ALIBABA CLOUD

阿里云

专有云企业版

云原生多模数据库Lindorm

产品简介

产品版本：V3.15.0

文档版本：20210813

阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。

2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。

4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。

5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。

6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击**设置**> **网络**> **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
Courier字体	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.什么是云原生多模数据库Lindorm

Lindorm是一款适用于任何规模、多种模型的云原生数据库服务，支持海量数据的低成本存储处理，提供宽表、时序、搜索、文件等多种数据模型，兼容HBase、Cassandra、Phoenix、OpenTSDB、Solr、SQL等多种开源标准接口，是互联网、IoT、车联网、广告、社交、监控、游戏、风控等场景首选数据库，也是为阿里巴巴核心业务提供关键支撑的数据库之一。

Lindorm基于存储计算分离、多模共享融合的云原生架构，具备弹性、低成本、简单易用、开放、稳定等优势，适合元数据、日志、账单、标签、消息、报表、维表、结果表、Feed流、用户画像、设备数据、监控数据、传感器数据、小文件、小图片等数据的存储和分析，其核心能力包括：

- 融合多模：支持宽表、时序、搜索、文件四种模型，提供统一联合查询和独立开源接口两种方式，模型之间数据互融互通，帮助应用开发更加敏捷、灵活、高效。
- 极致性价比：支持千万级高并发吞吐、毫秒级访问延迟，并通过高密度低成本存储介质、智能冷热分离、自适应压缩，大幅减少存储成本。
- 云原生弹性：支持计算资源、存储资源独立弹性伸缩，并提供按需即时弹性的Serverless服务。
- 开放数据生态：提供简单易用的数据交换、处理、订阅等能力，支持与MySQL、Spark、Flink、Kafka等系统无缝打通。

多模介绍

Lindorm系统支持宽表、时序、搜索、文件四种模型，提供统一联合查询和独立开源接口两种方式，模型之间数据互融互通，帮助应用开发更加敏捷、灵活、高效。多模型的核心能力由四大数据引擎提供，包括：

- 宽表引擎

  面向海量KV、表格数据，具备全局二级索引、多维检索、动态列、TTL等能力，适用于元数据、订单、账单、画像、社交、feed流、日志等场景，兼容HBase、Phoenix(SQL)、Cassandra(CQL)等开源标准接口。

  支持千万级高并发吞吐，支持百PB级存储，吞吐性能是开源HBase(Apache HBase)的3-7倍，P99时延为开源HBase(Apache HBase)的1/10，平均故障恢复时间相比开源HBase(Apache HBase)提升10倍，支持冷热分离，压缩率比开源HBase(Apache HBase)提升一倍，综合存储成本为开源HBase(Apache HBase)的1/2。

- 时序引擎

  面向IoT、监控等场景存储和处理量测数据、设备运行数据等时序数据。提供HTTP API接口(兼容OpenTSDB API)、支持SQL查询。针对时序数据设计的压缩算法，压缩率最高为15:1。支持海量数据的多维查询和聚合计算，支持降采样和预聚合。

- 搜索引擎

  面向海量文本、文档数据，具备全文检索、聚合计算、复杂多维查询等能力，同时可无缝作为宽表、时序引擎的索引存储，加速检索查询，适用于日志、账单、画像等场景，兼容开源Solr标准接口。

- 文件引擎

  提供宽表、时序、搜索引擎底层共享存储的服务化访问能力，从而加速多模引擎底层数据文件的导入导出及计算分析效率，兼容开源HDFS标准接口。

对于目前使用类HBase+ElasticSearch或HBase+OpenTSDB+ES的应用场景，比如监控、社交、广告等，利用Lindorm的原生多模能力，将很好地解决架构复杂、查询痛苦、一致性弱、成本高、功能不对齐等痛点，让业务创新更高效。

2.产品优势

Lindorm是一款适用于任何规模、多种模型的云原生数据库服务，其基于存储计算分离、多模共享融合的云原生架构设计，具备弹性、低成本、稳定可靠、简单易用、开放、生态友好等优势。

云原生弹性

- 基于存储计算分离的全分布式架构，支持计算资源和存储资源的独立弹性伸缩。
- 存储资源支持秒级在线扩缩，计算资源（宽表引擎、时序引擎、搜索引擎）支持分钟级在线伸缩。
- 提供按需即时弹性的Serverless服务，自适应弹性伸缩，无需人工容量管理。

低成本

- 提供性能型、标准型、容量型多种存储规格，可满足不同场景的性价比选择。
- 多种引擎共享统一的存储池，减少存储碎片，降低使用成本。
- 容量型存储单价为业界最低标准，大幅低于基于ECS本地盘自建。
- 内置深度优化的压缩算法，数据压缩率高达10:1以上，相比snappy提高50%以上。
- 内置面向数据类型的自适应编码，数据无需解码，即可快速查找。
- 支持智能冷热分离，多种存储规格混合使用，大幅降低数据存储综合成本。

高性能

- 宽表引擎：支持千万级并发吞吐，支持百PB级存储，吞吐性能是开源HBase的3~7倍，P99时延为开源HBase的1/10。
- 时序引擎：写入性能和查询性能是InfluxDB的1.3倍，是OpenTSDB的5~10倍。
- 搜索引擎：基于Lucene引擎深度优化，综合性能比开源Solr/ES提升30%。

多模互通融合

- 多模型之间支持数据互通，搜索引擎可无缝作为宽表引擎、时序引擎的索引存储，加速复杂条件查询。
- 支持统一的SQL访问，以及跨多模引擎关联查询。

易用

- 兼容多种开源标准接口，包括HBase/Cassandra/Phoenix、OpenTSDB、Solr，业务可以无缝迁移。
- 云托管服务，免运维。
- 图形化系统管理和数据访问，操作简单。

高可用

- 系统采用分布式多副本架构，集群自动容灾恢复。
- 支持跨可用区、强一致的容灾能力，具备金融级可用性标准。
- 支持全球多活部署。

高可靠

- 底层多副本存储，实现了数据的高可靠性。
- 提供企业级备份能力。
- 在阿里巴巴部署上万台，支持过10年天猫双十一，久经验证。

开放生态

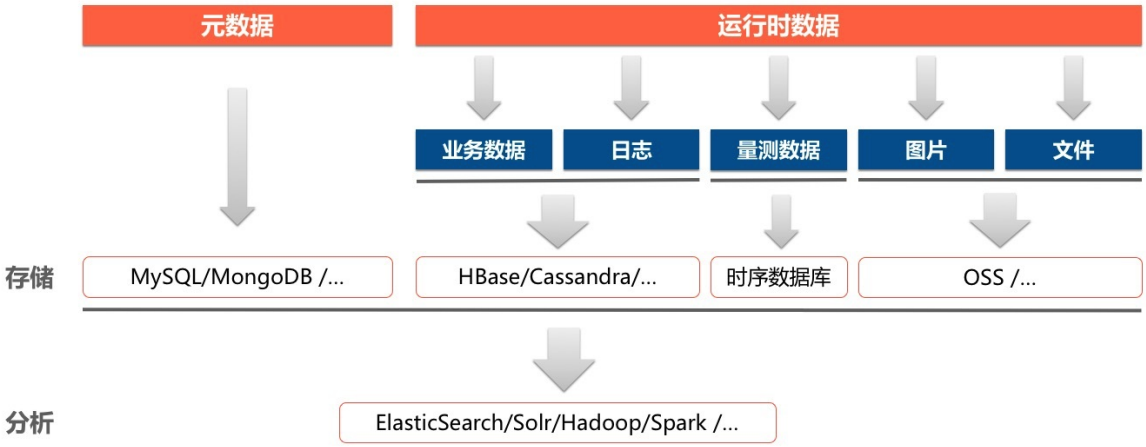
- 支持与MySQL、HBase、Cassandra等系统的平滑在线数据搬迁。
- 可轻松与Spark、Flink、DLA、MaxCompute等计算引擎无缝对接。
- 支持无缝订阅Kafka、SLS等日志通道的数据，并具备快速处理能力。
- 通过Lindorm Stream，可以实时订阅Lindorm的增量变更数据，自定义消费。

3.产品架构

本文介绍云原生多模数据库Lindorm的产品架构。

业务背景

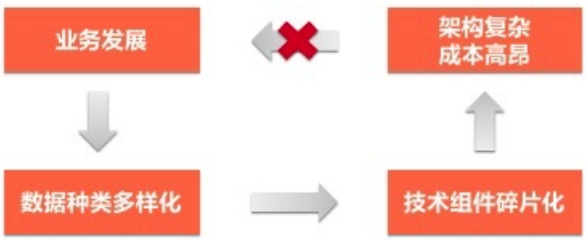
伴随着信息技术的飞速发展，各行各业在业务生产中产生的数据种类越来越多，有结构化的业务元数据、业务运行数据、设备或者系统的量测数据，也有半结构化的业务运行数据、日志、图片或者文件等。按照传统方案，为了满足多种类型数据的存储、查询和分析需求，在设计IT架构时，需要针对不同种类的数据，采用不同的存储分析技术，如下图：



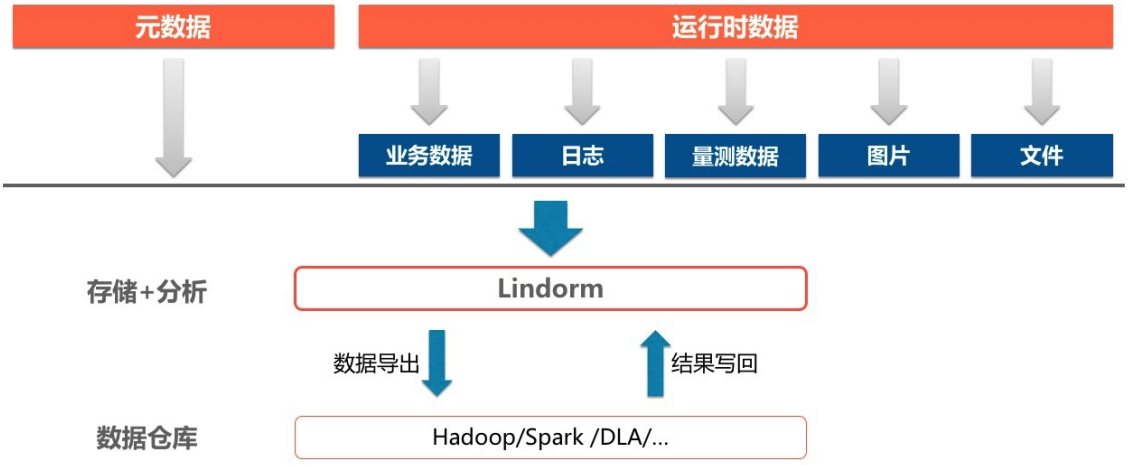
这种技术方案，是一种典型的技术碎片化的处理方案。针对不同的数据，使用不同的数据库来处理。有如下几个弊端：

- 涉及的技术组件多且杂
- 技术选型复杂
- 数据存取、数据同步的链路长

这些弊端会对信息系统建设带来巨大的问题，对技术人员要求高、业务上线周期长、故障率高、维护成本高。更进一步，技术碎片化导致技术架构割裂，不利于技术架构的长期演进和发展，最终导致技术架构无法跟进业务前进的步伐。举个简单的例子，当业务发展，需要支持跨可用区高可用、全球同步或者降低存储成本时，各技术组件都需要独立演进和发展，耗时耗人耗力，是一件非常痛苦的事情。并且随着业务的发展，数据的类型会越来越多，对不同种类数据的差异化处理需求会日渐增加，会导致数据存储碎片化更加严重。

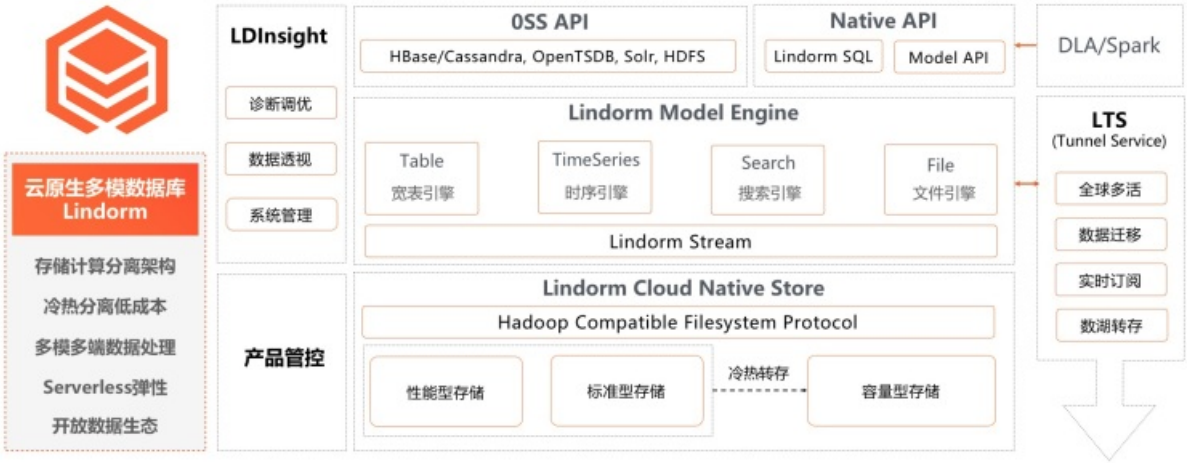


当前信息化技术发展面临的一个主要矛盾是"日益多样的业务需求带来的多种类型数据与数据存储技术架构日趋复杂成本快速上升之间的矛盾"。伴随5G、IoT、智能网联车等新一代信息技术的逐步普及应用，这个矛盾会越来越突出。为了解决这个问题，阿里云自研了业界独家云原生多模数据库Lindorm，满足多模型数据的统一存储、查询和分析需求。如下图所示，与传统方案相比，Lindorm系统将极大地简化数据存储技术架构设计，大幅度提升系统稳定性，降低建设成本投入。



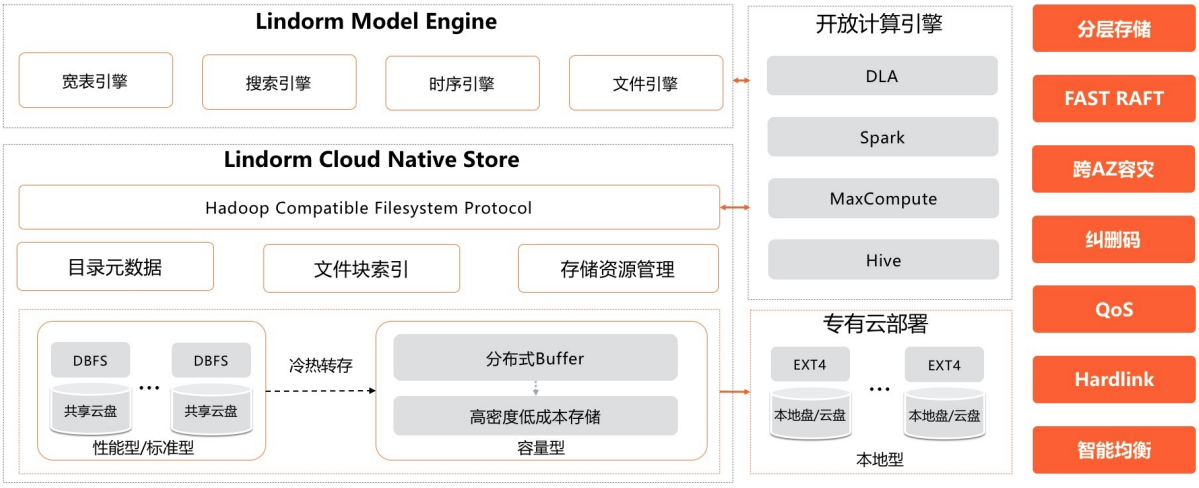
总体架构

Lindorm创新性地使用存储计算分离、多模共享融合的云原生架构，以适应云计算时代资源解耦和弹性伸缩的诉求。其中云原生存储引擎LindormStore为统一的存储底座，向上构建各个垂直专用的多模引擎，包括宽表引擎、时序引擎、搜索引擎、文件引擎。在多模引擎之上，Lindorm既提供统一的SQL访问，支持跨模型的联合查询，又提供多个开源标准接口（HBase/Phoenix/Cassandra、OpenTSDB、Solr、HDFS），满足存量业务无缝迁移的需求。最后，统一的数据Stream总线负责引擎之间的数据流转和数据变更的实时捕获，以实现数据迁移、实时订阅、数湖转存、数仓回流、单元化多活、备份恢复等能力。



存储引擎

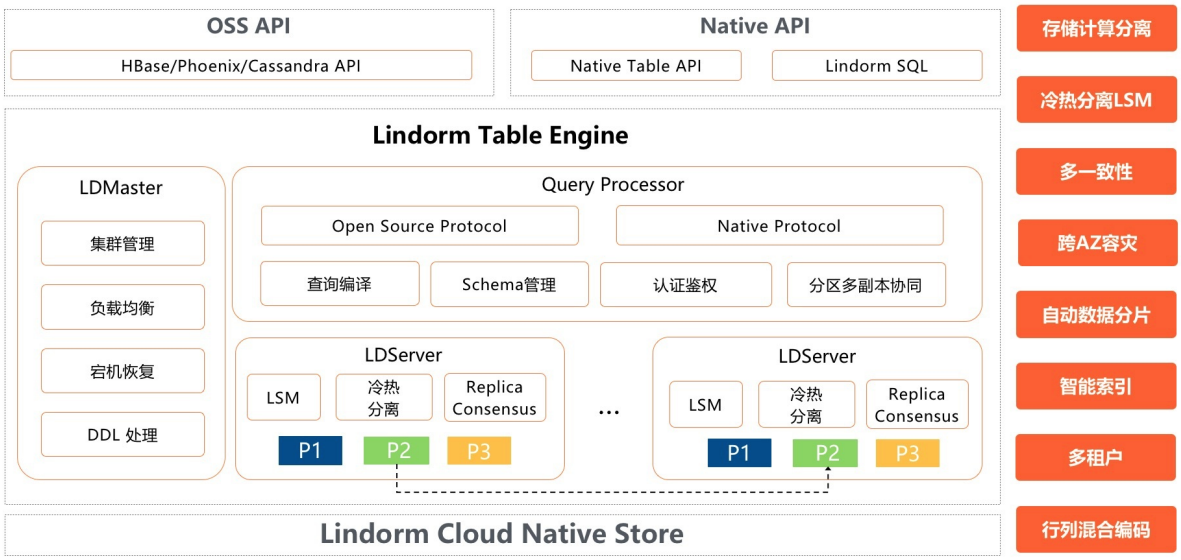
LindormStore是面向云基础存储设施设计、兼容HDFS协议的分布式存储系统，并同时支持运行在本地盘环境，以满足部分专有云、专属大客户的需求，向多模引擎和外部计算系统提供统一的、与环境无关的标准接口，整体架构如下：



LindormStore提供性能型、标准型、容量型等多种规格，并且面对真实场景的数据冷热特点，支持性能型/标准型、容量型多种存储混合使用的形态，以适应真实场景的数据冷热特点，可结合多模引擎的冷热分离能力，实现冷热存储空间的自由配比，让用户的海量数据进一步享受云计算的低成本红利。

宽表引擎

LindormTable是面向海量半结构化、结构化数据设计的分布式NoSQL系统，适用于元数据、订单、账单、画像、社交、feed流、日志等场景，兼容HBase、Phoenix(SQL)、Cassandra等开源标准接口。其基于数据自动分区+分区多副本+LSM的架构思想，具备全局二级索引、多维检索、动态列、TTL等查询处理能力，支持单表百万亿行规模、千万级并发、毫秒级响应、跨机房强一致容灾，高效满足业务大规模数据的在线存储与查询需求。面向海量半结构化、结构化数据设计的分布式NoSQL系统，能够兼容HBase、Phoenix(SQL)、Cassandra等开源标准接口。整体架构如下：



LindormTable的数据持久化存储在LStore中，表的数据通过自动Sharding分散到集群中的多台服务器上，并且每一个Parition可以拥有1至N个副本，这N个副本拥有主、从两种角色，主从副本可以加载在不同的Zone，从而保障集群的高可用和强一致。针对不同的一致性模式，主从副本之间的数据同步和读写模式如下：

- 强一致模式：只有主副本提供读写，数据会异步回放到从副本,主副本所在节点故障，从副本晋升为主(晋升之前会保障数据同步完成，从副本拥有所有最新数据，整体过程由Master协调负责。
- 最终一致模式：主从副本都提供读写，数据会相互同步，保证副本之间的数据最终一致。

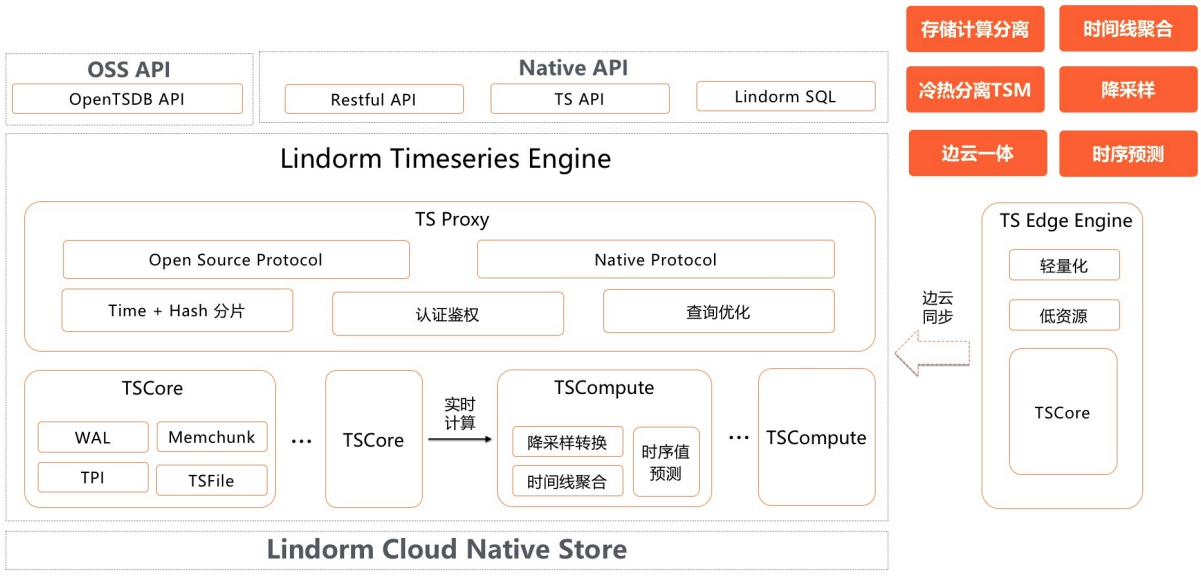
LindormTable的多副本架构基于PACELC理念设计，每一个数据表都支持单独支持设置一致性模式，从而拥有不同的可用性和性能。在最终一致模式下，服务端会对每一个读写请求在一定条件下触发多副本并发访问，从而大幅提升请求的成功率和减少响应毛刺。该并发机制建立在内部的异步访问框架上，相比于启动多线程，额外资源消耗可以忽略不计。对于并发访问的触发条件，主要包括两个类型：其一是限时触发，对于每一个请求，都可以单独设置一个GlitchTimeout，当请求运行时间超过该值未得到响应后，则并发一个请求到其他N-1个副本，最终取最快的那个响应；其二是黑名单规避，服务端内部会基于超时、抛错、检测等机制，主动拉黑存在慢、停止响应等问题的副本，使得请求能够主动绕开受软硬件缺陷的节点，让服务最大可能保持平滑。对于像掉电终止这样的不想应场景，在节点不可服务至失去网络心跳通常会存在一两分钟的延迟，利用LTable的这种多副本协同设计，可以将影响控制在10秒以内，大幅提升服务的可用性。

LindormTable的LSM结构面向冷热分离设计，支持用户的数据表在引擎内自动进行冷热分层，并保持透明查询，其底层结合LStore的冷热存储混合管理能力，大幅降低海量数据的总体存储成本。的LSM结构面向冷热分离设计，支持用户的数据表在引擎内自动进行冷热分层，并保持透明查询，其底层结合LStore的冷热存储混合管理能力，大幅降低海量数据的总体存储成本。

LindormTable提供的数据模型是一种支持类型的松散表结构。相比于传统关系模型，LTable除了支持预定义字段类型外，还可以随时动态添加列，而无需提前发起DDL变更，以适应大数据灵活多变的特点。同时，LTable支持全局二级索引、倒排索引，系统会自动根据查询条件选择最合适的索引，加速条件组合查询，特别适合如画像、账单场景海量数据的查询需求。

时序引擎

LindormTS是面向海量时序数据设计的分布式时序引擎，兼容开源OpenTSDB标准接口，其基于时序数据特点和查询方式，采用Timerange+hash结合的分区算法，时序专向优化的LSM架构和文件结构，支持海量时序数据的低成本存储、预降采样、聚合计算、高可用容灾等，高效满足IoT/监控等场景的测量数据、设备运行数据的存储处理需求，整体架构如下：

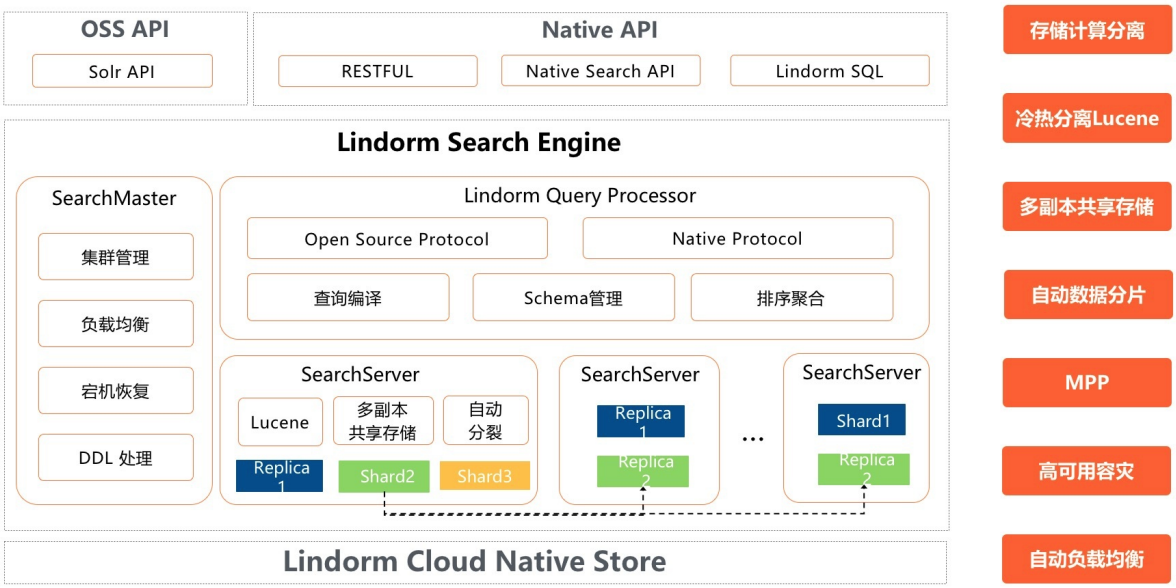


TSCore是时序引擎中负责数据组织的核心部分，其整体思想与LSM结构相似，数据先写入Memchunk，然后Flush到磁盘，但由于时序数据天然的顺序写入特征，定向专用的时序文件TSFile的结构设计为以时间窗口进行切片，数据在物理和逻辑上均按时间分层，从而大幅减少Compaction的IO放大，并使得数据的TTL、冷热分离等实现非常高效。

TSCompute是负责时序数据实时计算的组件，重点解决监控领域常见的降采样转换、时间线聚合、时序值预测等需求，其通过Lindorm Stream进行数据订阅，并完全基于内存计算，所以，整体非常的轻量、高效，适合系统已预置的计算功能。针对部分灵活复杂的分析需求，用户仍可以通过对接Spark、Flink等系统实现，从而支撑更多场景和适应业务变化。

搜索引擎

LindormSearch是面向海量数据设计的分布式搜索引擎，兼容开源Solr标准接口，同时可无缝作为宽表、时序引擎的索引存储，加速检索查询。其整体架构与宽表引擎一致，基于数据自动分区+分区多副本+Lucene的结构设计，具备全文检索、聚合计算、复杂多维查询等能力，支持水平扩展、一写多读、跨机房容灾、TTL等，满足海量数据下的高效检索需求，具体如下：



LindormSearch的数据持久化存储在LindormStore中，通过自动Sharding的方式分散到多台SearchServer中，每一个分片拥有多个副本，支持一写多读，提升查询聚合的效率，同时这些副本之间共享存储，有效消除副本之间的存储冗余。

在Lindorm系统中，LindormSearch既可以作为一种独立的模型，提供半结构化、非结构化数据的松散文档视图，适用于日志数据分析、内容全文检索；也可以作为宽表引擎、时序引擎的索引存储，对用户保持透明，即宽表或时序中的部分字段通过内部的数据链路自动同步搜索引擎，而数据的模型及读写访问对用户保持统一，用户无需关心搜索引擎的存在，跨引擎之间的数据关联、一致性、查询聚合、生命周期等工作全部由系统内部协同处理，用简单透明的方式发挥多模融合的价值。

文件引擎

LindormFile是基于LindormStore轻量封装的分布式文件模型服务，其核心是提供安全认证、限流保护、多网络访问等服务化能力，从而使得外部系统可以直接访问多模引擎的底层文件，大幅提升备份归档、计算分析等场景的效率。同时，用户可以在离线计算系统直接生成底层数据格式的物理文件，通过LindormFile入口，将其快速导入到LindormStore中，以减少对在线服务的影响。对于部分存储计算的混合场景，用户可以将计算前的源文件存在LindormFile，计算后的数据结果存在LindormTable，结合Spark/DLA等大规模计算引擎，实现简单高效的数据湖分析能力。

4.功能特性

云原生多模数据库Lindorm支持云原生弹性、低成本存储、高性能读写等功能，为您的业务稳定提供保障。

云原生弹性

- 基于存储计算分离的全分布式架构，支持计算资源和存储资源的独立弹性伸缩。
- 存储资源支持秒级在线扩缩，计算资源（宽表引擎、时序引擎、搜索引擎）支持分钟级在线伸缩。

低成本存储

- 提供多种存储规格，可满足不同场景的性价比选择。
- 多种引擎共享统一的存储池，减少存储碎片，降低使用成本。
- 内置深度优化的压缩算法，数据压缩率高达10:1以上，相比snappy提高50%以上。
- 内置面向数据类型的自适应编码，数据无需解码，即可快速查找。
- 支持智能冷热分离，多种存储规格混合使用，大幅降低数据存储综合成本。

高性能读写

- 宽表引擎：支持千万级并发吞吐，支持百PB级存储，吞吐性能是开源HBase的3~7倍，P99时延为开源HBase的1/10。
- 时序引擎：写入性能和查询性能是InfluxDB的1.3倍，是OpenTSDB的5~10倍。
- 搜索引擎：基于Lucene引擎深度优化，综合性能比开源Solr/ES提升30%。

开发生态

- 支持与MySQL、HBase、Cassandra等系统的平滑在线数据搬迁。
- 可轻松与Spark、Flink、DLA、MaxCompute等计算引擎无缝对接。
- 支持无缝订阅Kafka、SLS等日志通道的数据，并具备快速处理能力。
- 通过Lindorm Stream，可以实时订阅Lindorm的增量变更数据，自定义消费。

5.应用场景

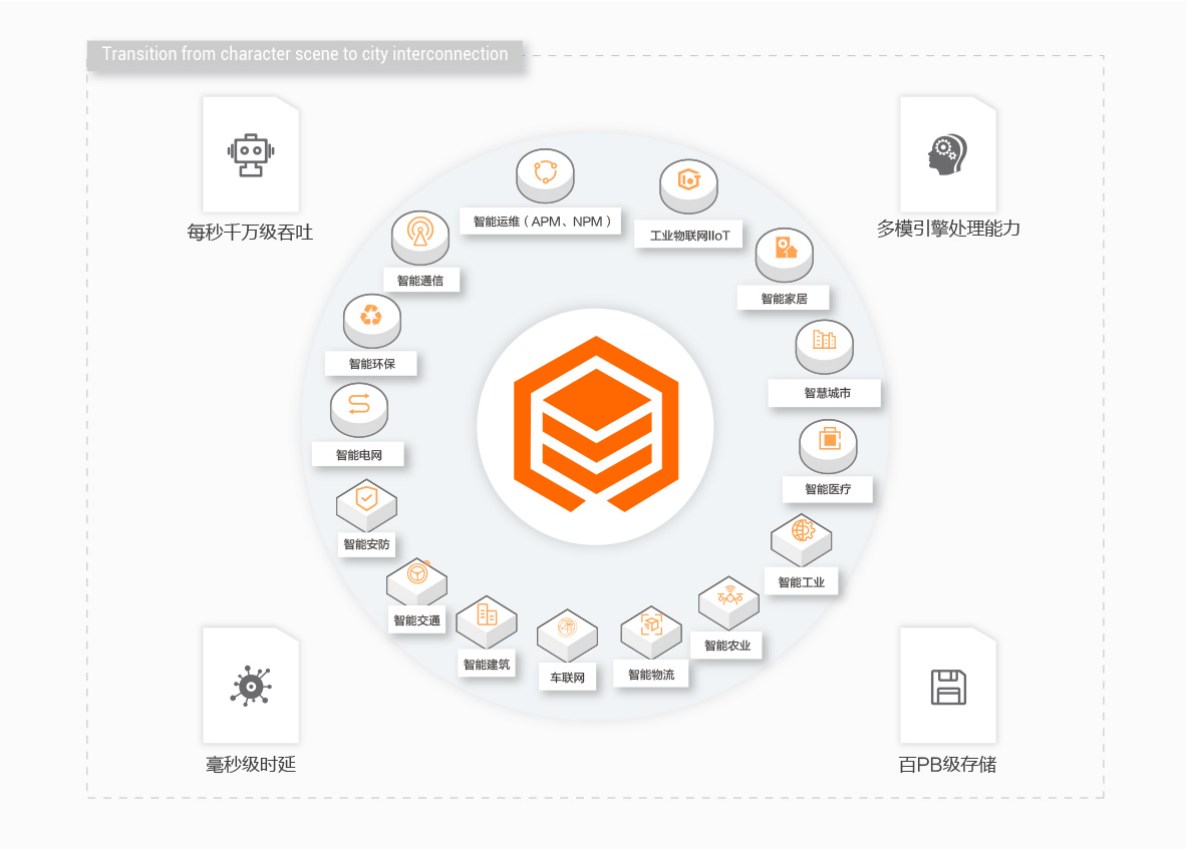
Lindorm是阿里云自研的云原生多模型数据库，面向海量多模型数据的低成本存储分析，构建万物互联时代的数据底座。Lindorm支持宽表模型、时序模型，提供自研的宽表引擎、时序引擎和搜索引擎，兼容HBase、Phoenix、OpenTSDB、Solr等多种开源标准接口，提供SQL查询、时序处理、检索分析等能力，满足结构化、半结构化的存储和分析需求，同时支持在线业务和离线业务。

Lindorm可以提供单个毫秒响应的性能，支持水平扩展到PB级存储和千万级QPS，在IoT、淘宝、支付宝、菜鸟等众多阿里巴巴核心服务中起到了关键支撑的作用。

阿里巴巴集团内部最佳实践

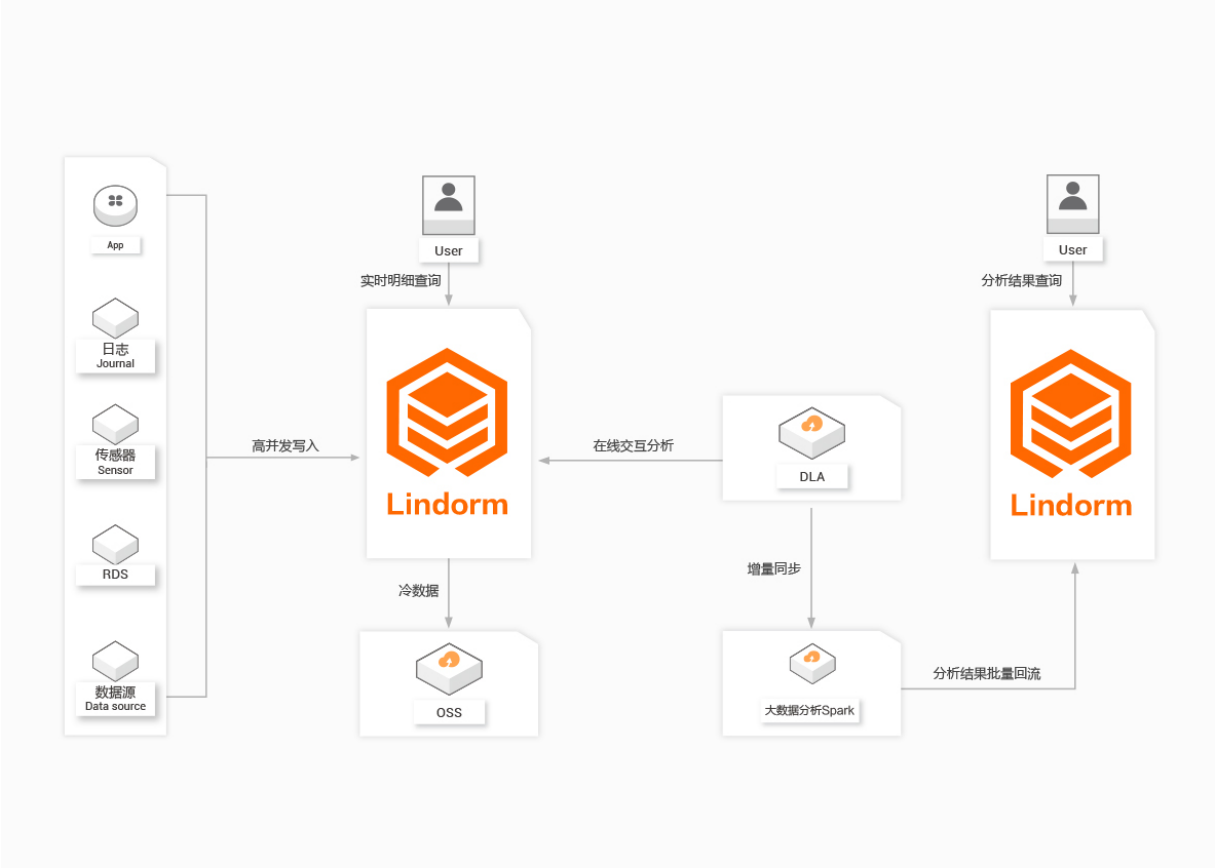
- Lindorm在阿里巴巴集团内部成熟业务中得到广泛使用。

- Lindorm基于自研的云原生多模架构支撑IoT业务飞速发展。



大数据场景：海量数据存储与分析

Lindorm支持海量数据的低成本存储、快速批量导入和实时访问，具备高效的增量及全量数据通道，可轻松与Spark、MaxCompute等大数据平台集成，完成数据的大规模离线分析。

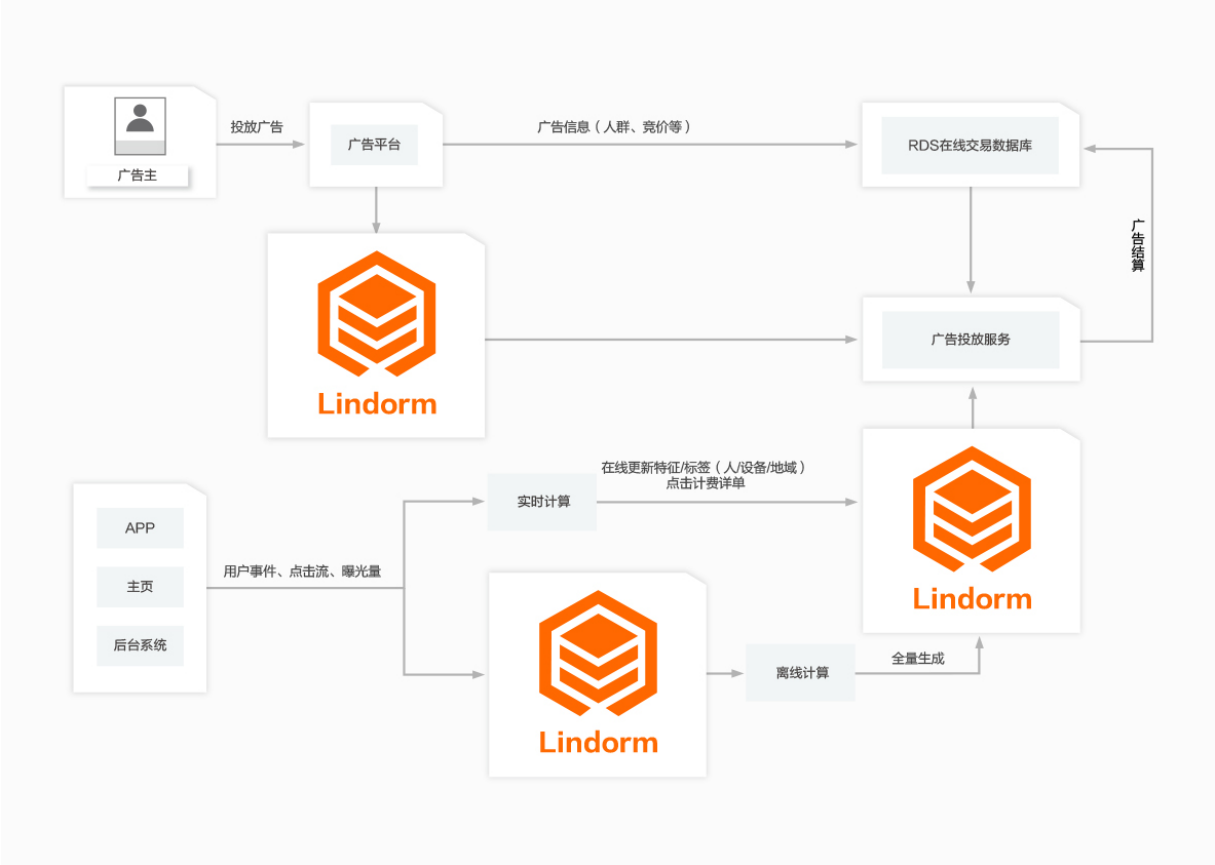


优势

- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 数据通道：通过BDS构建云Lindorm与异构计算系统的高效、易用的数据链路。
- 快速导入：通过BulkLoad将海量数据快速导入Lindorm，效率比传统方式提升一个数量级。
- 高并发：水平扩展至千万级QPS。
- 弹性：存储计算分离架构，支持独立伸缩，自动化扩容。

广告场景：海量广告营销数据的实时存储

使用Lindorm存储广告营销中的画像特征、用户事件、点击流、广告物料等重要数据，提供高并发、低延迟、灵活可靠的能力，帮助您快速构建领先的实时竞价、广告定位投放等系统服务。

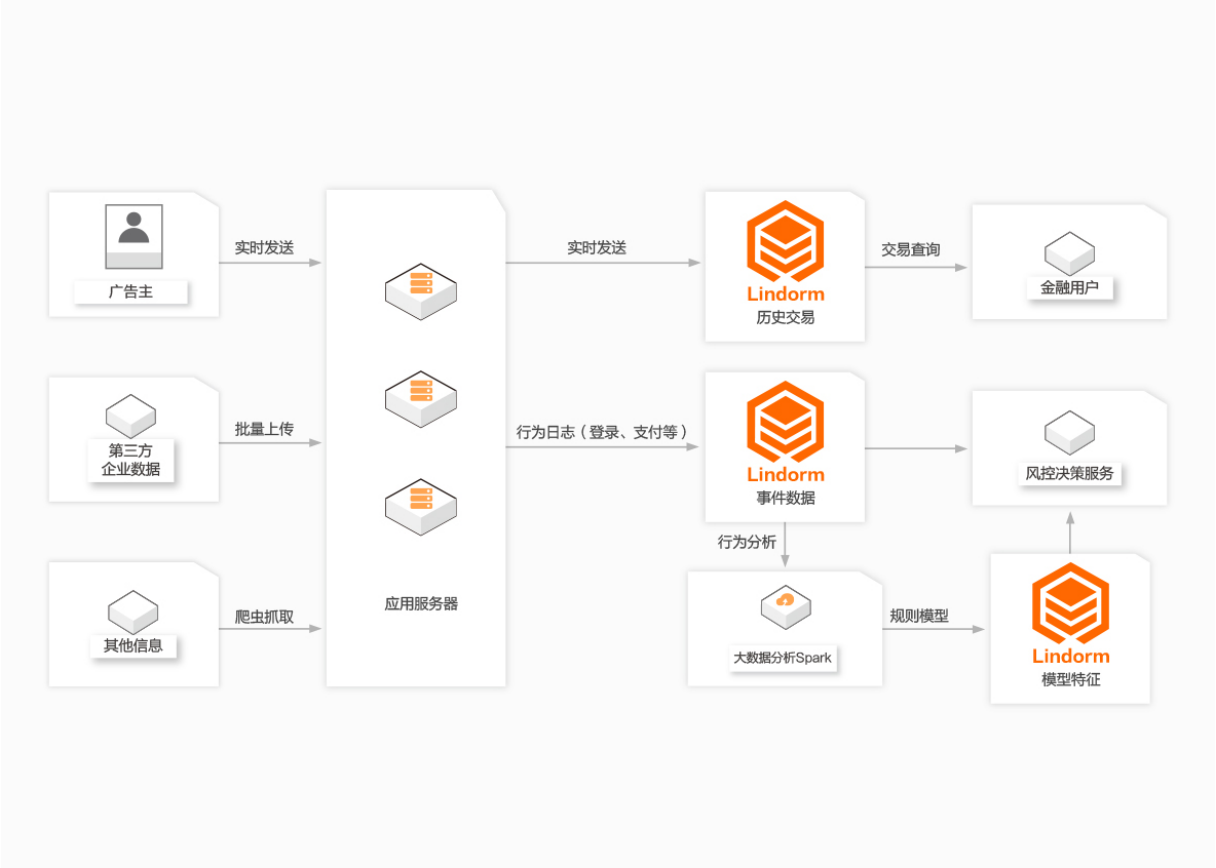


优势

- 低延迟：单个毫秒响应，支持双集群请求并发加速。
- 高并发：水平扩展至千万级QPS。
- 使用灵活：动态列，自由增减特征/标签属性；TTL，数据自动过期。
- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 数据通道：通过BDS构建云Lindorm与异构计算系统的高效、易用的数据链路。
- 高可用：主备双活容灾，请求自动容错。

金融&零售：海量订单记录与风控数据的实时存储

使用Lindorm存储金融交易中的海量订单记录，金融风控中的用户事件、画像特征、规则模型、设备指纹等重要数据，提供低成本、高并发、灵活可靠的能力，帮助您构建领先的金融交易与风控服务。

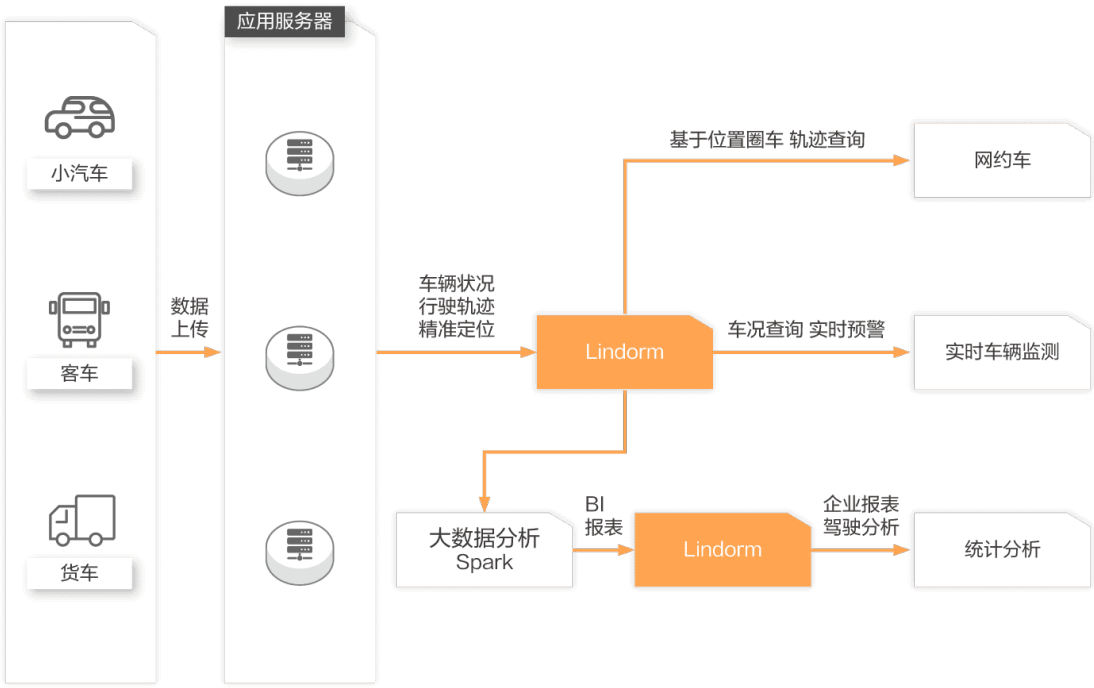


优势

- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 高并发：水平扩展至千万级QPS。
- 使用灵活：动态列，自由增减特征/标签属性；TTL，数据自动过期；多版本。
- 低延迟：单个毫秒响应，支持双集群请求并发加速。
- 数据通道：通过BDS构建Lindorm与异构计算系统的高效、易用的数据链路。
- 高可用：主备双活容灾，请求自动容错。

车联网：车辆轨迹与状况数据的高效存储处理

使用Lindorm存储车联网中的行使轨迹、车辆状况、精准定位等重要数据，提供低成本、弹性、灵活可靠的能力，帮助您构建领先的网约车、物流运输、新能源车检测等场景服务。

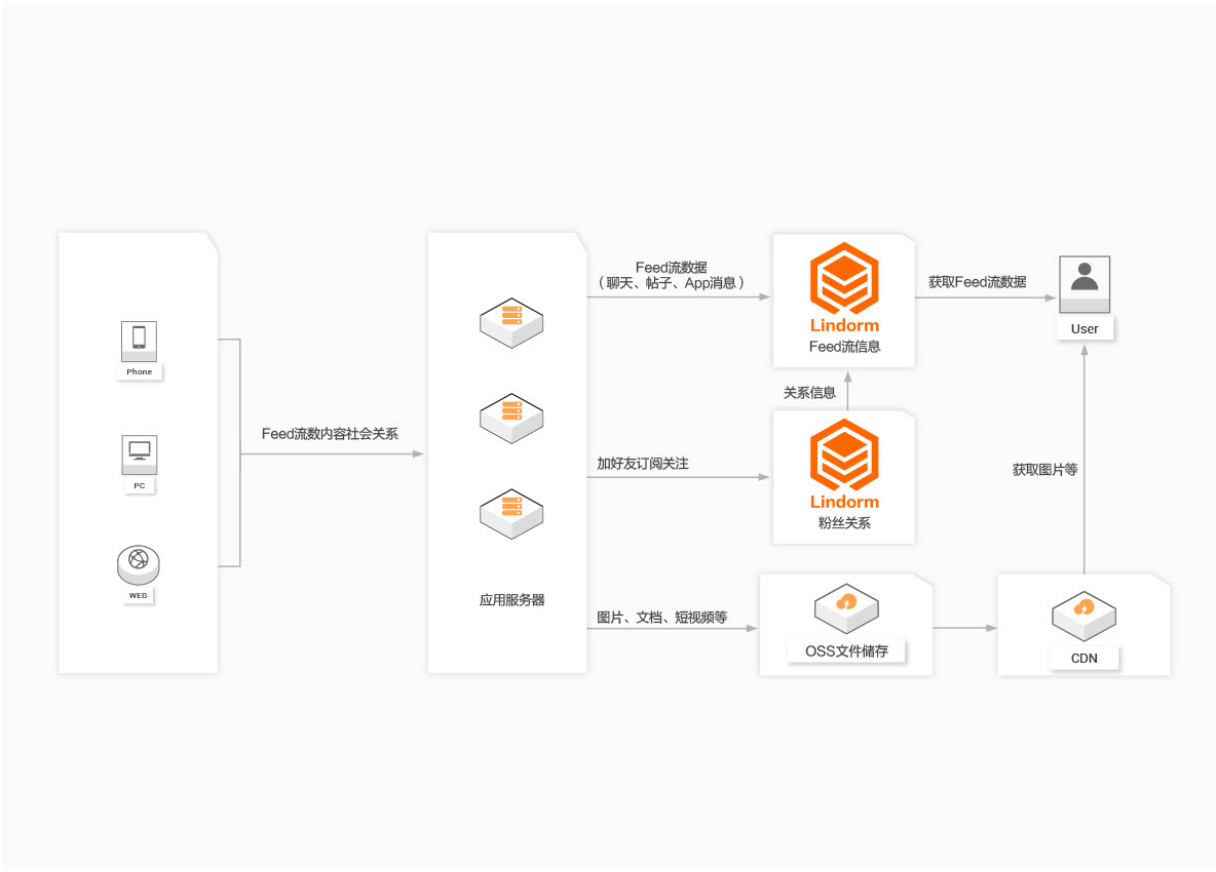


优势

- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 弹性伸缩：存储计算分离架构，支持独立伸缩，自动化扩容。
- 使用灵活：动态列，自由增减特征/标签属性；TTL，数据自动过期；多版本。
- 低延迟：单个毫秒响应，支持双集群请求并发加速。
- 数据通道：通过BDS构建Lindorm与异构计算系统的高效、易用的数据链路。
- 高可用：主备双活容灾，请求自动容错。

互联网社交：高效、稳定的社交Feed流信息存储

使用Lindorm存储社交场景中的聊天、评论、帖子、点赞等重要数据，提供易开发、高可用、延迟的能力，帮助您快速构建稳定可靠的现代社交Feed流系统。



优势

- 易开发：提供社交IM场景专属的Feed流功能，开发效率和运行性能提升一个数量级。
- 高可用：主备双活容灾，请求自动容错。
- 低延迟：单个毫秒响应，支持双集群请求并发加速。
- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 弹性：存储计算分离架构，支持独立伸缩，自动化扩容链路。

6.使用限制

宽表引擎

限制项	限制范围	限制说明
单集群规模	单集群规模最大100，单次最大5台	超过限制联系运维人员处理

时序引擎

时序引擎（TSDB）对某些具体指标进行了约束，您在使用TSDB时注意不要超过相应的限制值，以免系统出现异常。

单个时序引擎节点（30核96 GB），具体的限制项和限制值请参见下表。

限制项	限制值
每秒写入数据点数	48万
每秒查询并发数	400
单次查询返回数据点数量	9000万
单次查询中子查询数量	200
单次子查询的查询时间	90s

搜索引擎

限制项	限制值
单集群规模	建议最大50台
单集群索引表数量	建议最大1000个

7.基本概念

Lindorm

- HBase

  HBase是一个开源的非关系型分布式数据库（NoSQL）,基于谷歌的BigTable建模，是一个高可靠性、高性能、高伸缩的分布式存储系统，使用HBase技术可在廉价PC Server上搭建起大规模结构化存储集群。

- Phoenix

  Phoenix是一个开源的HBase SQL层。它不仅可以使用标准的JDBC API替代HBase client API创建表，插入和查询HBase，也支持二级索引、事务以及多种SQL层优化。

- QPS

  每秒查询率QPS是对一个特定的查询服务器在规定时间内所处理流量多少的衡量标准。

- 二级索引

  HBase的一级索引就是rowkey，rowkey之外的索引能力称为二级索引。

- Hadoop

  Hadoop是一个由Apache基金会所开发的分布式系统基础架构。

时序引擎

- 时间序列数据库

  英文全称为Time Series Database，提供高效存取时序数据和统计分析功能的数据管理系统。

- 时序数据

  基于稳定频率持续产生的一系列指标监测数据。例如，监测某城市的空气质量时，每秒采集一个二氧化硫浓度的值而产生的一系列数据。

- 度量

  监测数据的指标，例如风力和温度。

- 标签

  度量（Metric）虽然指明了要监测的指标项，但没有指明要针对什么对象的该指标项进行监测。标签（Tag）就是用于表明指标项监测针对的具体对象，属于指定度量下的数据子类别。

  一个标签（Tag）由一个标签键（TagKey）和一个对应的标签值（TagValue）组成，例如“城市（TagKey）= 杭州（TagValue）”就是一个标签（Tag）。更多标签示例：机房 = A 、IP = 172.220.110.1。

  > **说明** 当标签键和标签值都相同才算同一个标签；标签键相同，标签值不同，则不是同一个标签。

  在监测数据的时候，指定度量是“气温”，标签是“城市 = 杭州”，则监测的就是杭州市的气温。

- 标签键

  为指标项监测指定的对象类型（会有对应的标签值来定位该对象类型下的具体对象），例如国家、省份、城市、机房、IP 等。

- 标签值

标签键（TagKey）对应的值。例如，当标签键（TagKey）是“国家”时，可指定标签值（TagValue）为“中国”。

- 值

  度量对应的值，例如15 级（风力）和20 ℃（温度）。

- 时间戳

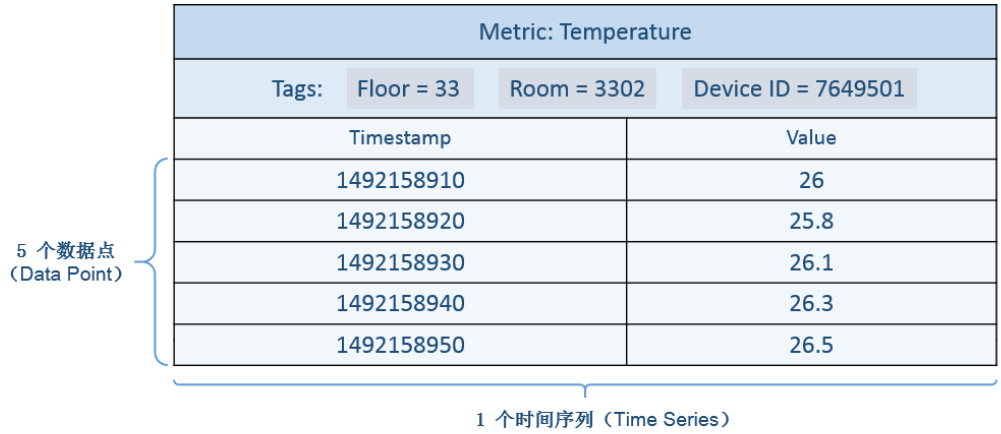
  数据（度量值）产生的时间点。

- 时间点

  针对监测对象的某项指标（由度量和标签定义）按特定时间间隔（连续的时间戳）采集的每个度量值就是一个数据点。“一个度量 + N 个标签（N >= 1）+ 一个时间戳 + 一个值”定义一个数据点。

- 时间序列

  也称作时间线。针对监测对象的某项指标（由度量和标签定义）按特定时间间隔（连续的时间戳）采集的一系列数据点就是一个时间序列。“一个度量 + N 个标签（N >= 1）+ N 个时间戳（N >= 1）+ N 个值（N >= 1）”定义一个时间序列。示意图如下：



- 时间线

  等同于时间序列的概念。

- 时间精度

  时间线数据的写入时间精度——秒、分钟、小时或者其他稳定时间频度。例如，每秒一个温度数据的采集频度，每 5 分钟一个负载数据的采集频度。

- 数据组

  如果需要对比不同监测对象（由标签定义）的同一指标（由度量定义）的数据，可以按标签这些数据分成不同的数据组。例如，将温度指标数据按照不同城市进行分组查询，操作类似于该SQL语句：select temperature from xxx group by city where city in （shanghai, hangzhou)。

- 空间聚合

  当同一个度量（Metric）的查询有多条时间线产生（多个指标采集设备），那么为了将空间的多维数据展现为成同一条时间线，需要进行合并计算，例如，当选定了某个城市某个城区的污染指数时，通常将各个环境监测点的指标数据平均值作为最终区域的指标数据，这个计算过程就是空间聚合。

- 降采样

当查询的时间区间跨度较长而原始数据时间精度较细时，为了满足业务需求的场景、提升查询效率，就会降低数据的查询展现精度，这就叫做降采样，比如按秒采集一年的数据，按照天级别查询展现。

- 数据时效

  数据时效是设置的数据的实际有效期，超过有效期的数据会被自动释放。