DTCC

第十一届中国数据库技术大会

DATABASE TECHNOLOGY CONFERENCE CHINA 2020

架构革新 高效可控

IT168.com ChinaUnix ITPUB

北京国际会议中心 | 2020/12/21-12/23

基于Spark如何实现万亿大数据的即席检索与分析

南京录信软件技术有限公司

2020/12

个人简介

录信创始人 母延年

1.2008年新浪开始研究lucene,lucene的资深粉丝

2.2011年阿里巴巴开源Mdrill项目:5000亿数据

3.2014年腾讯Hermes项目:每天10000亿+数据

4.2018年作为lucene的粉丝,以lucene的发音,创办录信数软

5.个人域名： http://www.lucene.cn

Spark已经很快了，我们还能做些什么?

Spark 好比是一辆高性能的跑车

再好的跑车-也需要良好的道路来支撑（Parquet就比Txt快）

我们旨在万亿级数据量的这个领域里给Spark铺一条更好的路。

1：路下的基础，路基的材料如何选择？如何抗冲击！

2：路上的环境，路的形状，宽度，弯度？跑的更快！

我们铺的这条路具体什么样?

每天万亿

要做就做业界
数据规模最大的数据库

秒级响应

常规查询毫秒级响应
统计分析秒级响应

全栈能力

一份数据、一套系统
一种接口、90%场景

实时高并发

数据毫秒级延迟
每秒亿级的查询与更新

低成本

廉价硬件，少SSD
大硬盘，省节点

多表关联

业界老大难的问题
技术上该如何突破?

Spark这辆跑车在我们铺的这条路上能玩出什么花样

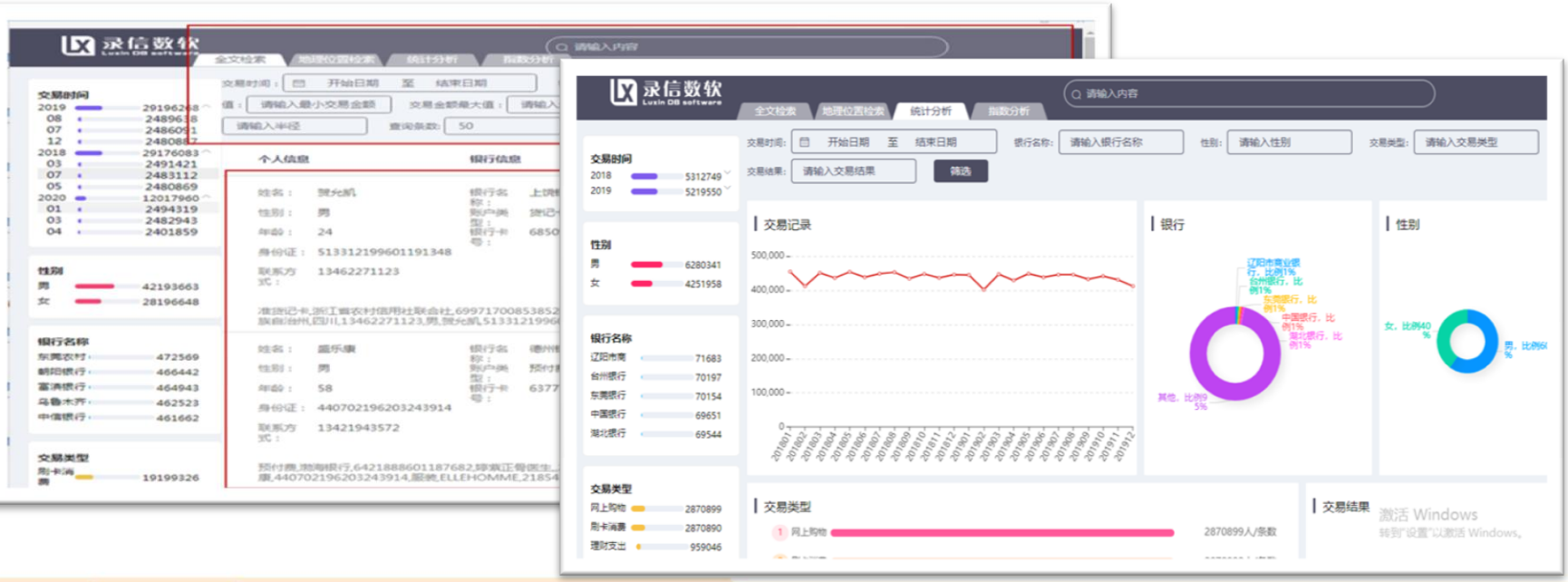
----业务场景介绍

线下系统：一个巨大的实时数仓、万亿秒查

录信数软 Luxin DB software

1.互联网早已落地多年；公安、军队、国安、情报分析已大规模使用

2.汽车、银行、高铁，数据开始增大； 物联网、5G、工业能源、民用设备基础设施逐步健全



线上系统-万亿级的高并发数据库

旨在替代

1：传统数据库分库分表方案，如oracle与mysql等。

2：分片数固定或HASH方式分片的系统，
如SOLR/ES的全文检索，部分KV系统等。

特点：

1：每秒亿级别的写入与查询的能力

2：数据分片弹性可伸缩,改变分片数无需重新导数据，秒级分裂扩展，可以应对爆炸式的增长。

1.网站应用->支付宝的账单，店铺，贴吧
2.物联网应用->车辆分析、电表、某设备的监控
3.银行，保险，运营商->企业与个人的风控、账单
4.无数的APP小程序的应用->个人中心
5.智慧城市->车站调度、城市大脑

这条路具体是怎么铺设?

----核心技术与实现原理

整体思路-基于 Spark 的索引方法(关键技术一)

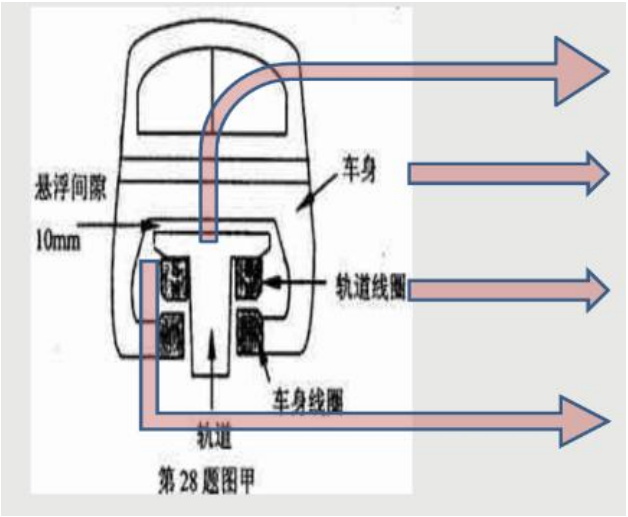
一个带有
索引的Spark
大数据
OLAP系统

查询的时候借助索引，
避免了对数据的暴力扫描，
查询与分析性能直接提升了100以上

- ✓ 大幅度的加快数据的检索速度。
- ✓ 显著减少查询中分组、统计和排序的时间。
- ✓ 大幅度的提高系统的性能和响应时间，从而节约资源。

修正了大量的开源Spark的BUG，
趟平开源Spark在生产系统中
出现的各种问题

不同需求不同索引格式



4：车上的指挥调度，预先规划好线路：让用户能合理灵活的组织数据的分布。

3：根据路况动态更换轮胎：全栈数据库的设计，一张表多个索引，根据查询自动切换索引。上坡、下坡、转弯动态更换。

2：按路况铺不同的路：专用场景，专用索引。

1：最底层的地基：基于HDFS之上，撑万亿数据

最底层的地基：能否撑万亿数据的基础(关键技术二)

核心技术：
基于HDFS之上的分布式索引

为什么是我们：
十年的技术积累，业界首次尝试分布式索引
技术思路历经阿里Mdrill、腾讯Hermes实力验证，业界数据规模最大的数据库

成效是什么
目前有200余套生产系统在使用录信数据库
十余个超万亿规模的生产系统稳定运行

最底层的地基：撑万亿数据

1.使用本地文件系统的各种弊端。

2.基于本地文件系统的Mpp数据库会是未来么

3.Hadoop的慢真的没办法跨越么

4.业界常规通用的优化思路

5.录信独有的优化办法

全栈数据库的设计

一份数据、 一套系统、 一种接口

根据不同的场景定义多种不同的索引格式

不同的数据列不同的索引格式

根据SQL与catalog自动选择自适应的索引格式

相对于多个系统组合在一起的融合系统相比

节省大量的空间与机器资源.

系统内的数据可以互通，多个列之间的数据是一致的.

系统个数的减少运维和学习成本大大降低

一个由丰富的索引构成的全栈数据库

为什么可以做到

全文索引、倒排索引、异构索引、列簇索引、地理位置索引、层次索引，预计算索引、标签分析索引、多列联合索引、并且未来将是可第三方拓展的。

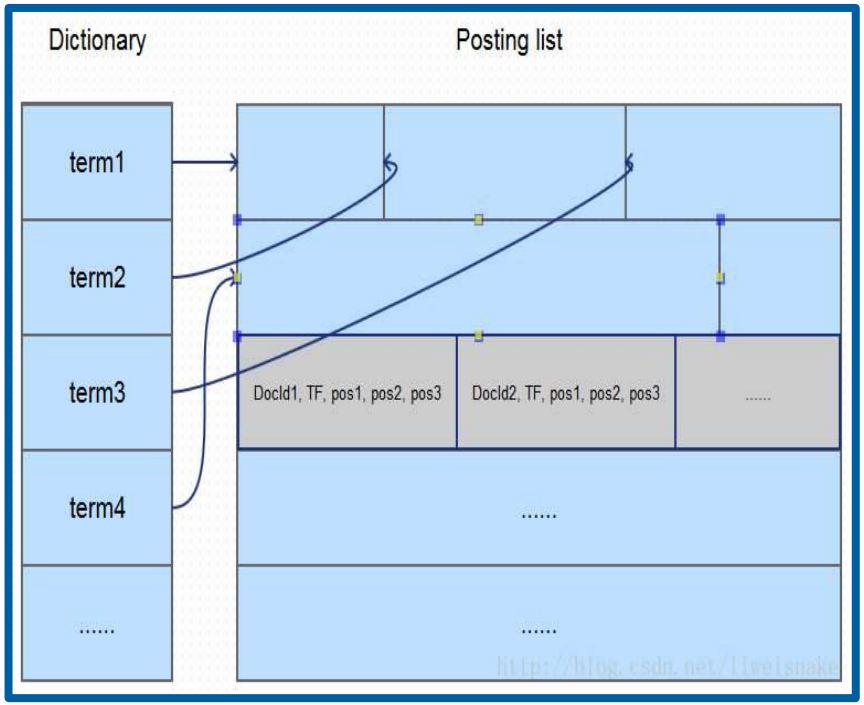
市场现状

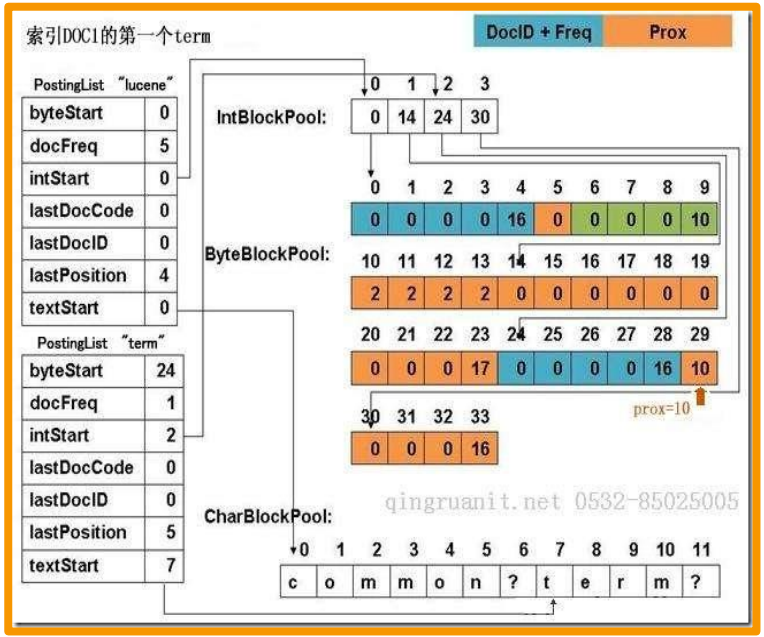
产品种类很多，但所有的计算都是“硬算”，在给数据加上索引这一领域市场还很空白

成效是什么

万亿秒查，而暴力扫描几百亿需半个小时以上。

全文检索-倒排索引：新华字典谁都用过





统计分析索引：统计分析、排序(关键技术三)

业界方案

暴力shuffle计算

我们的做法

lucene倒排索引与parquet嵌套列存储技术的组合

成效是什么

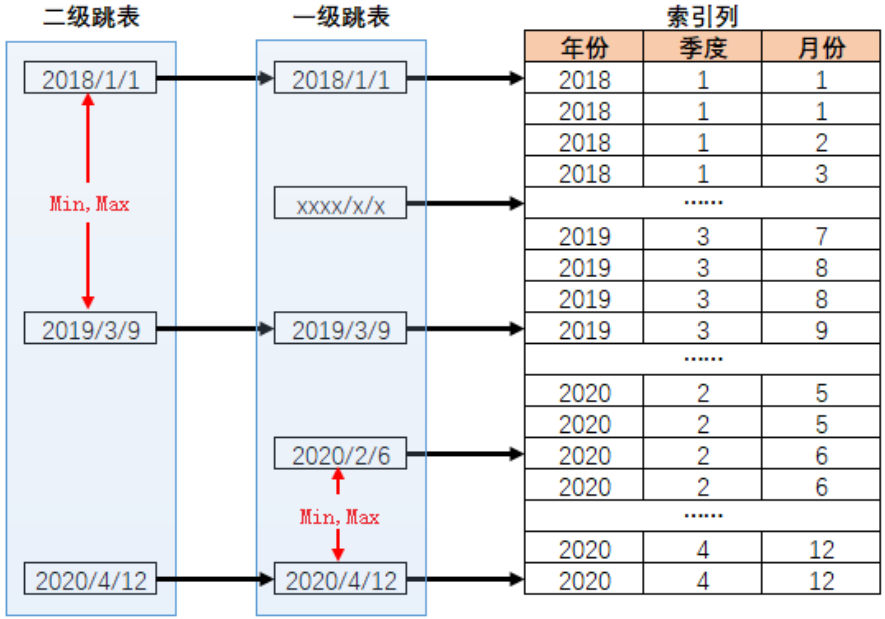
1:lucene倒排特性的干预数据分布,让parquet更高效。

2:parquet可以切片并行计算,改进lucene的不足

3: parquet多列特性，支持多列多层次筛选与导出

实时多维钻取分析领域则是录信的强项

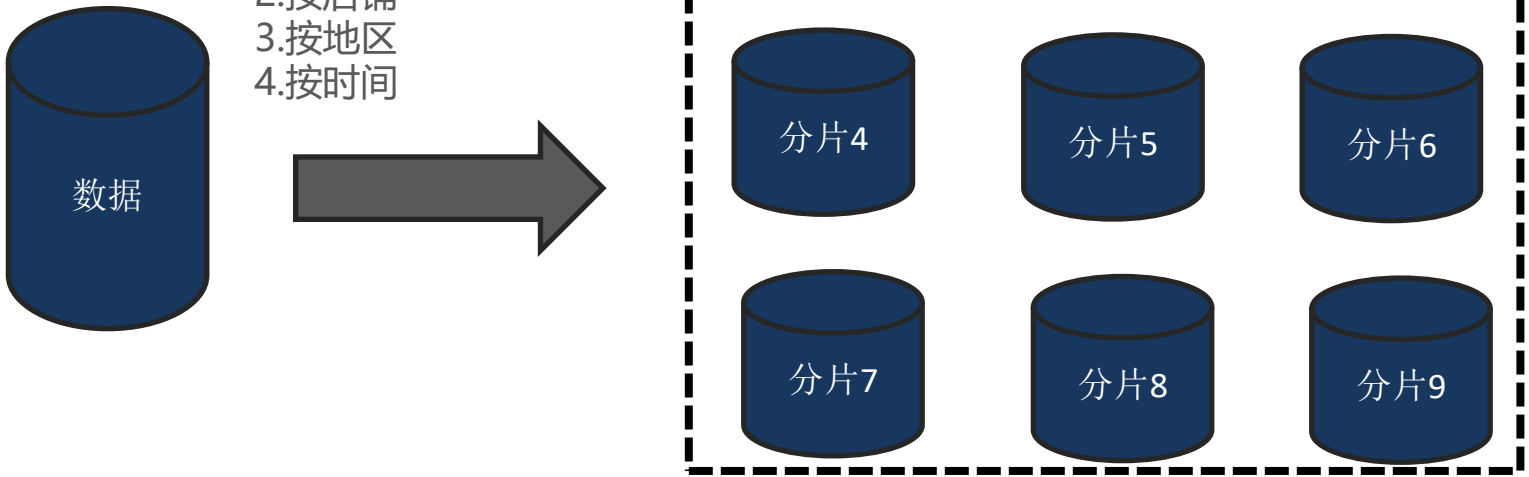
适合标签分析、用户画像、多列统计，数据导出等



高并发更新与查询-将数据按规则切分是最好的选择

录信数软
Luxin DB software

有很多种数据切分的方式
1.按用户ID hash
2.按店铺
3.按地区
4.按时间



传统数据切片存在的弊端

数据分片的瞬间分裂与收缩能力

一开始使用1个分片，随着数据量的变多自动分裂成2个、4个、8个....

无从判断一开始应该给这张数据表分多少个分片合适，分少了吞吐量不够，分多了资源浪费

一开始分的分片数量够，后来业务量爆炸增长分片数又不够了

某一个数据分片的数据分布不均衡

服务的瞬间迁移能力

据分片迁移过去了这些服务才能启动，还是将服务直接启动立即提供服务

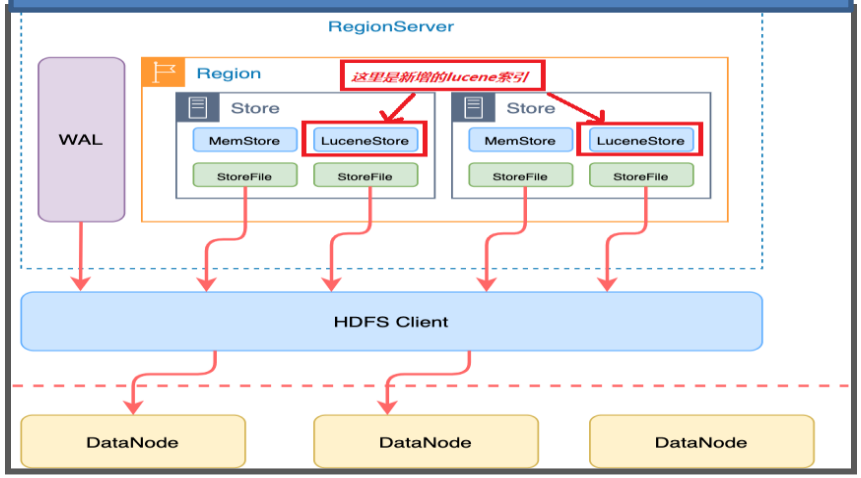
hash等规则的改变，涉及数据的重新分布，一般这类系统就要从新导入数据

本地盘存储的系统很难具备这样的能力

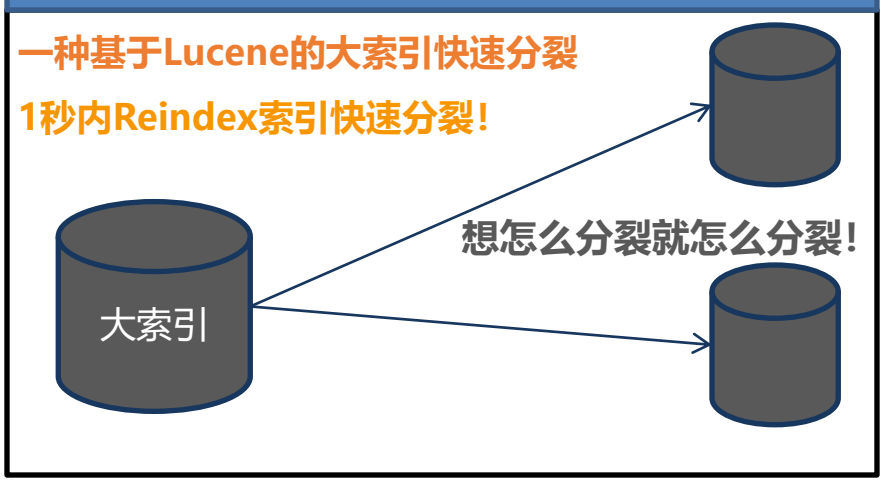
指挥调度，灵活的组织数据分布(关键技术四)

不重复造轮子:我们将Hbase的数据管理特性移植到Spark-Executor中,用于解决数据分片的弊端

1： 基于rowKey设计干预数据分布



2： RS可以灵活分裂与合并



节省成本方式方法(关键技术五)

想怎么存就怎么存

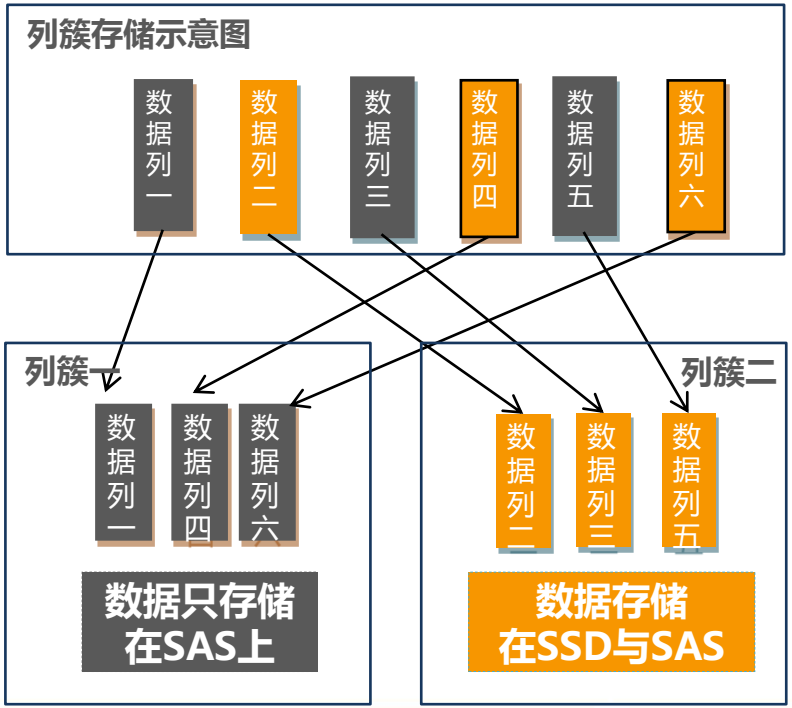
存储根据场景分类

列簇存储

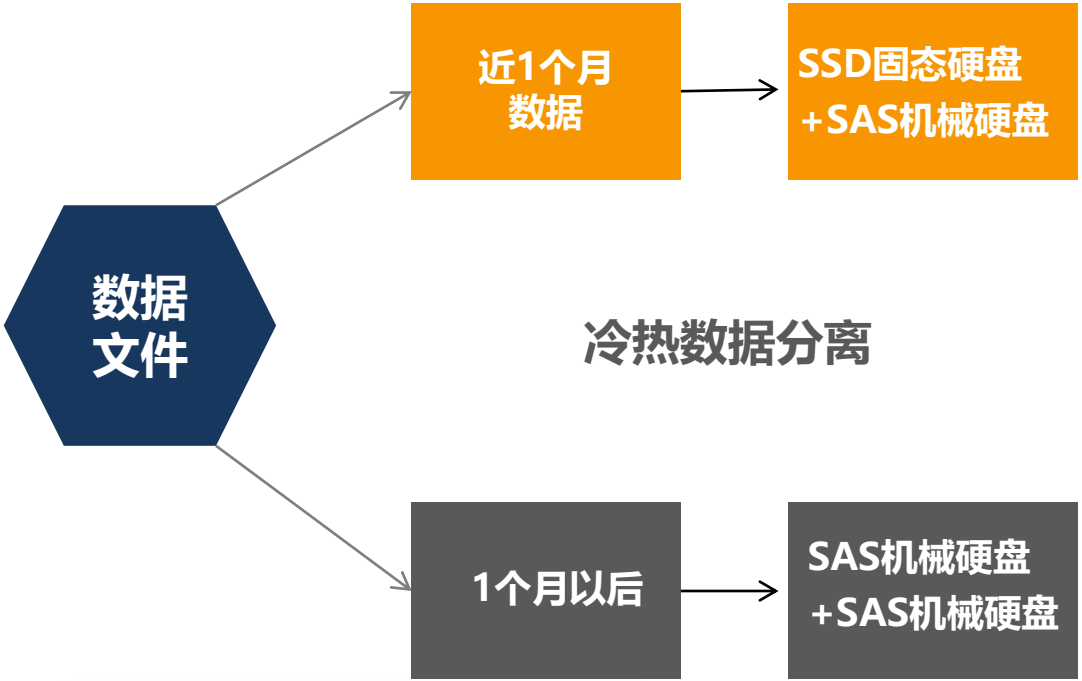
按列簇混合使用不同磁盘

按列簇设置不同存储生命周期

按列簇设置不同的索引格式：行存、列存、混合存储、联合索引



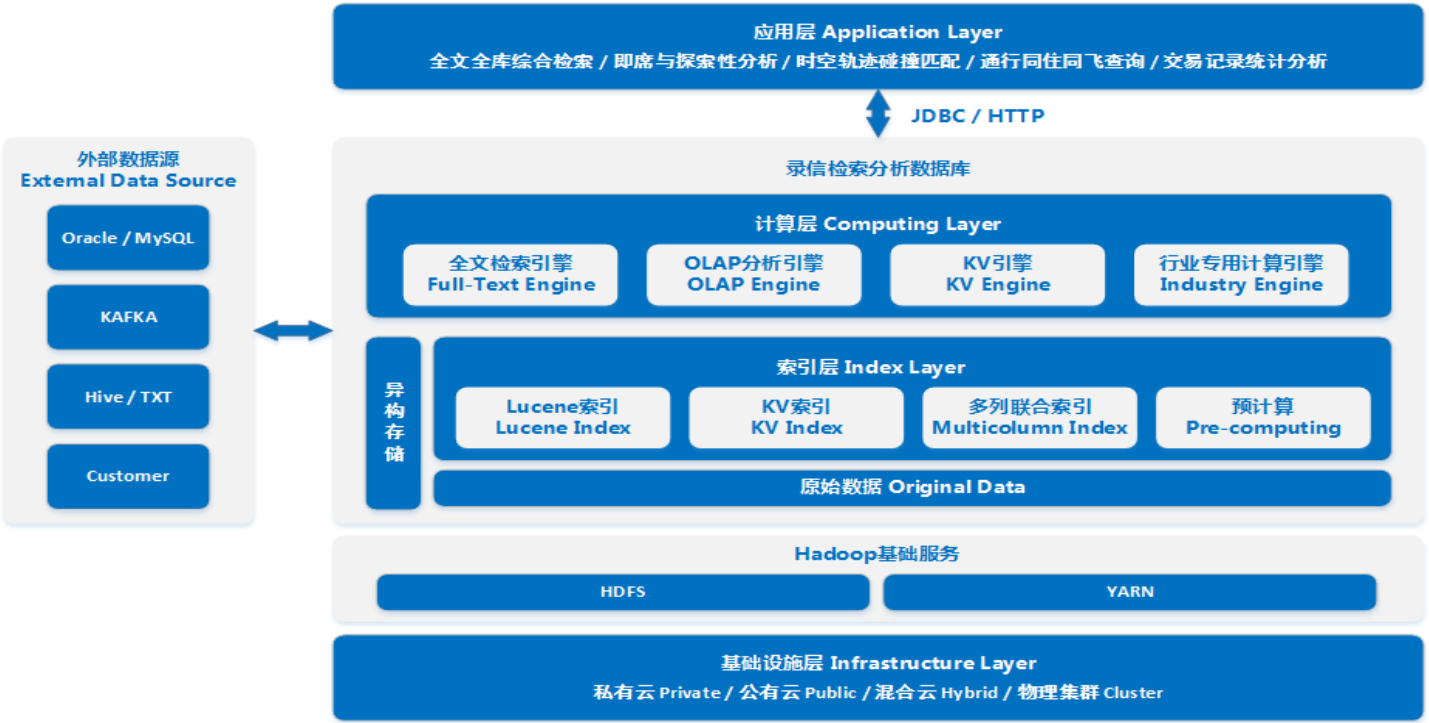
节省成本方式方法



索引存SSD
数据存SATA

SSD固态硬盘
-> 机械硬盘

系统架构


应用层 Application Layer
全文全库综合检索 / 即席与探索性分析 / 时空轨迹碰撞匹配 / 通行同住同飞查询 / 交易记录统计分析
JDBC / HTTP
外部数据源
External Data Source
Oracle / MySQL
KAFKA
Hive / TXT
Customer
录信检索分析数据库
计算层 Computing Layer
全文检索引擎
Full-Text Engine
OLAP分析引擎
OLAP Engine
KV引擎
KV Engine
行业专用计算引擎
Industry Engine
异构存储
索引层 Index Layer
Lucene索引
Lucene Index
KV索引
KV Index
多列联合索引
Multicolumn Index
预计算
Pre-computing
原始数据 Original Data
Hadoop基础服务
HDFS
YARN
基础设施层 Infrastructure Layer
私有云 Private / 公有云 Public / 混合云 Hybrid / 物理集群 Cluster

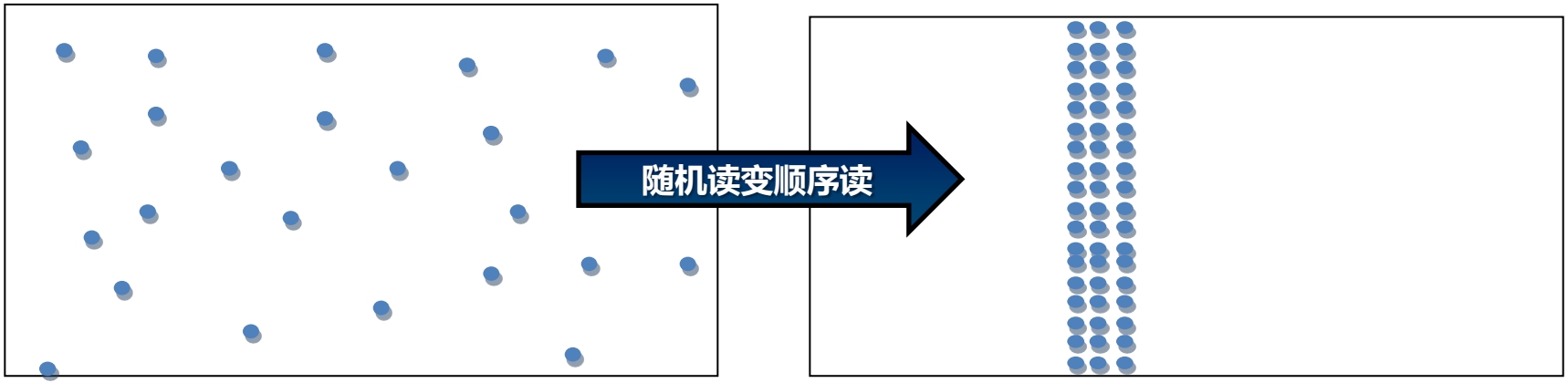
地理位置检索

1：我们抛弃docvalues方式的二次验证与剪切。

2：我们按照地理位置临近数据临近存储的方式存储数据，在此基础上进行二次验证

改造优势

- 大幅度的减少随机读取的次数，降低磁盘负载。
- 通过地理位置临近数据临近存储的方式构造硬盘上的连续读取，因常规磁盘的连续读取的性能会远远高于随机读写的性能，从而大幅度的提升查询响应的速度。



多表关联技术突破（尚未验证）

多表关联一直是业界的难题，目前都是通过暴力的”shuffle”实现的

通过分片的方式干预相同key在节点间的分布

通过索引技术让相同的key在同一个位置存储

基于索引的实现是基于lucene，而lucene的倒排则采用parquet的嵌套列存储实现

基于录信的多列联合索引技术，可以在物化关联的技术上，还能实现双表高效的筛选与过滤

录信数据库需要您的参与

THANKS

加入我们