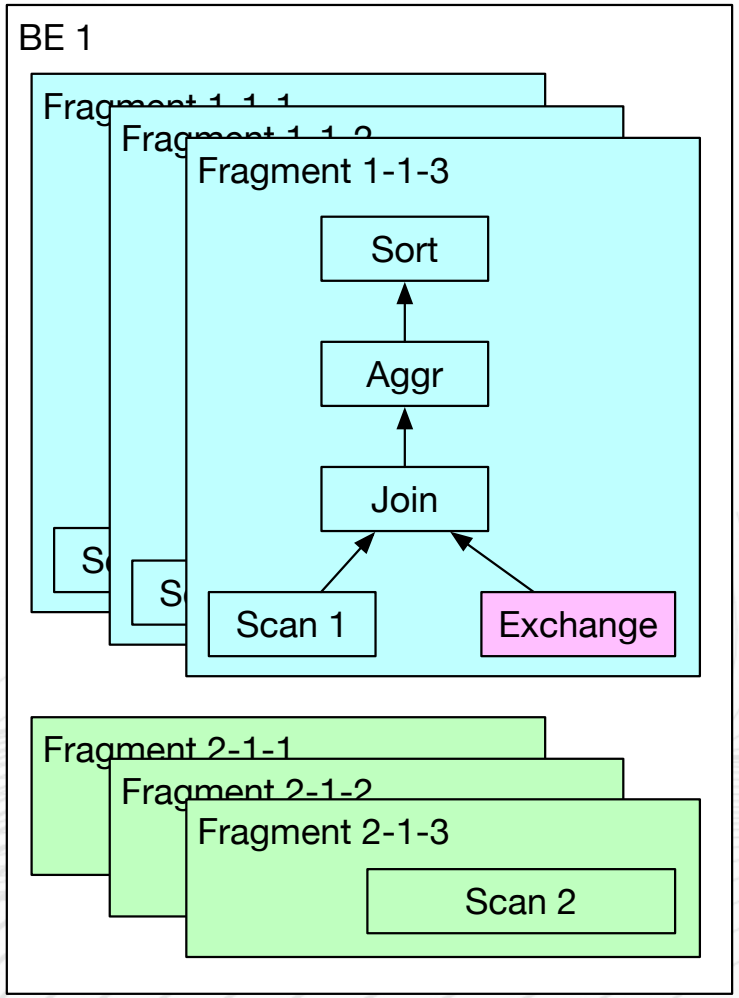
- 节点间并行



查询引擎

并发执行模型

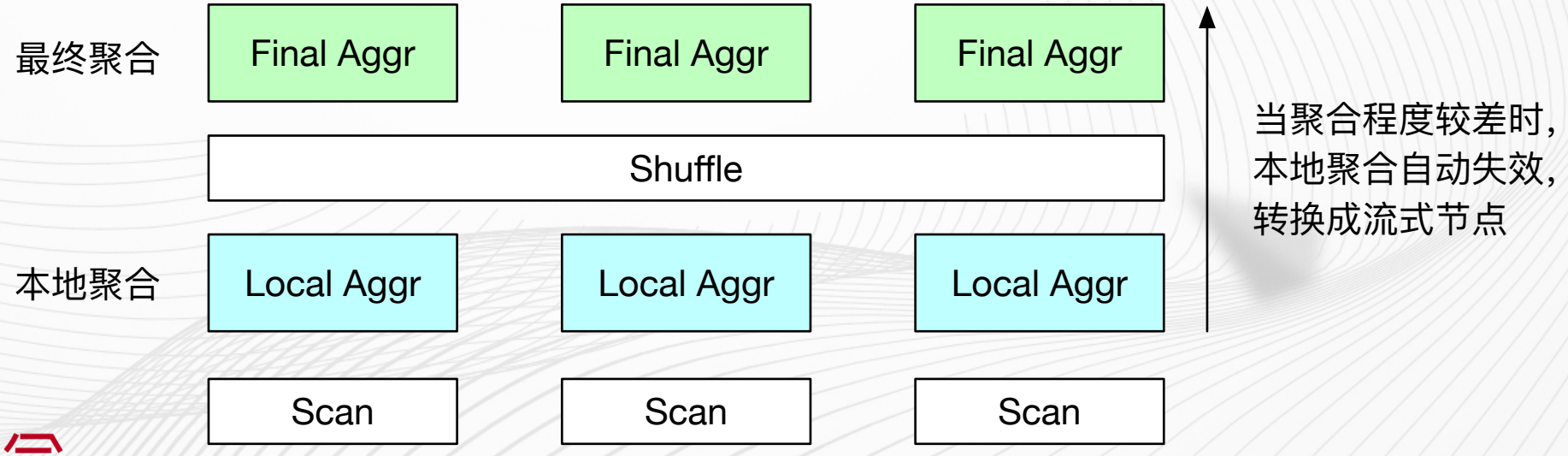
- 节点内并行



查询引擎

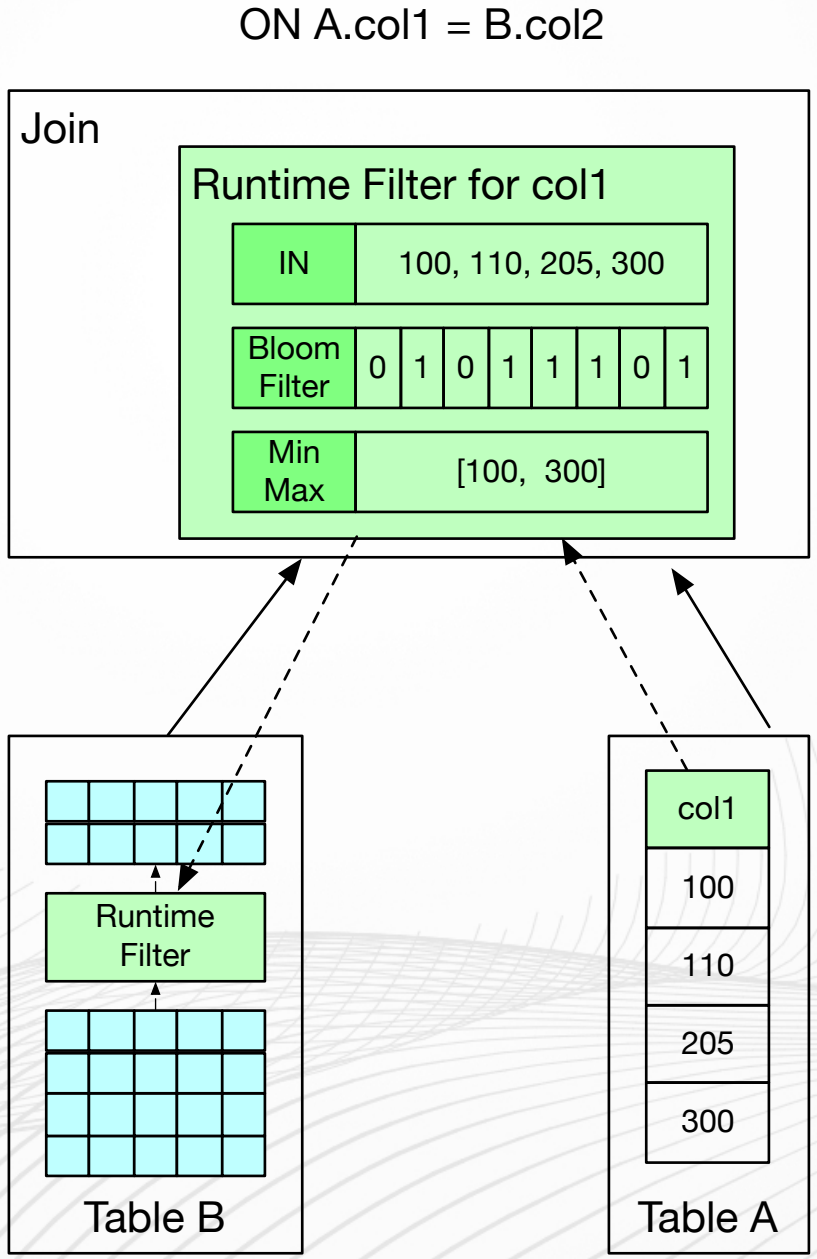
自适应的两阶段聚合

- 第一层先聚合，减少数据传输
- 自动感知聚合程度，减少阻塞时间



Runtime Filter

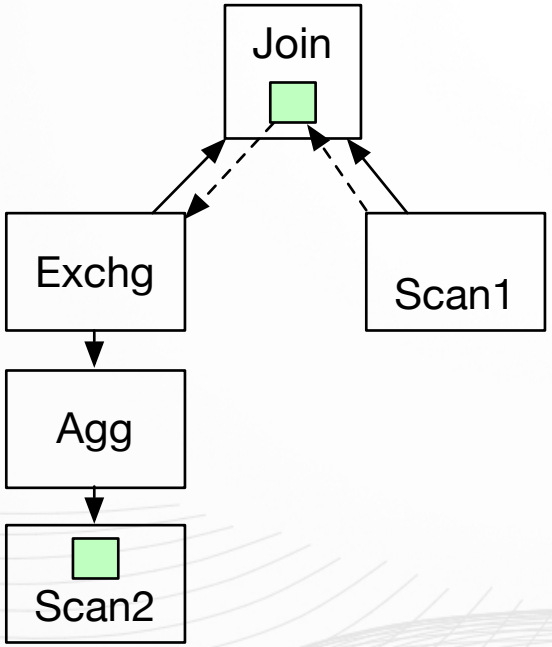
- 减少Probe侧计算量

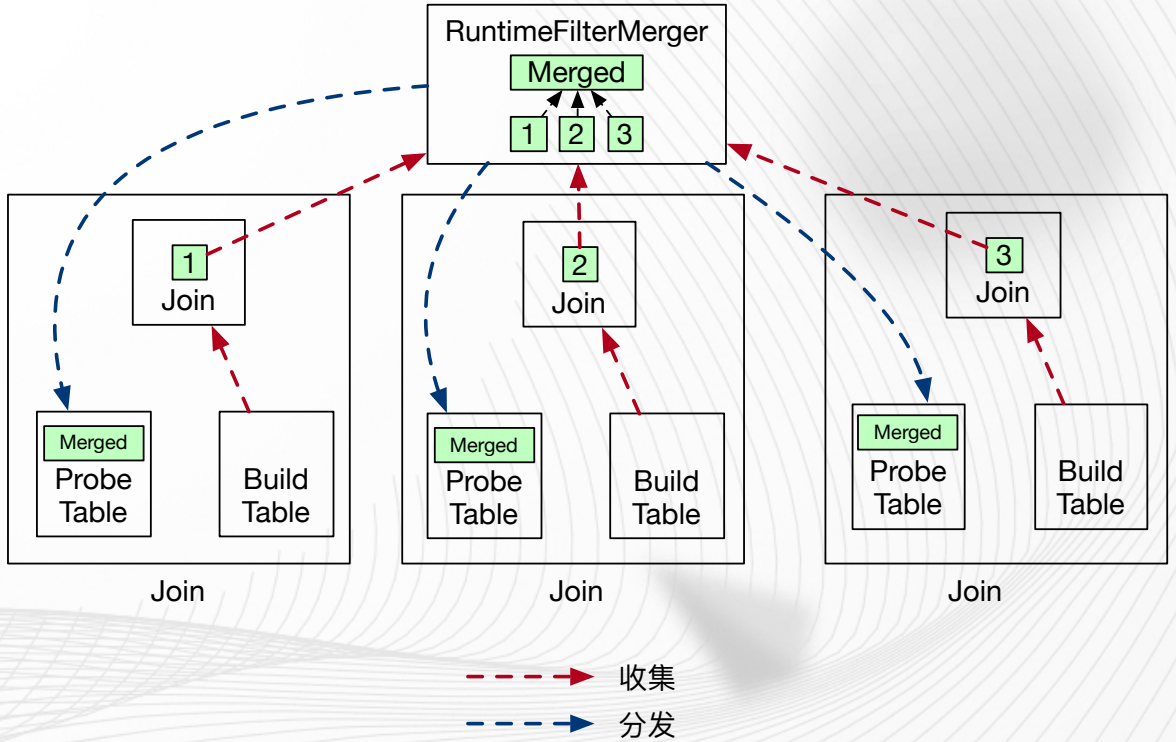


查询引擎

Runtime Filter

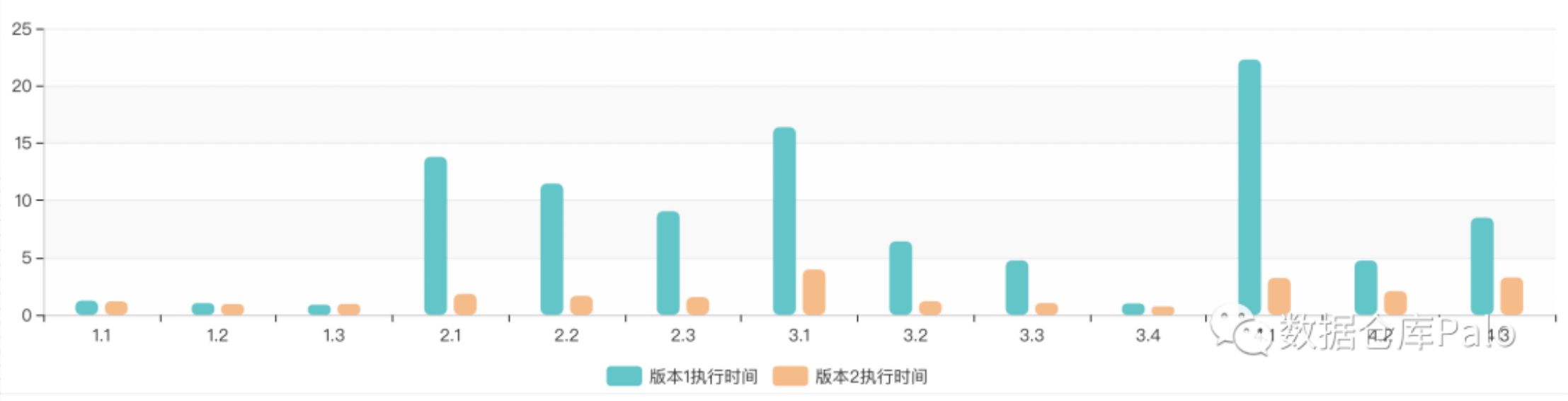
- 自动穿透
- Runtime Filter 合并





查询引擎

Runtime Filter



查询优化器

基于规则的优化器（RBO）

- **常量折叠**
- 子查询改写
- 提取公共表达式
- 智能预过滤
- 谓词下推

```
WHERE event_date > str_to_date("2020-09-01", "%Y-%m-%d %H:%i:%s")
```

↓

```
WHERE event_date > 2020-09-01 00:00:00
```

查询优化器

基于规则的优化器（RBO）

- 常量折叠
- 子查询改写
- 提取公共表达式
- 智能预过滤
- 谓词下推

```
SELECT * FROM tbl1
WHERE
col1 IN (SELECT col2 FROM tbl2) a
```

```
SELECT tbl1.* FROM tbl1 JOIN tbl2 on tbl1.col1 = tbl2.col2;
```

查询优化器

基于规则的优化器（RBO）

- 常量折叠
- 子查询改写
- **提取公共表达式**
- 智能预过滤
- 谓词下推

```
SELECT * FROM tbl1
WHERE
(a > 1 AND b = 2) OR (a > 1 AND b = 3) OR (a > 1 AND b = 4)
```



```
SELECT * FROM tbl1
WHERE
(a > 1) AND (b =2 OR b =3 OR b =4)
```

查询优化器

基于规则的优化器（RBO）

- 常量折叠
- 子查询改写
- 提取公共表达式
- 智能预过滤
- 谓词下推

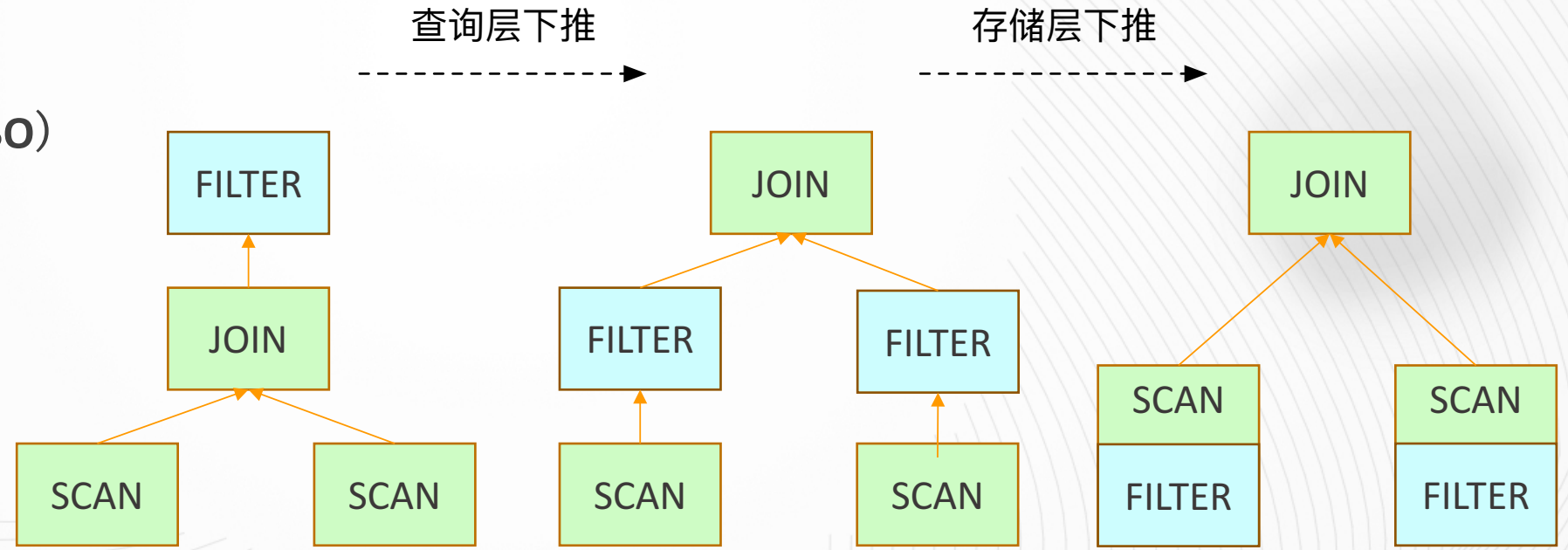
```
SELECT * FROM tbl1
WHERE
(1 < a <3 AND b IN ('a') ) OR (2 < a < 4 AND b IN ('b'))
```

```
SELECT * FROM tbl1
WHERE
(1 < a < 4 ) AND (b IN ('a', 'b'))
AND ((1 < a < 3 AND b IN ('a') ) OR (2 < a < 4 AND b IN ('b')))
```

查询优化器

基于规则的优化器（RBO）

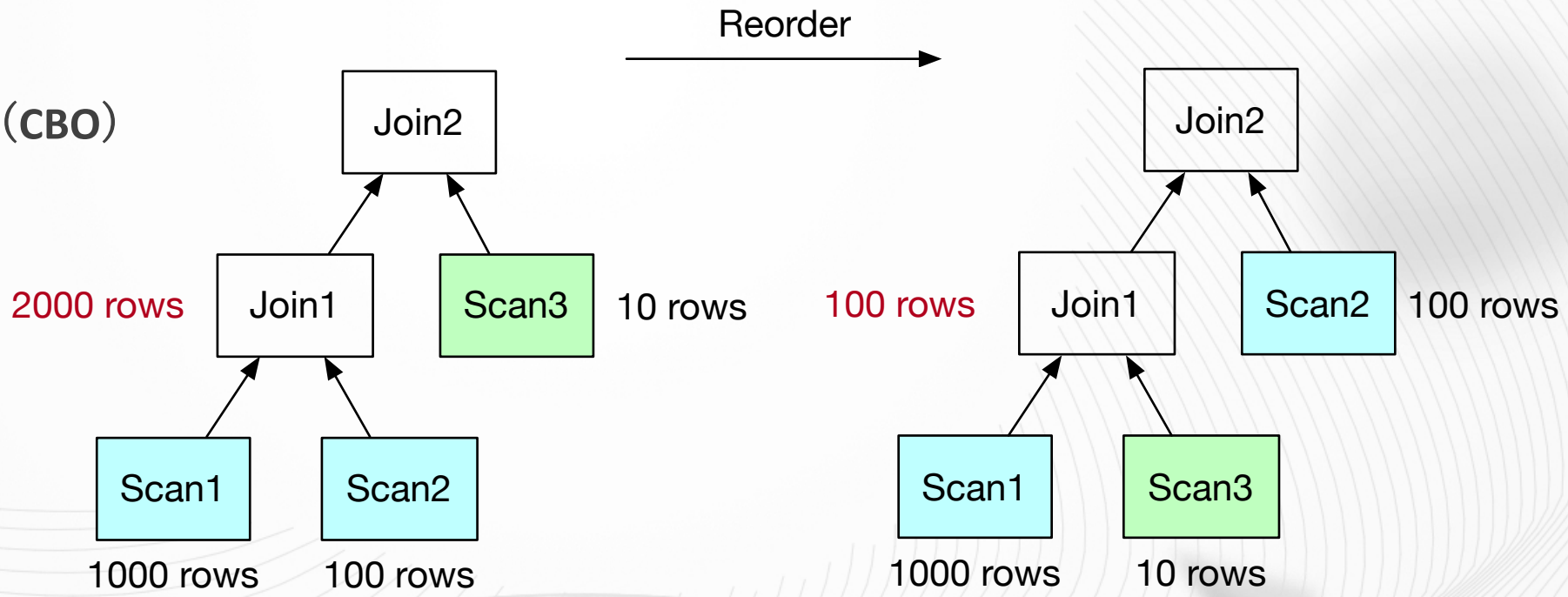
- 常量折叠
- 子查询改写
- 提取公共表达式
- 智能预过滤
- **谓词下推**



查询优化器

基于代价的优化器（CBO）

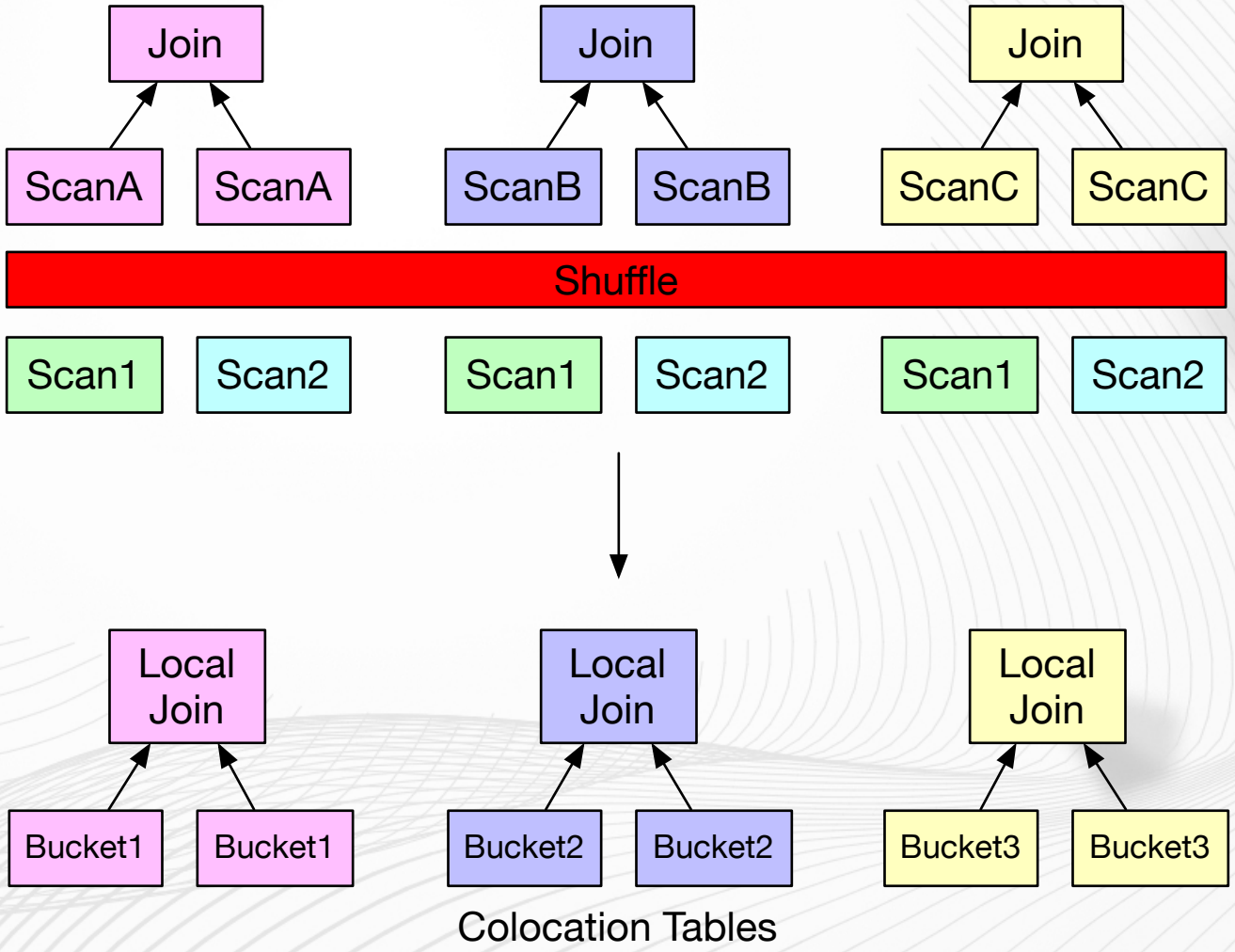
- **Join Reorder**
- Colocation Join
- Bucket Join



查询优化器

基于代价的优化器（CBO）

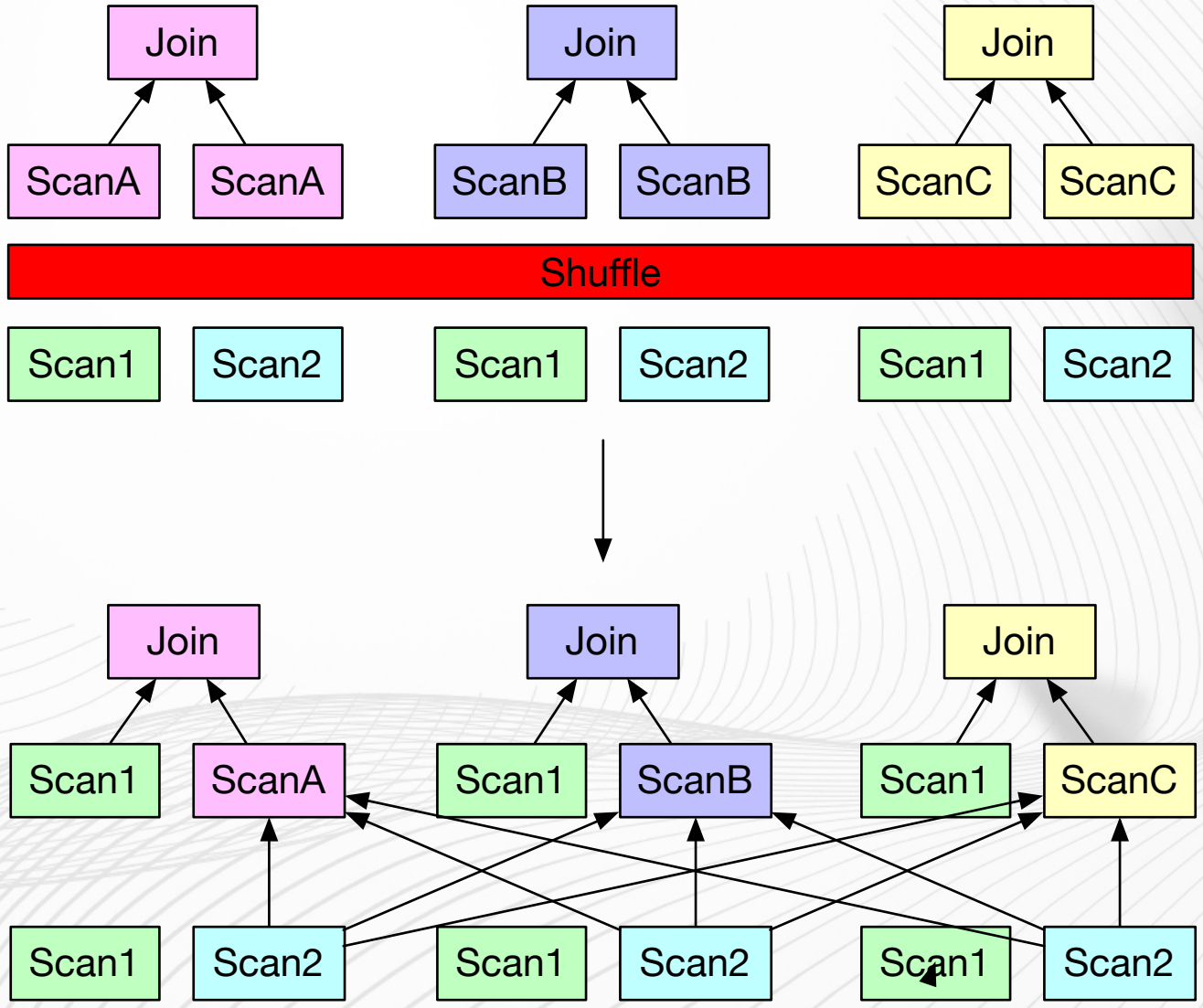
- Join Reorder
- **Colocation Join**
- Bucket Join



查询优化器

基于代价的优化器（CBO）

- Join Reorder
- Colocation Join
- **Bucket Join**



Bucket Shuffle

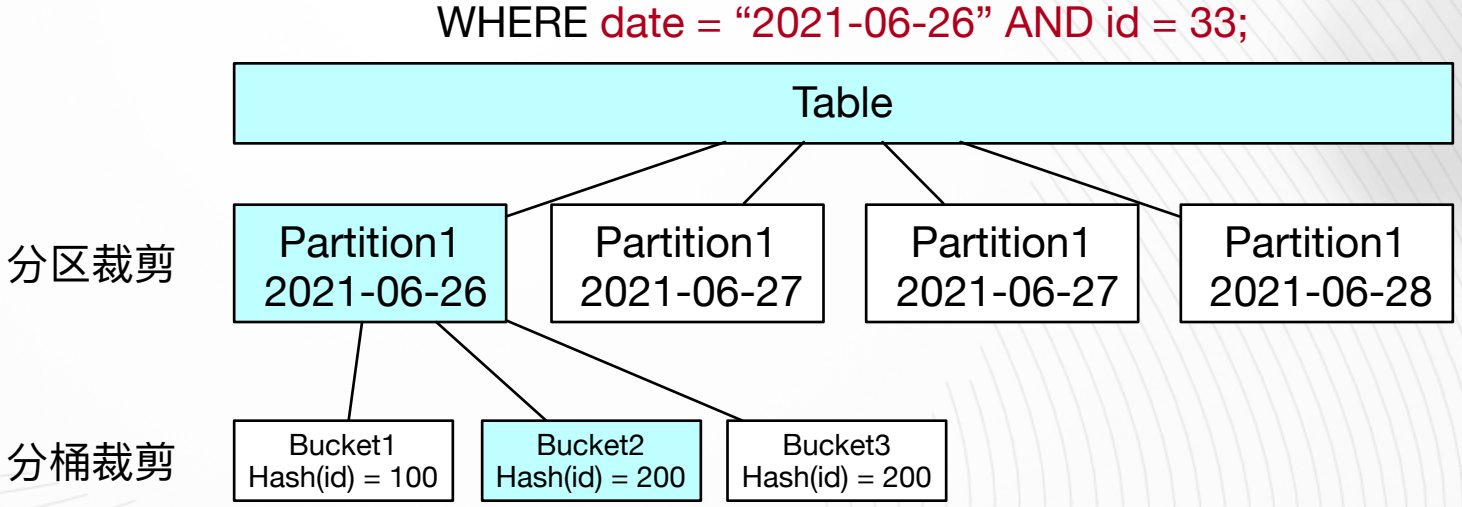
GOTC

3 功能丰富

- 高并发？Adhoc？
- 精确去重？
- 用户画像？漏斗分析？
- 预聚合Cube？
- 数据更新？

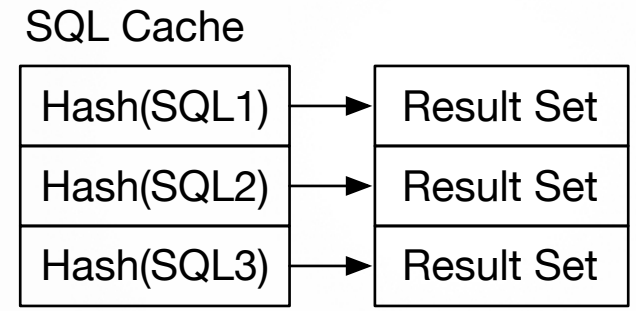
高并发查询

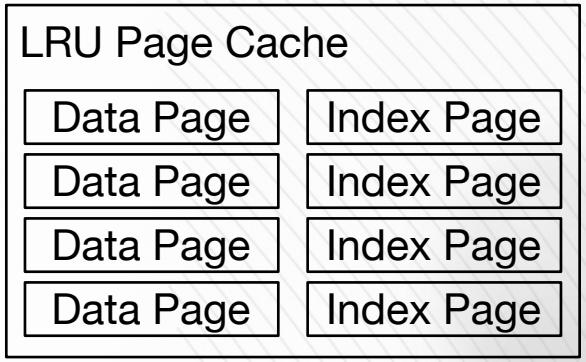
- 分区、分桶裁剪
- SQL/Partition/Page Cache

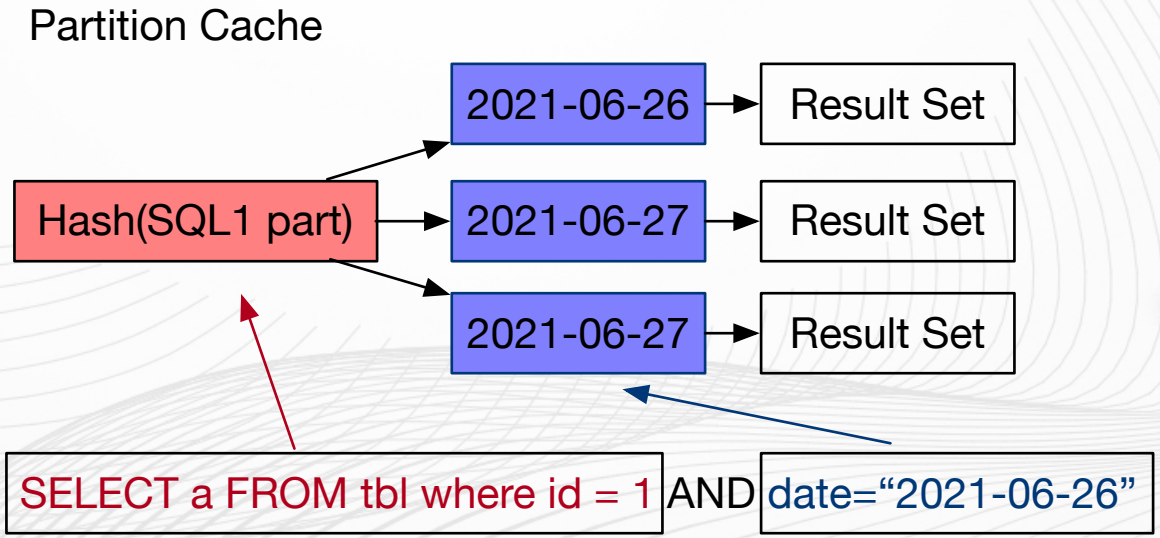


高并发查询

- 分区、分桶裁剪
- **SQL/Partition/Page Cache**







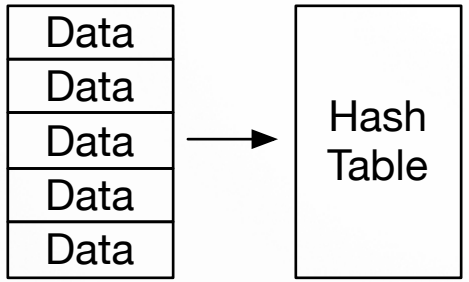
GOTC

Adhoc 和库内 ETL

- 复杂SQL支持
- 窗口函数、Grouping Set 等高级语法
- UDF、UDAF

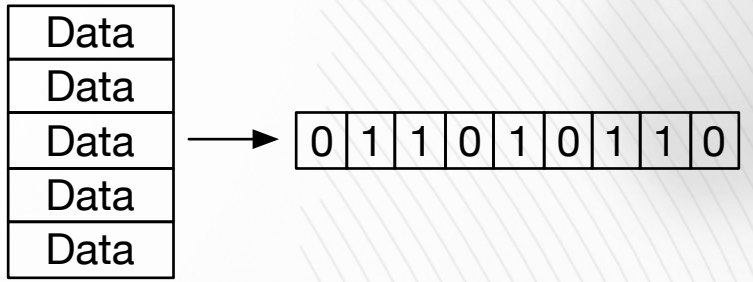
Bitmap 数据类型

- 高基数精确去重
- 用户画像
- 漏斗分析



Count Distinct
实时构建Hash表

计算量大
内存开销大

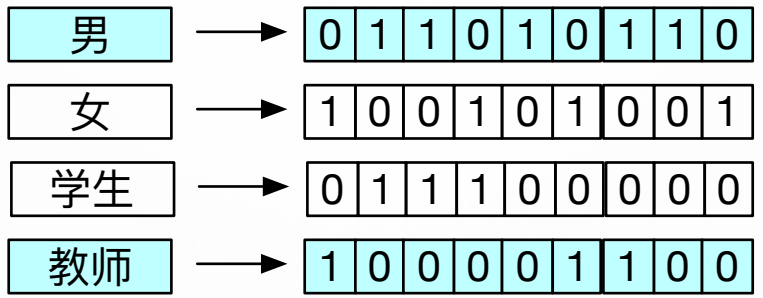


存储为Bitmap
求1的个数

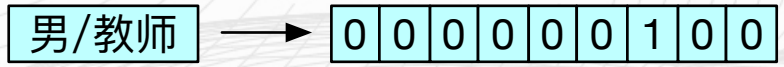
速度快
内存开销可控

Bitmap 数据类型

- 高基数精确去重
- **用户画像**
- 漏斗分析



求男性教师的人群



GOTC

Bitmap 数据类型

- 高基数精确去重
- 用户画像
- 漏斗分析

```
SELECT
intersect_count(user_id, page, 'login') as login_uv,
intersect_count(user_id, page, ‘browse’) as browse_uv,
intersect_count(user_id, page, ‘login’, ‘browse’) as retention
FROM web_log
WHERE page in (‘login’, ‘browse’);
```

物化视图

- 明细 + 聚合统一模型
- 数据一致
- 查询自动路由

ID	日期	城市	消费
1	2021-06-26	北京	100
2	2021-06-27	北京	200
3	2021-06-27	上海	300

日期	SUM(消费)
2021-06-26	100
2021-06-27	500 (200+300)

日期	COUNT_DISTINCT(ID)
2021-06-26	1
2021-06-27	2

城市	MAX(消费)
北京	200
上海	300

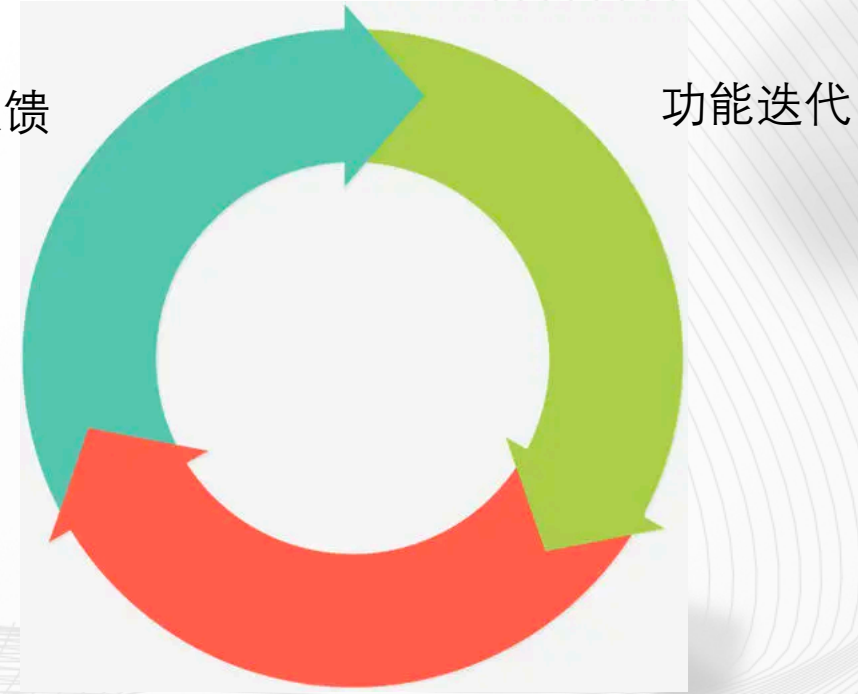
GOTC

数据更新

- 基于主键的更新
- 部分列更新
- **Update** 语句支持
- 同步业务数据库变更：**Marked Delete + Sequence Column**

4 开源开放

- 场景获取
- Bug修复
- 用户反馈
- 互助互利



GOTC

03

Doris 的进行时和未来式

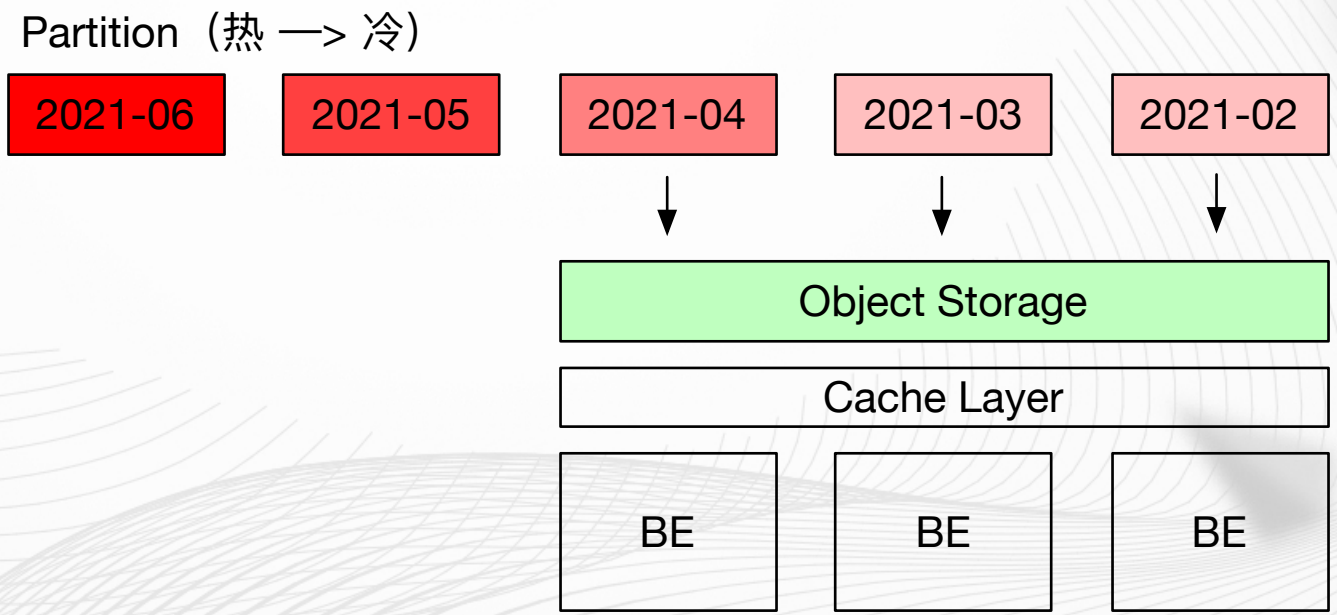
GOTC

Doris 进行时 – 性能！还是性能！

- 向量化引擎
  - Cache 亲和度、虚函数调用、分支预测、SIMD指令集
- 内存管理
  - 内存对齐、内存控制、线程间竞争、HugePage

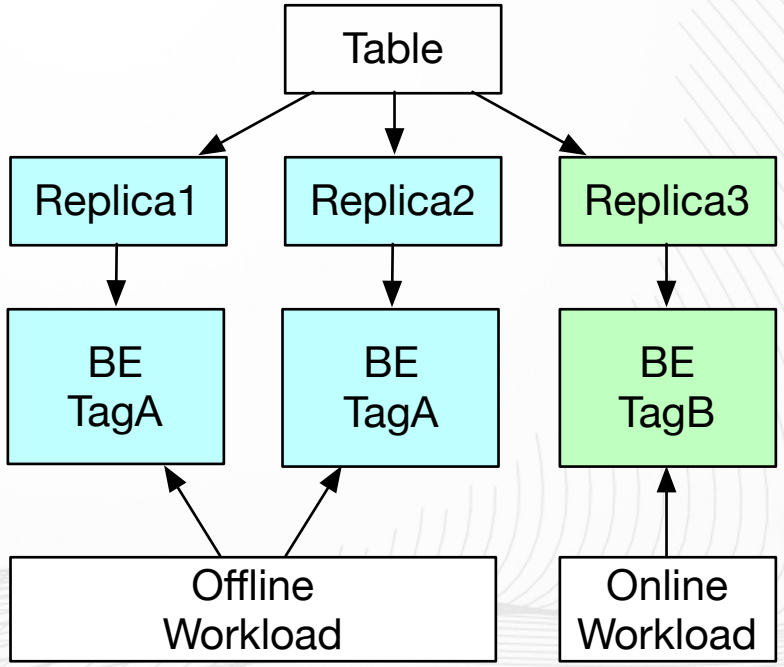
Doris 进行时 – 更低的成本

- 深存储和冷热数据分级



Doris 进行时 – 更好的隔离

- 资源划分
- 一个集群，两个场景



GOTC

Doris 未来式 – 湖仓一体和云原生

- Iceberg、Flink、Pulsar
- 存算分离：更好的弹性

GOTC

Doris 版本说明

- Apache Doris 官方版本
  - https://github.com/apache/incubator-doris
  - http://doris.incubator.apache.org/
- Palo 发行版
  - http://palo.baidu.com/home

ApacheDoris微信公众号

GOTC

THANKS

全球开源技术峰会

THE GLOBAL OPENSOURCE TECHNOLOGY CONFERENCE