大数据架构稳定性保障实践

顺丰科技有限公司

林国强 大数据架构负责人

引子

大数据发展至今，已经有近10年时间，在这10年时间里面，大数据架构发生了很多变化

而这些变化，不断冲击当前企业大数据架构，给业务部门和信息部门都来带很大挑战

200TB

每天新增

+

200PB+

存量规模

如此数据规模下，**如何保证大数据架构稳定性？**本次演讲将会分享顺丰科技大数据团队的相关实战经验

目录

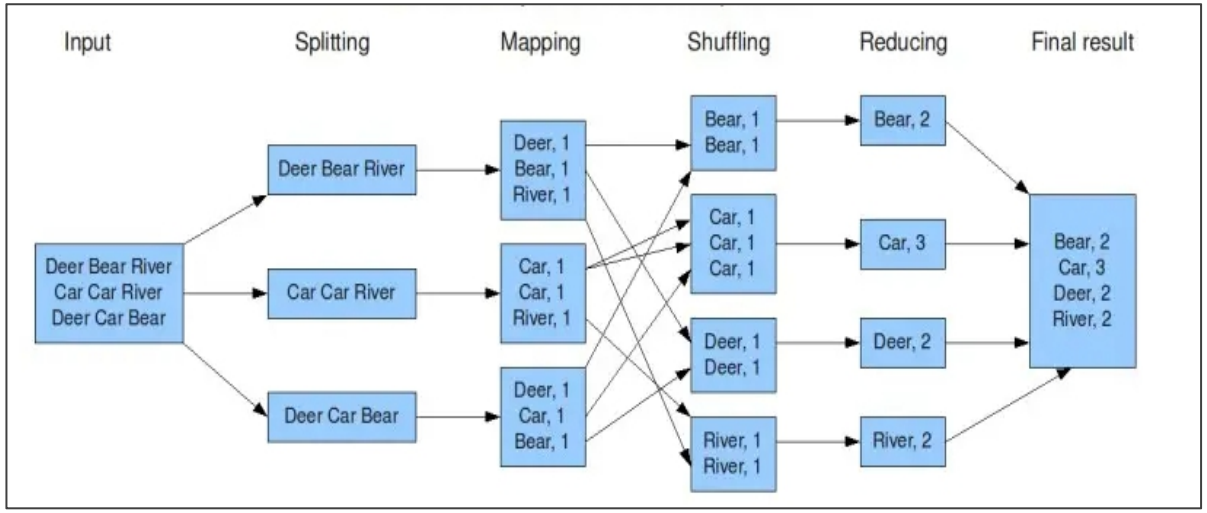
一、大数据架构历史变迁

洪荒期、远古期、近古期、近现代、现如今

大数据架构变迁-洪荒期&MR

MR原理

- Map/Reduce是一个用于大规模数据处理的分布式计算模型，它最初是由Google工程师设计并实现的，Google已经将它完整的MapReduce论文公开发布了
- 其中对它的定义是，**Map/Reduce是一个编程模型，是一个用于处理和生成大规模数据集的相关的实现。**用户定义一个map函数来处理一个key/value对以生成一批中间的key/value对，再定义一个reduce函数将所有这些中间的有着相同key的values合并起来。很多现实世界中的任务都可用这个模型来表达。

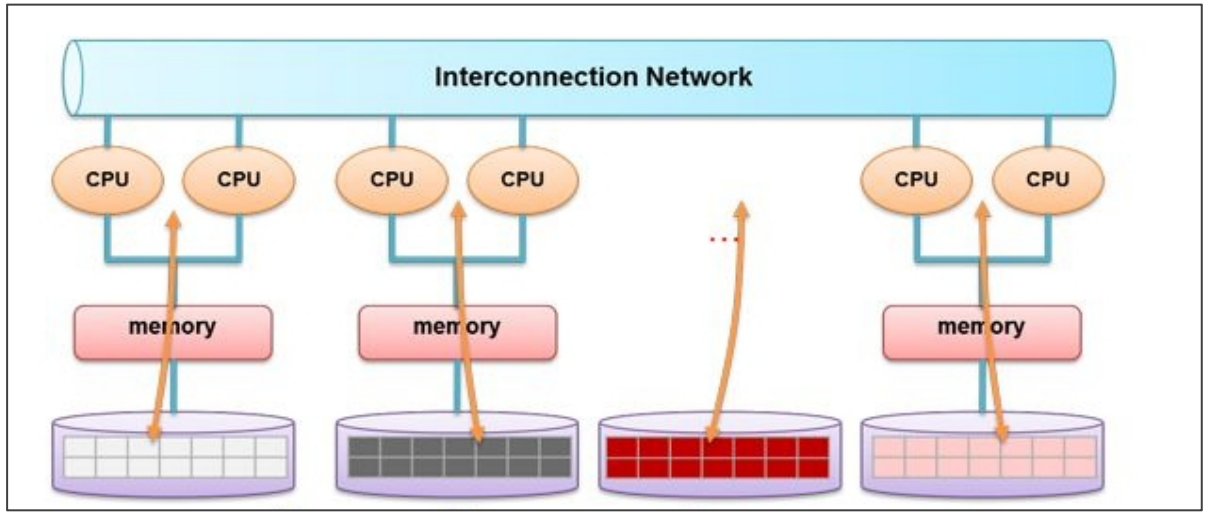


价值	• oracle、mysql、db2等传统数据库，无法处理海量数据，日增长100亿级，每天100TB左右的离线专题数据分析 • 引入hadoop mr架构解决离线跑批问题
变化	• Oracle存储全部需要改为MR/HSQL，重新编写后端调度
收益	• 公司解决了大规模数据分析问题，一部分员工因为解决了关键业务痛点，脱颖而出，成立最原始的大数据团队

大数据架构变迁-远古期&MPP

MPP原理

- MPP即大规模并行处理（Massively Parallel Processor ）。 每个节点都有独立的磁盘存储系统和内存系统，业务数据根据数据库模型和应用特点划分到各个节点上，每台数据节点通过专用网络或者商业通用网络互相连接，彼此协同计算，作为整体提供数据库服务。
- 非共享数据库集群有完全的可伸缩性、高可用、高性能、优秀的性价比、资源共享等优势

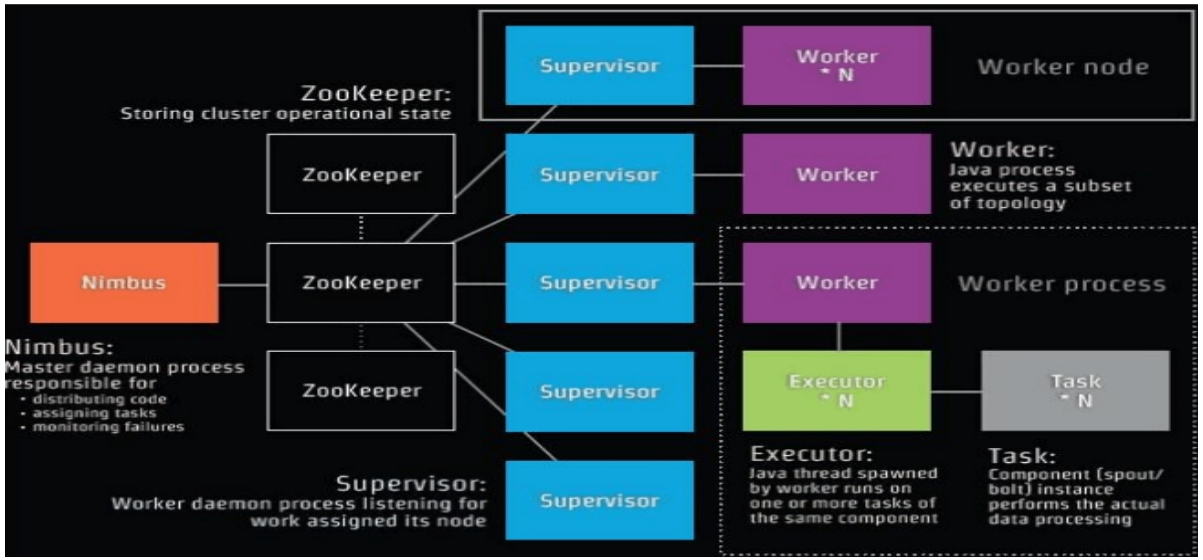


价值	• 架构简单， 端到端解决湖和仓的问题，在中小规模场景下，比较有优势，解决了原来hadoop架构响应速度和并发度问题（Scalability：100级别），并且开发人员只需掌握sql即可
变化	• 针对中小规模场景下，可以直接替换hadoop • 在大规模场景下，需要作为hadoop的后端输出承载，面向业务侧提供高价值数据分析
收益	• 公司使用更加简单架构，更简单的开发模式，应对不断变化的需求，一部分员工因为对mpp熟悉，独立一个团队，专注此项工作

大数据架构变迁-近古期&Storm

Storm原理

- Storm采用Master/Slave体系结构,分布式计算由Nimbus和Supervisor两类服务进程实现,Nimbus进程运行在集群的主节点,负责任务的指派和分发,Supervisor运行在集群的从节点,负责执行任务的具体部分。
- Nimbus： Storm集群的Master节点，负责资源分配和任务调度，负责分发用户代码，指派给具体的Supervisor节点上的Worker节点，去运行Topology对应组件（Spout/Bolt）的Task。
- Supervisor： Storm集群的从节点，负责接受Nimbus分配的任务，启动和停止属于自己管理的worker进程。通过Storm的配置文件中的supervisor.slots.ports配置项，可以指定在一个Supervisor上最大允许多少个Slot，每个Slot通过端口号来唯一标识，一个端口号对应一个Worker进程（如果该Worker进程被启动）

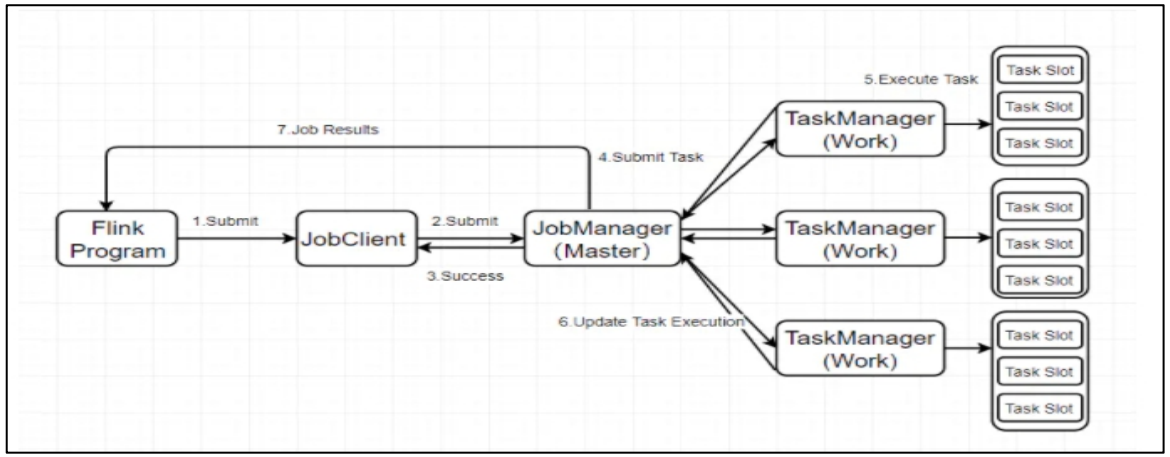


价值	• storm没出来之前，大家主要是写后端的预警程序，实现实时预警，需求响应时间长，且大规模场景下的处理非常复杂，storm之后，有一个相对好的架构，支撑实时流处理业务，能够更快速响应业务，处理海量实时数据
变化	• 需要把原先java、c、c++等编写的流处理程序，切换到storm，有一定的迁移工作，但是架构更稳定
收益	• 公司有更弹性、更简单的架构处理实时流数据，能更快速应对业务需求，同时，一部分员工因为对这部分比较熟悉，成立实时数据团队

大数据架构变迁-近现代&Flink/Spark

Flink原理

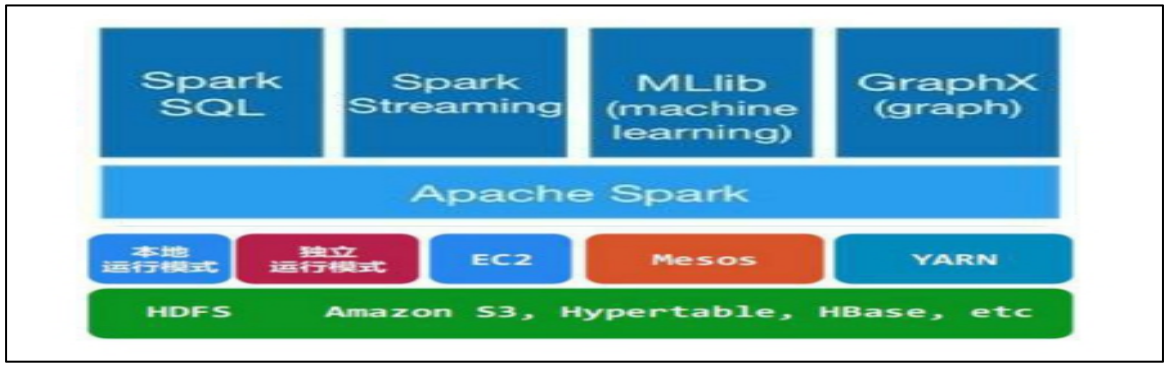
- Flink 是一个流处理框架，支持流处理和批处理，特点是流处理可容错、可扩展、高吞吐、低延迟。批处理是只有处理一批完成后，才会经过网络传输到下一个节点，流处理的优点是低延迟，批处理的优点是高吞吐



- **价值**：Flink比Storm的吞吐性能更强，具备一定的批处理能力，技术生态栈支持更广，架构更统一。
- **变化**：需要把基于storm编写的实时流处理程序，迁移至flink，改造量还是比较多
- **收益**：公司具备吞吐性能更强的流处理架构，基于flink能够做更多场景，如实时预测、实时TF；由原来实时流处理团队负责这部分架构

Spark原理

- Spark是一个围绕速度、易用性和复杂分析构建的大数据处理框架，最初在2009年由加州大学伯克利分校的AMPLab开发，并于2010年成为Apache的开源项目
- Spark基于内存的迭代计算框架，适用于需要多次操作特定数据集的应用场合。需要反复操作的次数越多，所需读取的数据量越大，受益越大

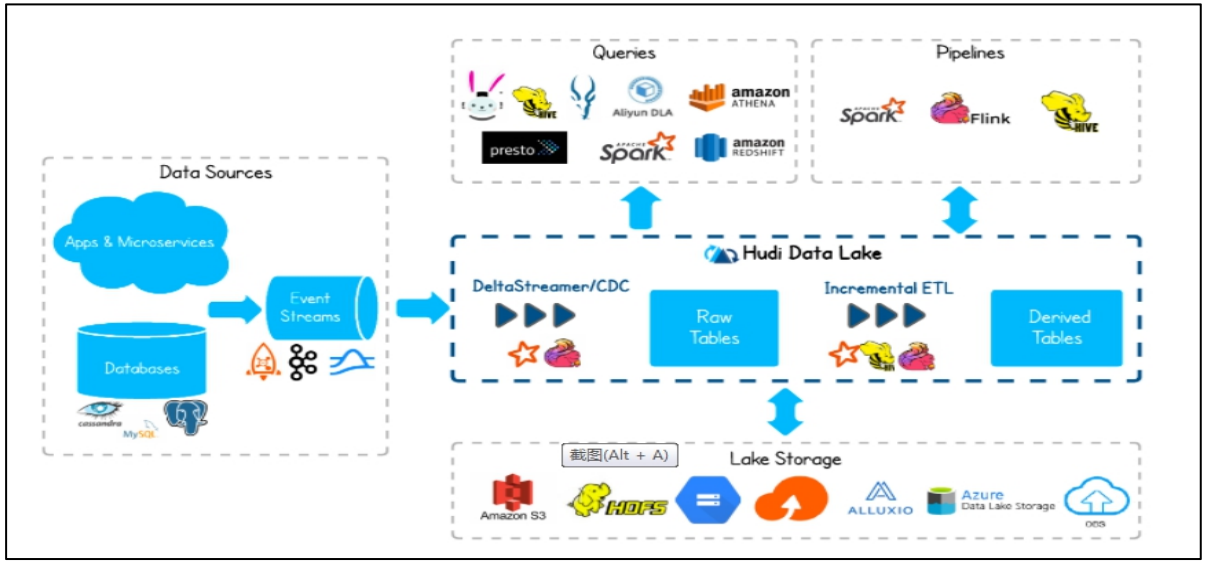


- **价值**：Spark相比Hadoop mr架构，计算过程不需要反复落盘，减少大量IO操作，大大提高计算速度。且技术生态栈较广，很好支持ML和流处理相关板块。
- **变化**：从HSQL迁移至Spark SQL，最开始时，还是需要不少工作量；
- **收益**：公司离线数据湖计算能力大致提高了2~3倍；成立一个新的算法团队，承担Spark计算框架业务

大数据架构变迁-现如今&实时数据湖架构

实时数据湖原理

- Hudi是**Hadoop Updates and Incrementals**的简写，它是由Uber开发并开源的Data Lakes解决方案，最初是用于解决数仓中 Lambda 架构中数据一致性的问题，将增量处理模型替代流式处理模型，并提供了 Upsert 和 Incremental Pull 两个非常重要的 feature
- Update/Delete记录：Hudi使用细粒度的文件/记录级别索引来支持Update/Delete记录，同时还提供写操作的事务保证。查询会处理最后一个提交的快照，并基于此输出结果。
- 变更流：Hudi对获取数据变更提供了一流的支持：可以从给定的时间点获取给定表中已updated/inserted/deleted的所有记录的增量流



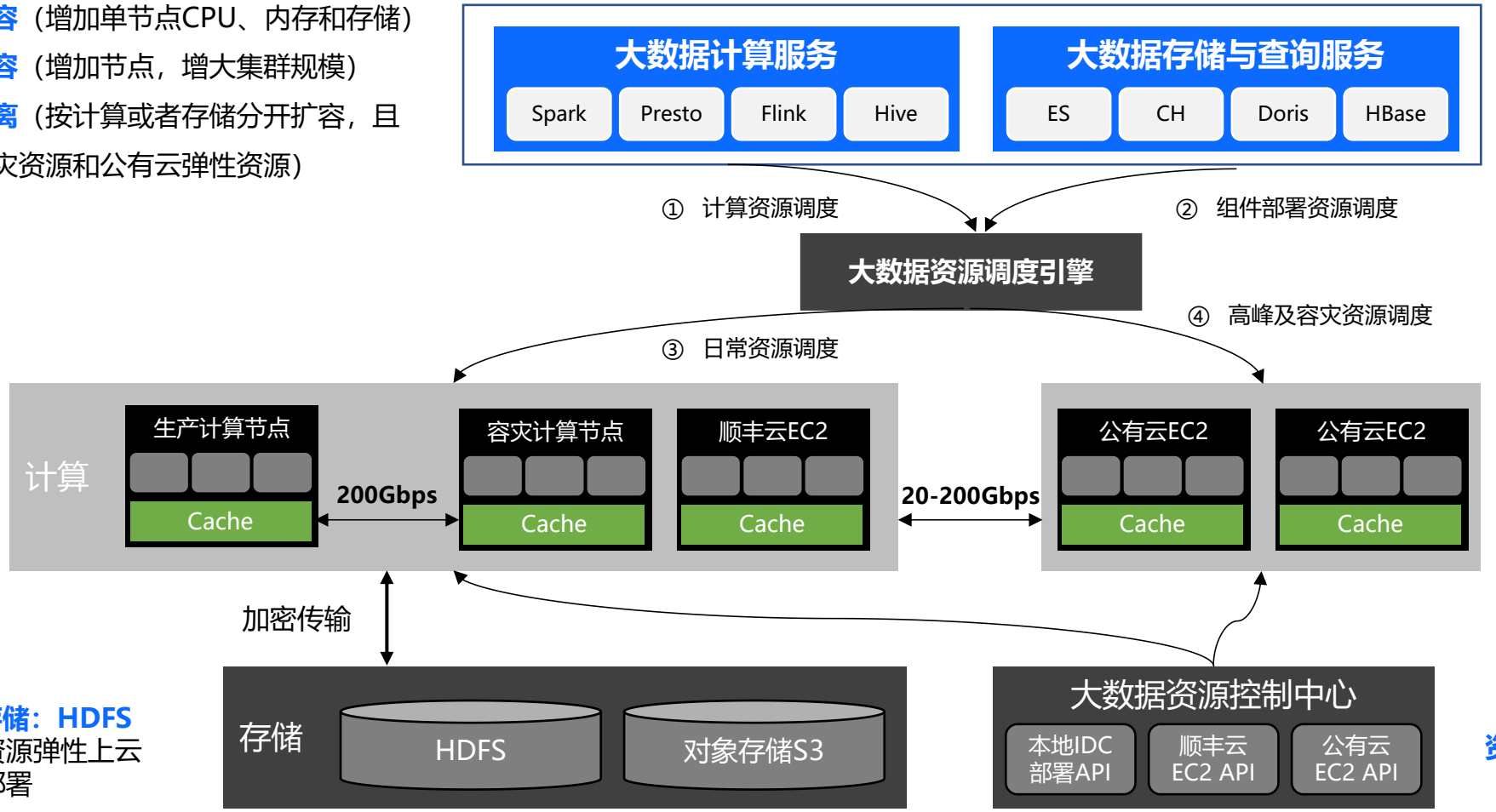
价值	• 解决了lambda架构指标一致性和资源重复投入问题，同时提高了指标分析时效性，提升了管理和运营的决策效率
变化	• 从hive/spark切换到hudi体系，会在数据接入侧需要进行调整，从overwrite切换为merge into，开发侧需要修改增量获取方式，代价不大，局部改动
收益	• **业务指标时效，从T+1天到T+0**，大大提升了指标时效，面向业务侧具备显性价值。同时，一部分员工因为比较熟悉，单独成立实时数据湖团队

二、架构稳定的关键因素

扩展性、可用性、自适性、易用性、先进性

架构稳定性关键因素-扩展性

- **纵向扩容**（增加单节点CPU、内存和存储）
- **横向扩容**（增加节点，增大集群规模）
- **存算分离**（按计算或者存储分开扩容，且复用容灾资源和公有云弹性资源）



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

跨机房缓存
- 统一名字空间
- 数据分层存储及分布策略
- 数据加密存储

跨机房存储：HDFS
- 存储资源弹性上云
- 多云部署

资源弹性伸缩自动部署

架构稳定性关键因素-可用性&容灾双活

Yarn

难点：跨机房部署带宽压力大
解决方案：
改造源码，通过标签调度结合HDFS跨机房容灾、Alluxio，实现跨机房双活

HBase

难点：客户端与业务系统嵌入深，切换难
解决方案：
建立HBase管理平台，提供HBase SDK，实现远程一键切换

HDFS

难点：跨机房部署性能低、不稳定
解决方案：
改造源码实现双活

ElasticSearch

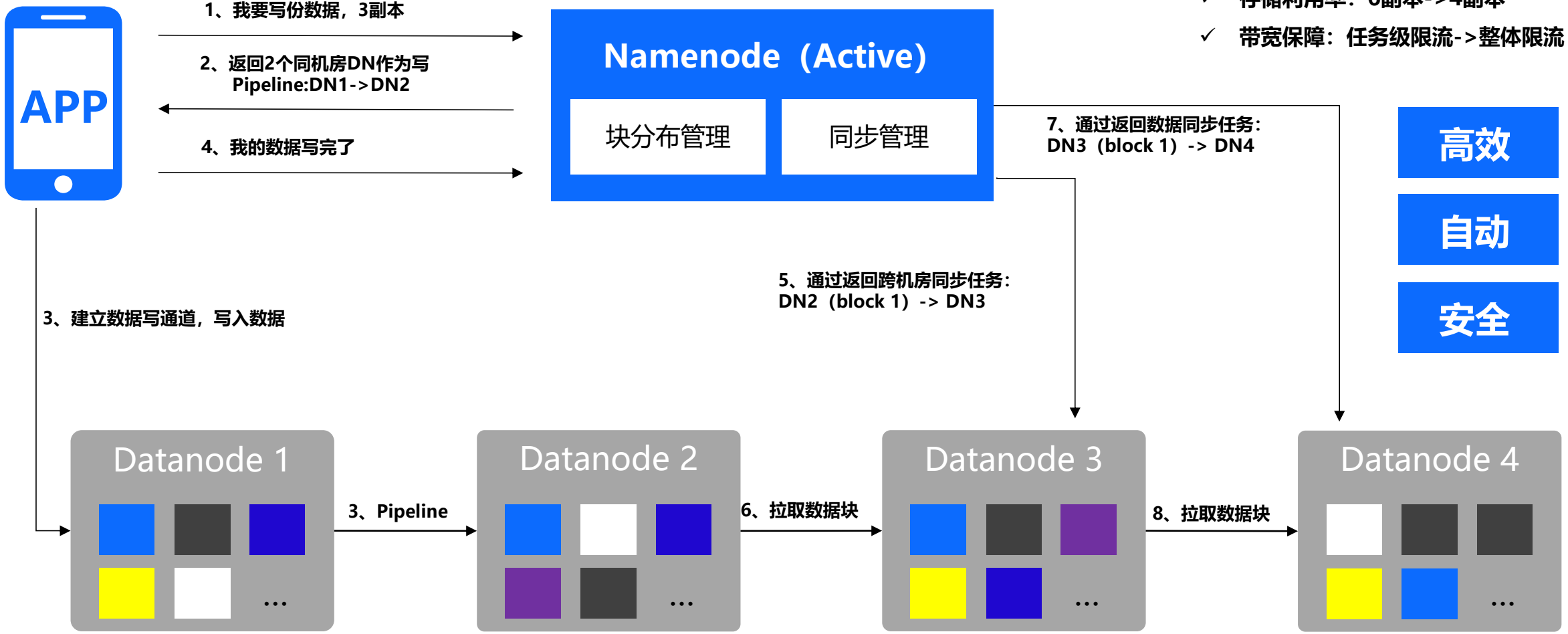
难点：双活依靠双写，有一致性问题，效率低，客户端不便切换
解决方案：
改造源码，通过CCR方式实现数据主从同步；建立数据服务平台，让ES的使用服务化

Kafka

难点：主备集群间偏移量不一致，客户端与业务系统嵌入深，切换难
解决方案：
建立Kafka管理平台，改造MirrorMaker，实现偏移量的同步，并提供Kafka SDK，实现远程一键切换

架构稳定性关键因素-可用性&容灾双活

- ✓ 写入性能：3副本->2副本
- ✓ 同步效率：DistCP->自动块分布
- ✓ 存储利用率：6副本->4副本
- ✓ 带宽保障：任务级限流->整体限流



架构稳定性关键因素-自适性&自动化评估

人工 → 全自动

开发

测试

上线

运营

1 解决方案选型

根据业务场景，评估
推荐方案

2 性能评估

- 业务逻辑性能评估
- 性能调优

3 压力评估

- 运营安全性
- 容量、预算

4 容量评估

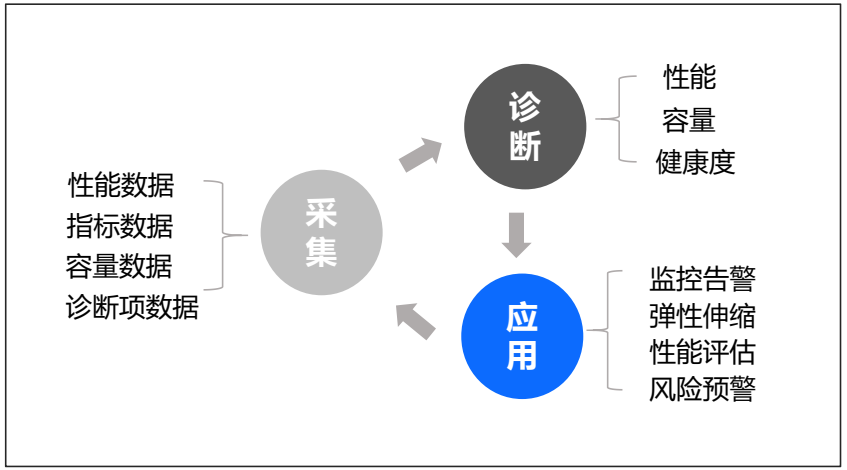
根据用户容量变化
指引自动扩缩容

5 风险评估

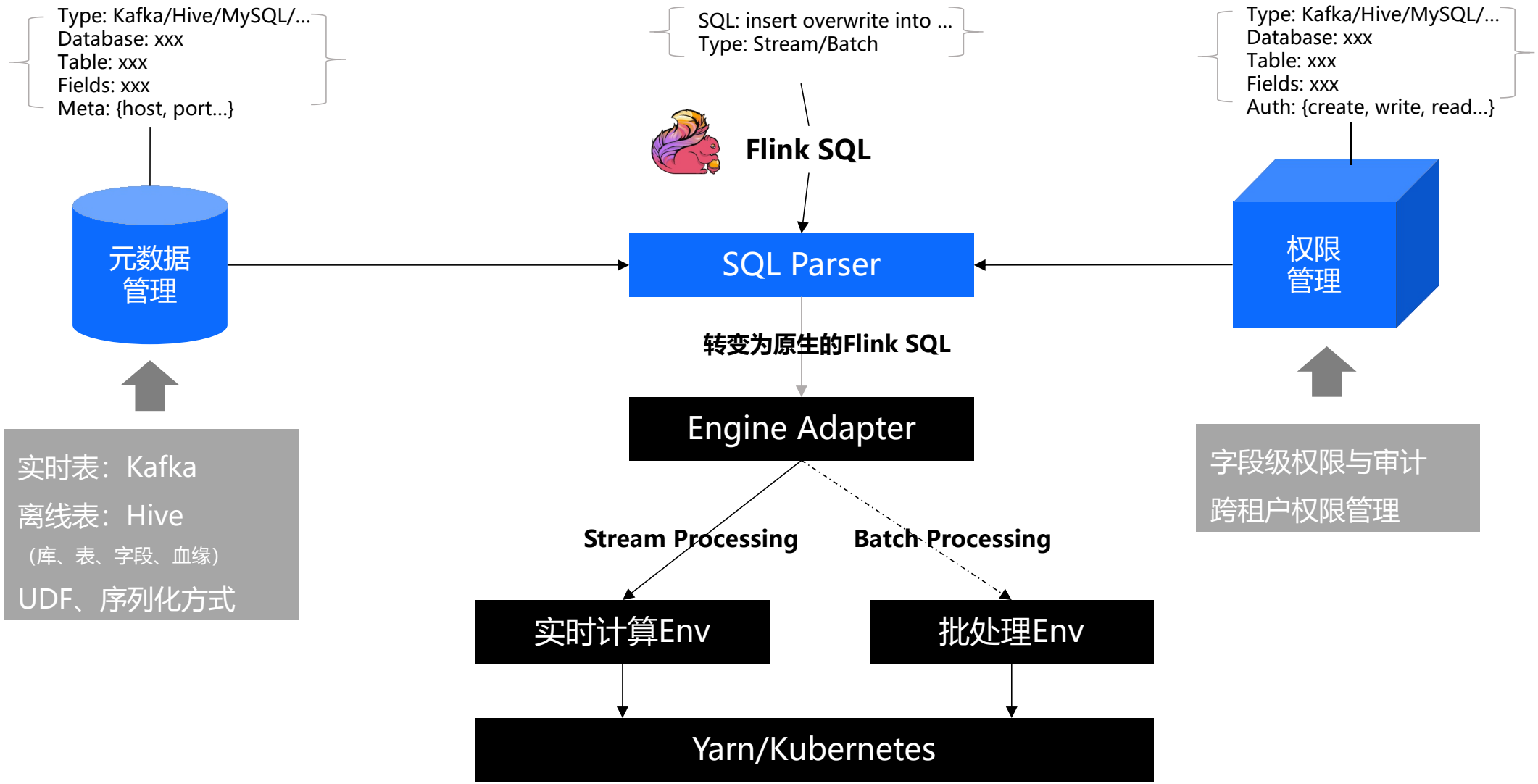
定期体检：诊断系统核心
指标，规避潜在风险

6 诊断项沉淀

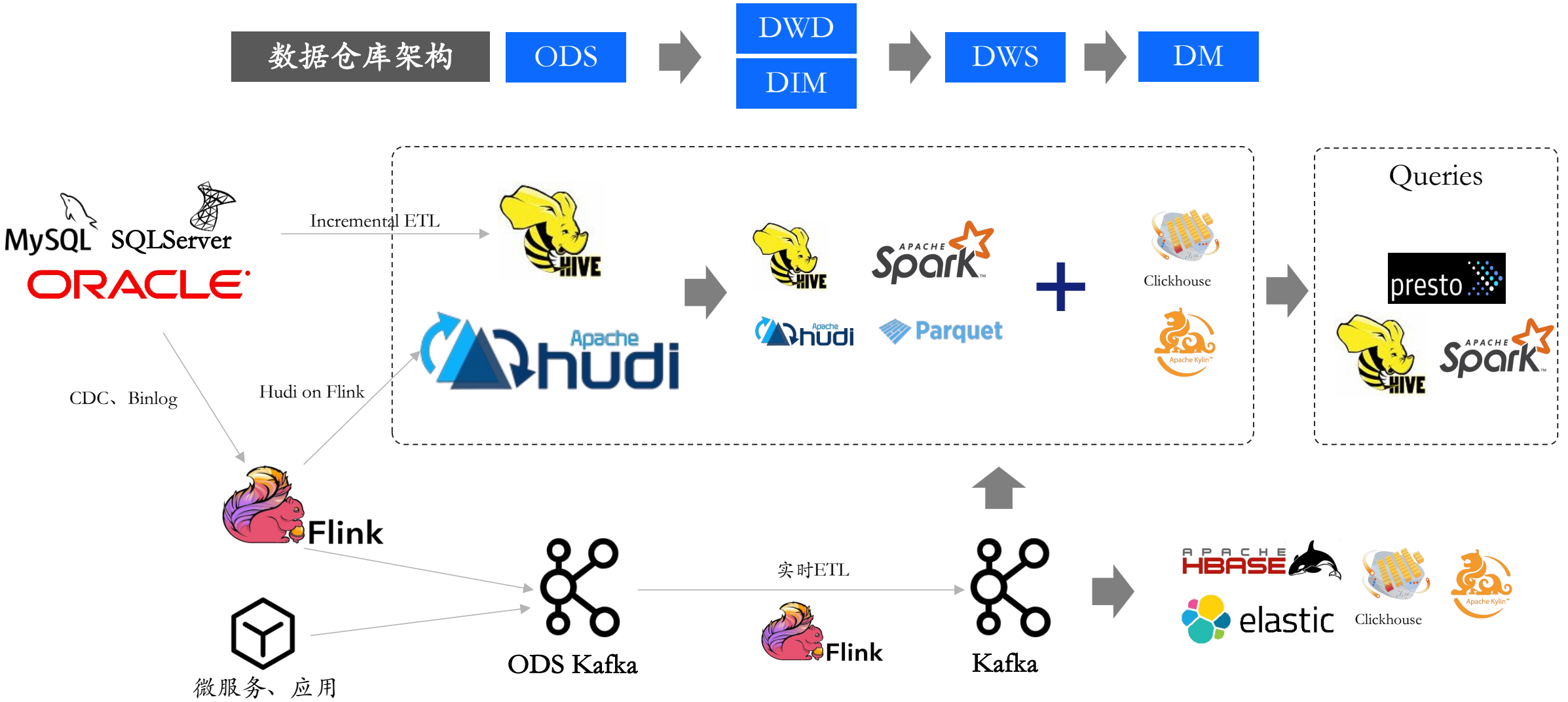
- 精细化运营
- 用户侧反馈



架构稳定性关键因素-易用性&批流一体化



架构稳定性关键因素-先进性&数据仓库实时化



三、未来大数据架构畅想

产业趋势

传统大数据厂商	公有云厂商	新兴独角兽、科技公司 Snowflake、Databrick
核心打法：**平台（私有化为主）+数据治理+定制化开发方式** **发展情况**：基本没有太多创新，更多是项目方式，项目毛利平均在40%左右。行业上主要聚焦在金融、政府、零售、地产、制造，平均实施周期2~3月	**核心打法**：**云基础设施+生态能力** **发展情况**：都布局云原生数据湖能力，如datalake产品，相对早期，市场感知度不强。大数据EMR的布局相对成熟些，行业打法上，目前还是以生态为主，聚焦IaaS。平均实施周期1~2天	**核心打法**：**聚焦单品** **发展情况**：商业模式就是单品，不承接数据治理和定制化开发，做好标准化（SQL）支持、接口开放性和线上运营支持。聚焦金融、互联网、零售、央国企、制造等行业，平均实施周期1~2天。

第一代云上数仓（发展期） **私有化数据湖**	第二代云上数仓（成熟期） **云上数据湖**	第三代云上数仓（幻灭期） **云原生数据湖**	第四代云上数仓（萌芽期） **云原生实时数据湖**
主要代表产品：***等厂商，相比传统oracle、db2，**解决大规模OLAP分析场景**	主要代表产品：AWS EMR、Alibaba EMR、Cloudera**一定程度上增加弹性能力**，解放IT维护成本	主要代表产品如：***Datalake通过**存算分离、弹性伸缩**等技术，实现动态伸缩和精准计费	主要代表产品：snowflake、databricks等，通过**存算分离、实时数仓、多云融合**等核心技术，实现弹性伸缩和多云统一架构
• Hadoop技术路线，存算一体 • 以私有化为主，按节点license结算 • 除大数据节点外，提供数据治理和定制化开发服务	• Hadoop技术路线，存算一体 • 依托公有云IaaS资源，以EMR形式对外提供服务 • 降低集群扩缩容和运维自动化成本	• 计算存储分离、精细化资源管理 • 具备DLF能力（元数据迁移、对象存储元数据发现、元数据管理） • 通过弹性伸缩，降低计算成本，同时提供DLF能力，帮助客户快速建仓	• 多云适配，AWS、Azure、GCP、alibaba cloud、tencent cloud等 • 支持实时数仓统一架构，实现批流合一和数仓指标实时化 • 兼顾私有云的数据安全需求和公有云的弹性资源需求

场景趋势

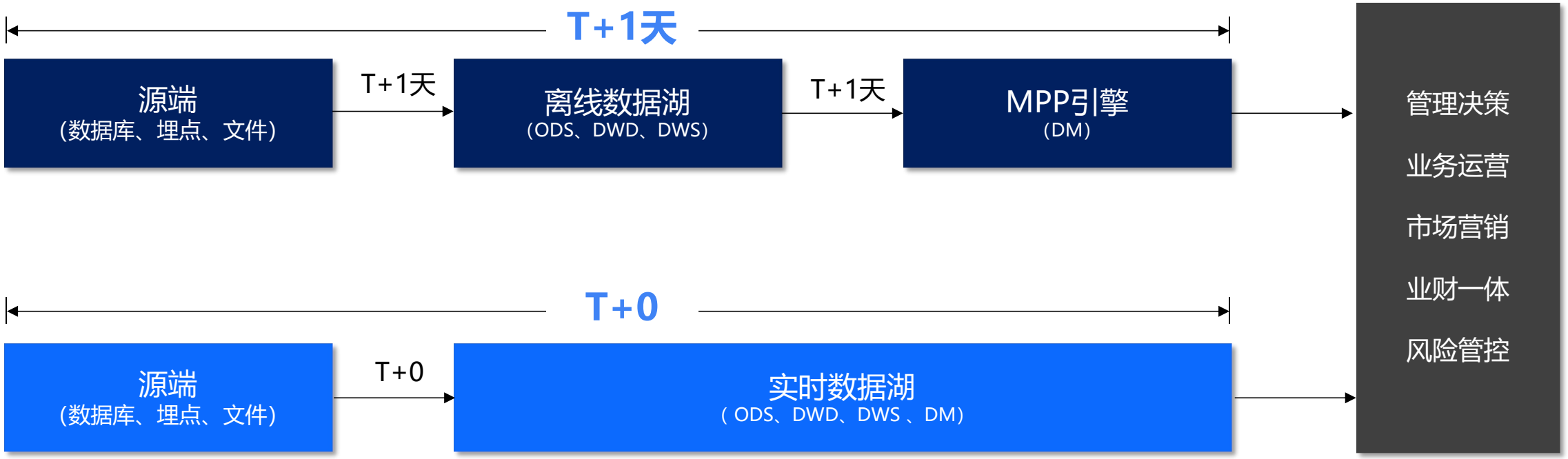
	“实时数仓，批流合一”场景	“存算分离，弹性伸缩”场景	“多云管理，跨云计算”场景
痛点	• 面向金融、快消零售和物流行业，**以前大部分指标是T+1天，少部分T+0** • 客户需求大部分指标1分钟内呈现，使用离线+实时lambda架构，**不仅耗费大量资源，还会出现指标不一致情况，如某垂直电商**	• 金融、快消零售和物流行业具备季节性属性，业务高峰时后台计算资源需求成本增长，**扩充IT资源耗资巨大且浪费** • **容灾机房、公有云等资源池无法充分利用，**这两部分的闲置计算资源较多	• 跨国企业和大型央国企，**业务常涉及多朵云，**比如某化妆品企业两朵云、某零售头部企业三朵云、某奶制品巨头国内三朵云等 • 如何解决多云环境下，**统一数据湖管理和合规跨云计算，是客户最关心的问题**
说明	• 目前主流实时数仓技术hudi，虽已开源但是有不少生产问题，**包括性能和稳定性问题，离实际生产应用还有一段距离** • 顺丰在这个基础上，已经解决了社区尚未解决的问题，**并在内部落地，数仓计算效率提高4倍，数仓时效到1分钟以内**	• 目前国内主流公有云云目前只聚焦**在自家单朵云的弹性伸缩能力上发展** • 我们从客户角度出发，**目前已经具备混合云弹性伸缩能力**	• 目前Snowflake和Databricks支持多云适配，但不支持跨云统一管理 • **顺丰已经支持多云管理和部分跨云计算**

架构趋势

云原生实时数据湖，打造**存算分离、实时数仓、湖仓一体**三大核心能力

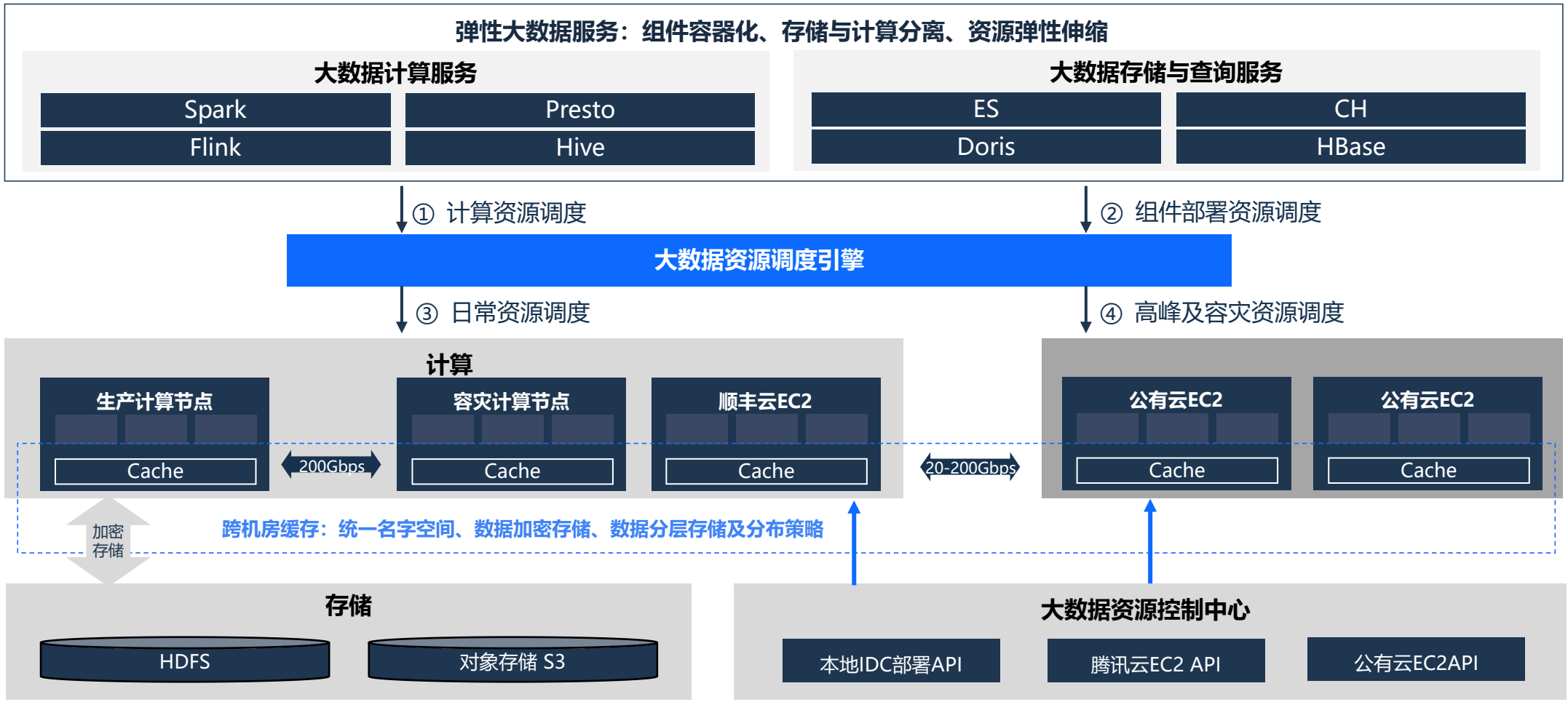
客户价值：T+1-->T+0

天下武功、唯快不破，谁的数据结果出得快，谁赢的可能性就越大



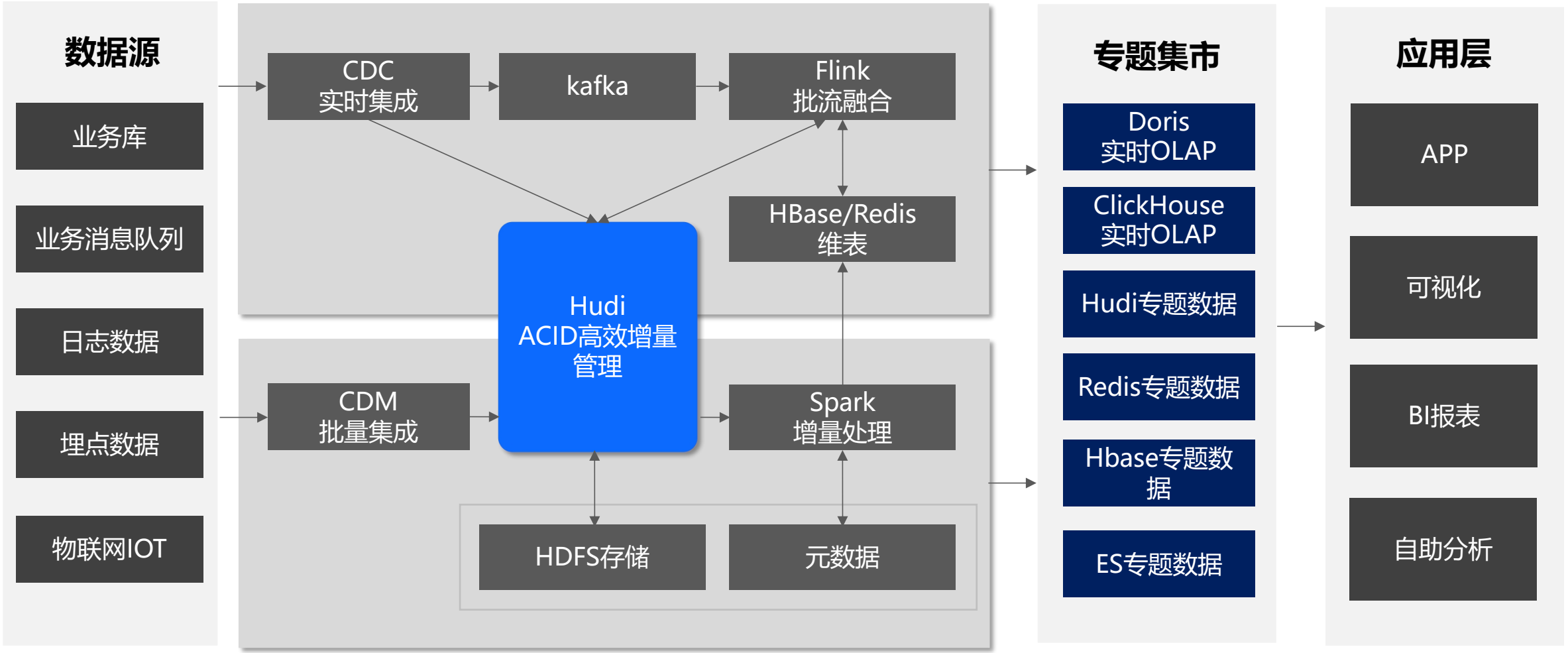
关键能力-极致弹性（成本极致优化）

- 通过存算分离技术，复用容灾和公有云资源，确保了数据安全的同时，复用公有云弹性资源



关键能力-实时数据湖（指标 T+1天->T+0 ）

- 基于Hudi升级后大幅提升的数仓更新时效，由原来的“天”级别提升到“秒”级别



关键能力-统一SQL（开发更高效&高速）

跨云、跨大数据引擎全局统一元数据管理，支持基于代价估算的全局解析执行引擎

- 支持**跨云、跨大数据分析引擎**的融合分析
- 支持无感优化用户大数据架构，支撑已有技术生态，实现向云上数仓的平滑过渡

关键能力-安全托管（安全计算）

确保客户对数据密钥有自主管理权

确保通信从南北向到东西向都是安全的，确保数据落地的加密程度是足够

密钥管理

- KMS密钥生成
- KMS密钥托管
- 生物密钥加盐生成器

RPC服务层

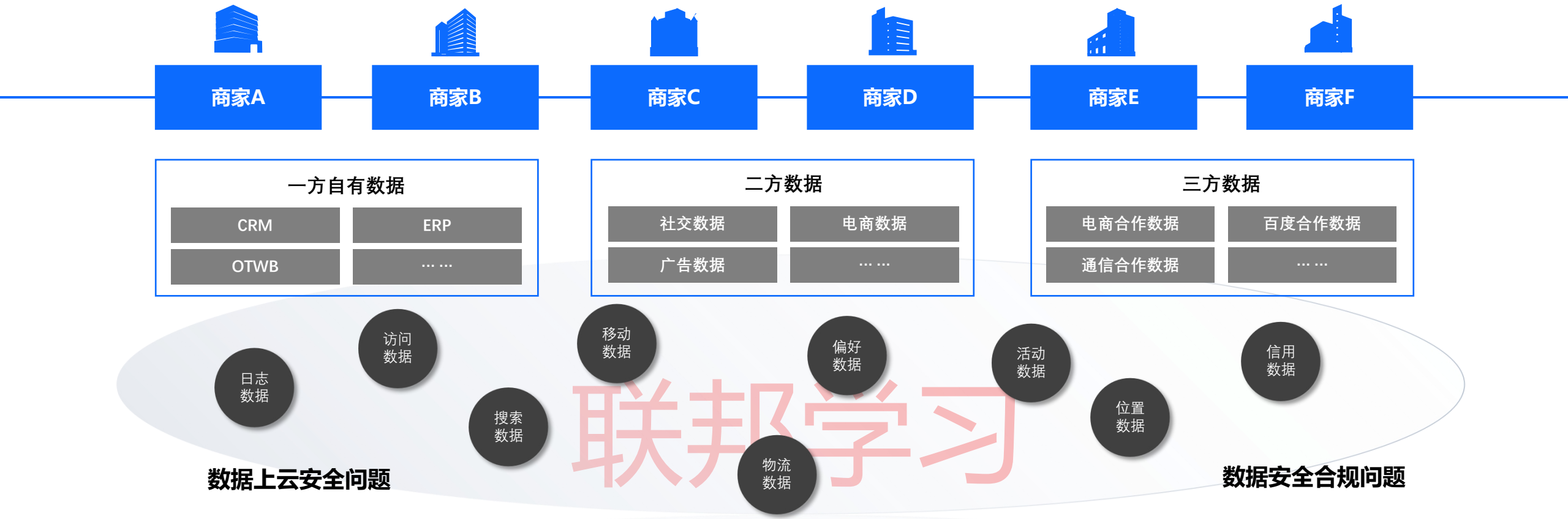
- 白名单过滤
- 南北向链路SSL
- 东西向链路SSL
- 库表列行权限控制

弹性存储层

- 全数据AES256强加密
- 密钥周期性更新
- 数据定期重新加密

关键能力-数据生态（数据生态共赢）

默认为每一个公有云和私有云客户部署一个联邦学习节点，数据不共享，但是模型参数共享，构建隐私计算数据交易市场



顺丰-云原生实时数据湖

我们的使命：

“让每个用户的数字化更简单、更安全、更高效，为全球数字经济和人类美好生活贡献力量”

扫码申请体验