Kafka 分级存储在腾讯云的实践与演进

腾讯云 / ShilinLu(鲁仕林)

自我介绍

腾讯云专家工程师，腾讯云消息队列 Kafka 内核负责人，专注于中间件、消息队列、Serverless 领域。

目录

1. Kafka 架构遇到的问题与挑战

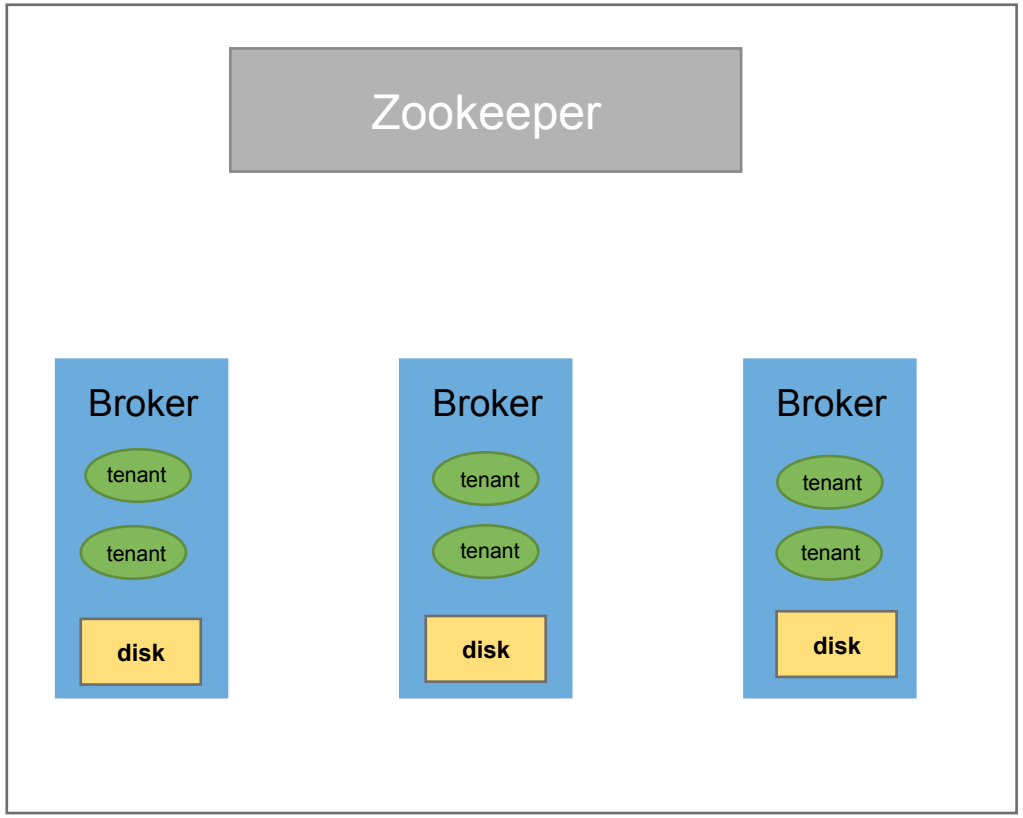
2. Kafka 弹性架构方案类比

3. Kafka 分级存储架构及原理

4. 腾讯云的落地与实践

Kafka 架构遇到的问题与挑战

Kafka 架构

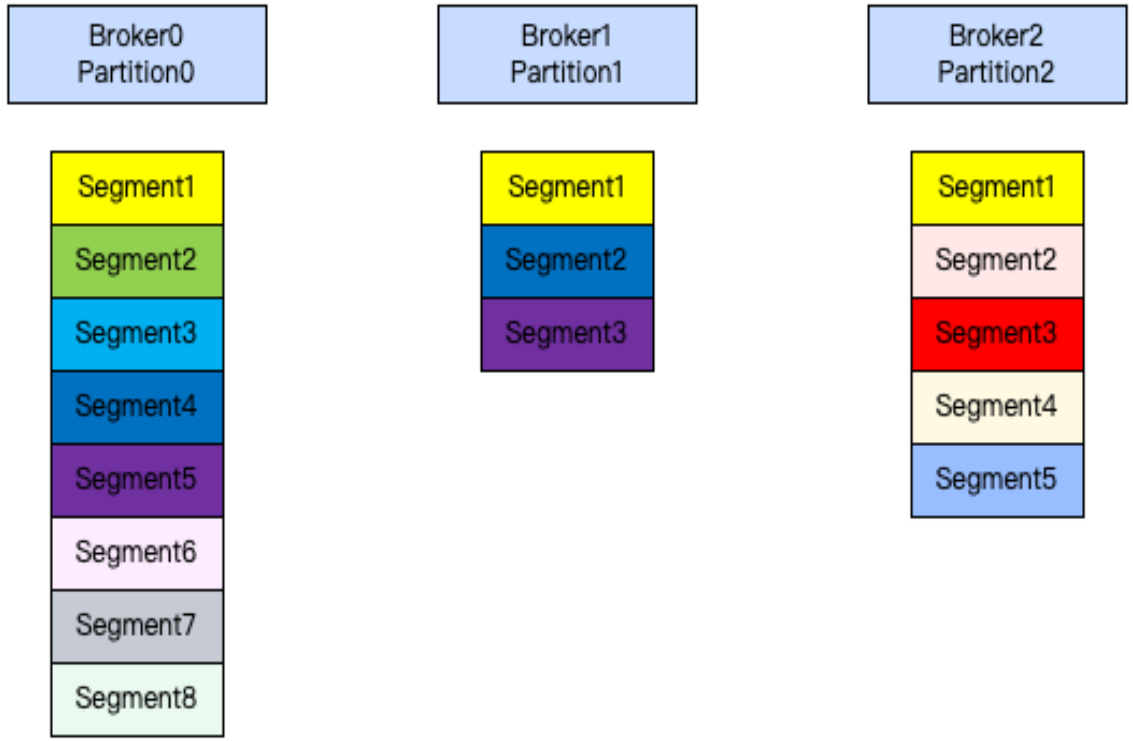


基于 Zookeeper 构建的集群

1. 基于 Zookeeper/Kraft 构建集群
2. 物理机/VM + 本地磁盘构建集群

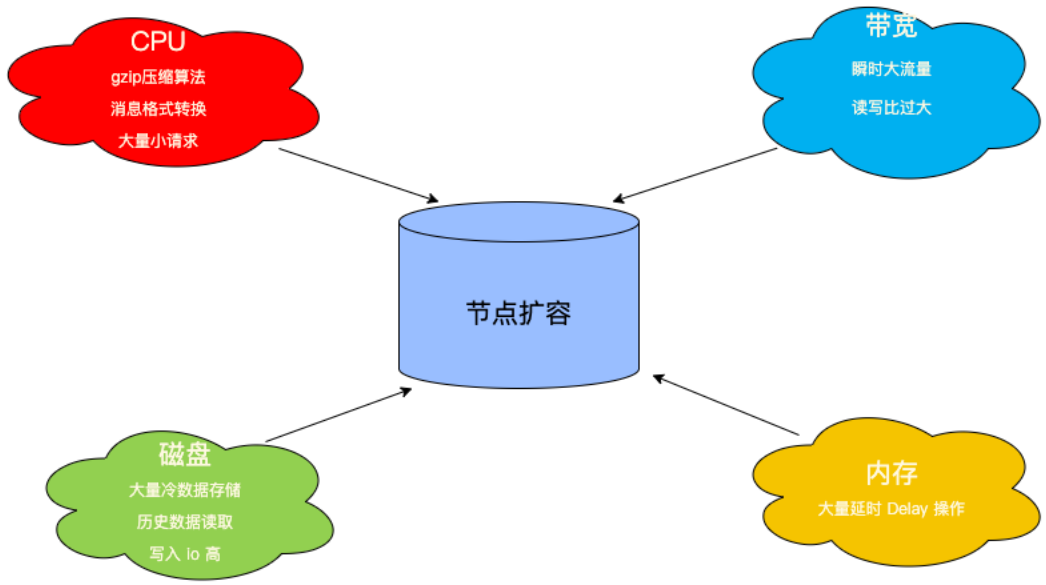
1. 本地状态重，运维困难
2. 节点维度扩容，资源浪费
3. 历史数据回溯，影响集群读写 SLA

运维难度大



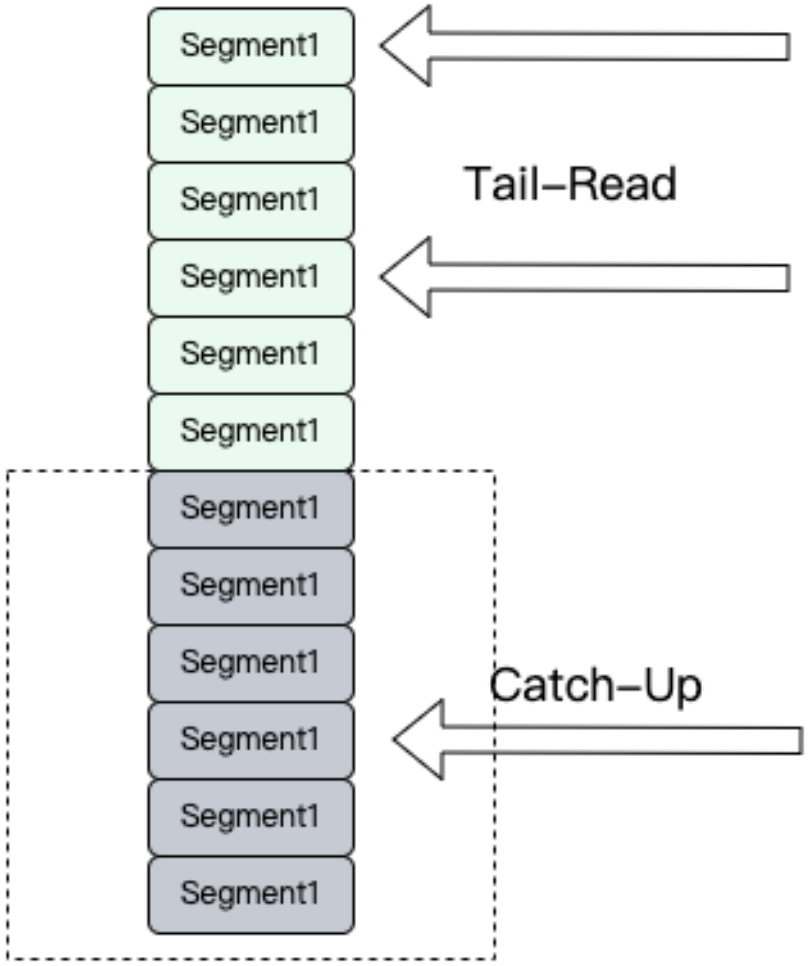
1. 节点间数据分布不均
2. 节点系统指标瓶颈(带宽、磁盘、CPU等)
3. 节点内数据多，迁移较慢且影响读写

资源浪费



- CPU
  - 压缩算法(Gzip、Snappy、Zstd等)
  - 消息格式转换(V0、V1、V2)
  - 生产/消费版本或者配置
- 磁盘
  - 存储空间，大量冷数据存储
  - 历史数据 Tail-Read 读取，磁盘 IO 瓶颈
  - HDD 磁盘导致大吞吐下磁盘 IO 瓶颈
- 带宽
  - 瞬时流量突刺
  - 集群扇出度/读写比大
- 内存
  - Broker 限流后产生大量的 Delay 操作

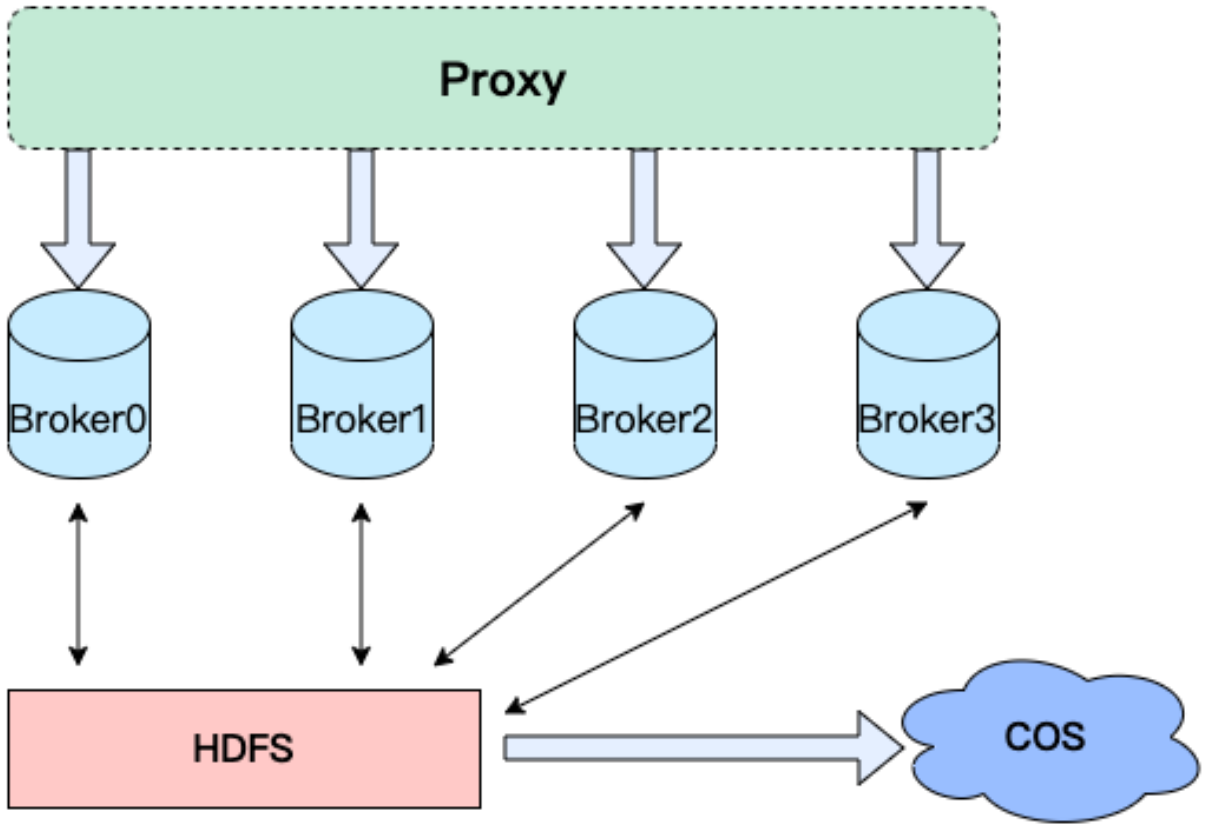
历史数据影响 SLA



- Catch-Up 读
  - 污染 Page-Cache，影响其他读流程
  - 直接读取磁盘，磁盘 IO 高，影响写入刷盘

Kafka 弹性架构方案类比

存储计算分离架构

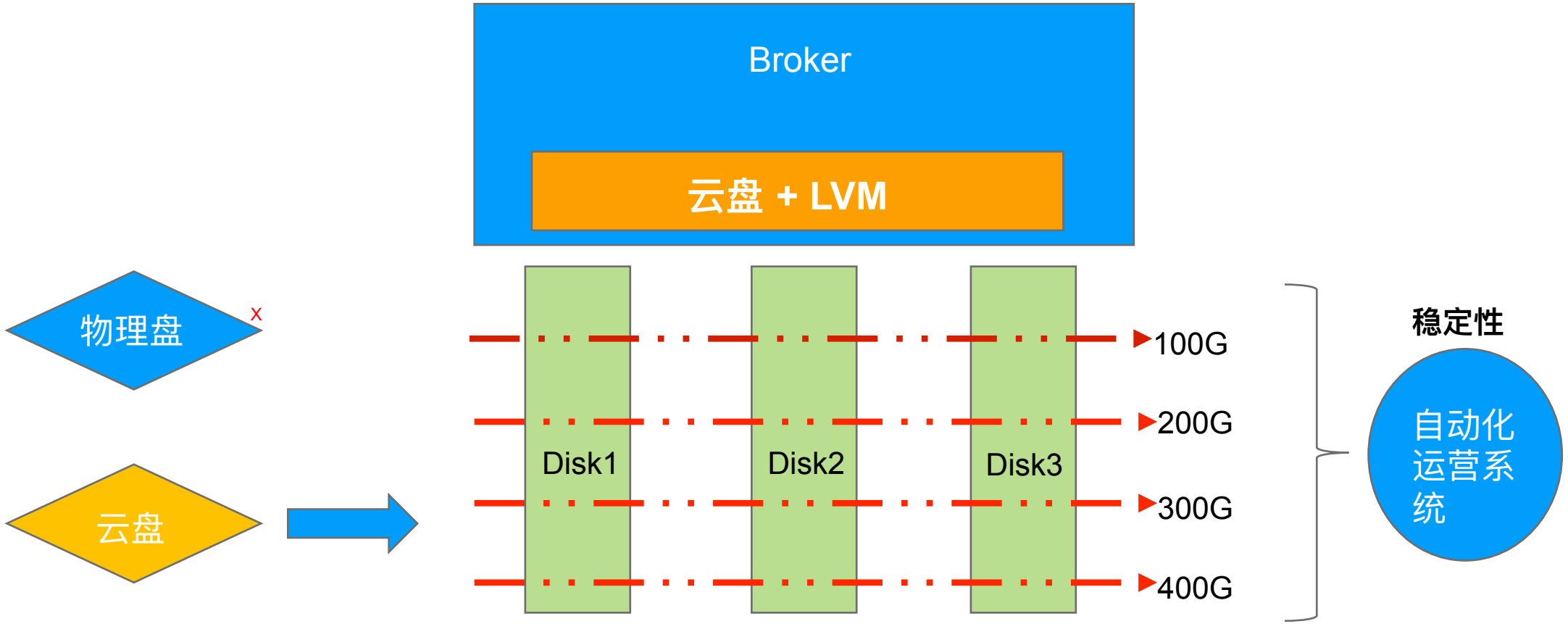


- Proxy
  - 统一接入
  - 协议转换
  - 功能上浮
- Broker
  - 存储底座适配
  - Partition Holder
  - 负载均衡
- 存储层
  - HDFS
  - COS/S3

弹性的本地存储架构

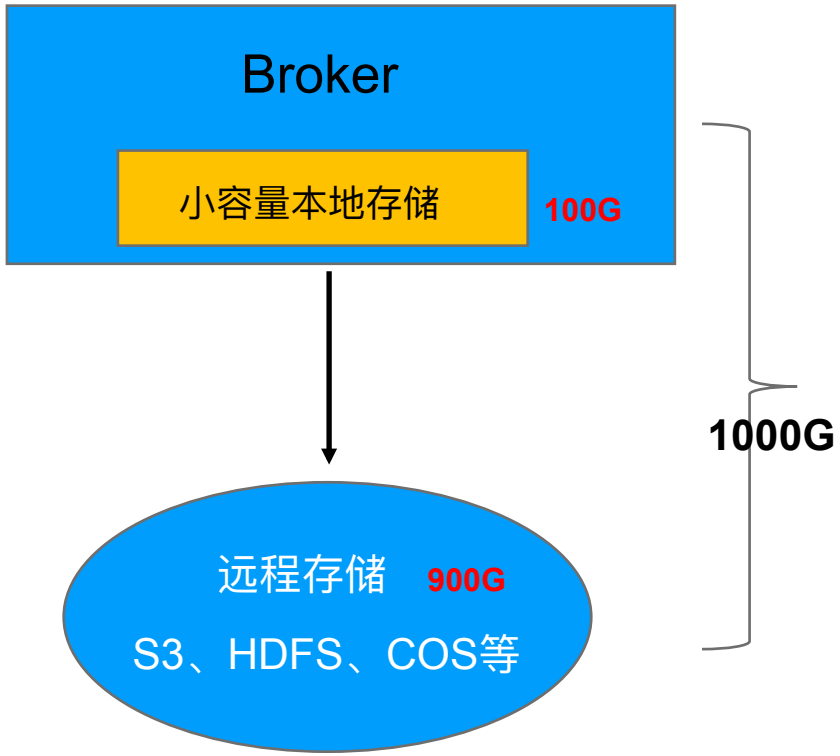
1. 核心是：**云盘 + 运营调度系统**

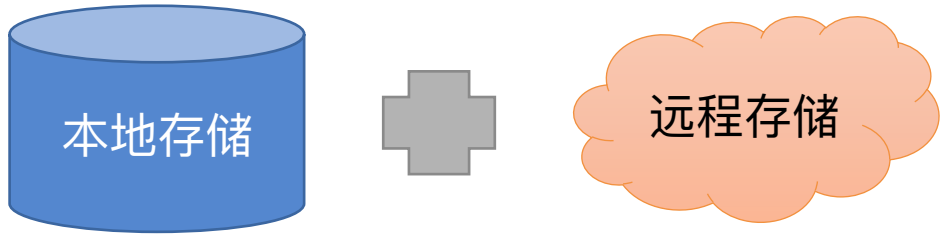
2. 多块云盘 + LVM逻辑卷 + 运营调度系统



弹性的远程存储架构

1. 通过更廉价的存储进一步降低成本
2. 1000 > 100*1 + 900*0.3 = **370**





- 本地弹性存储
  - 本地存储服务写流量/Tail-Read 读，提供与原生 Kafka 一致的延时、可用性和一致性
  - 远程存储故障或者性能衰退，本地存储支持弹性扩容提供读写服务
- 远程弹性存储
  - 远程存储服务 Catch-Up 读，冷热数据分离
  - 按需使用，按需计费
  - 支持多模存储，多介质存储

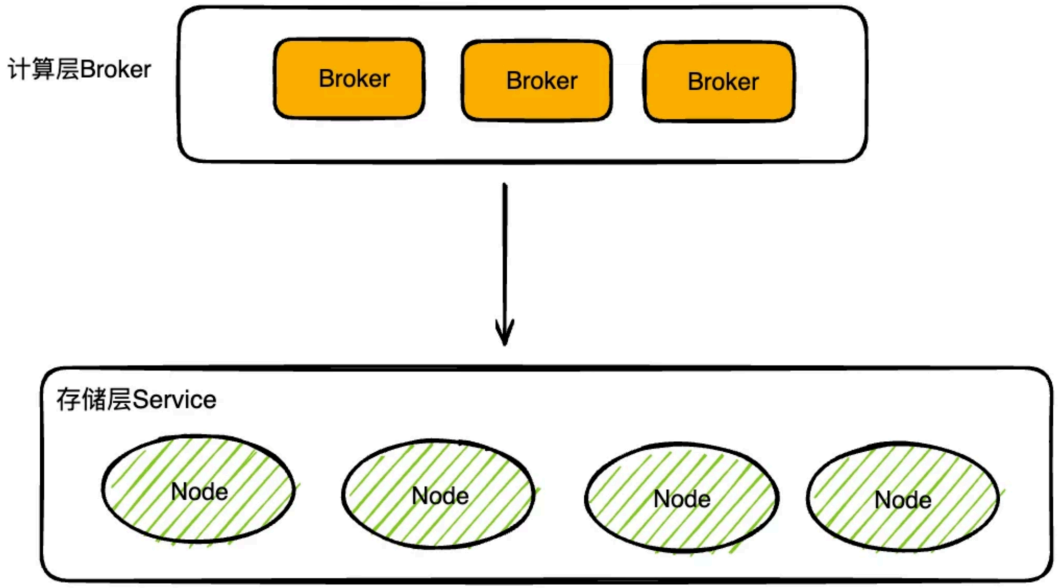
实现计算层快速弹性的方案

少迁移数据，不迁移数据

1. **尽量迁移少量的老数据 => 【分层存储】**
2. 计算存储分离，弹性的计算层

1. 云盘快速挂载新节点
2. 使用分布式文件系统存储数据
3. **计算存储分离架构**

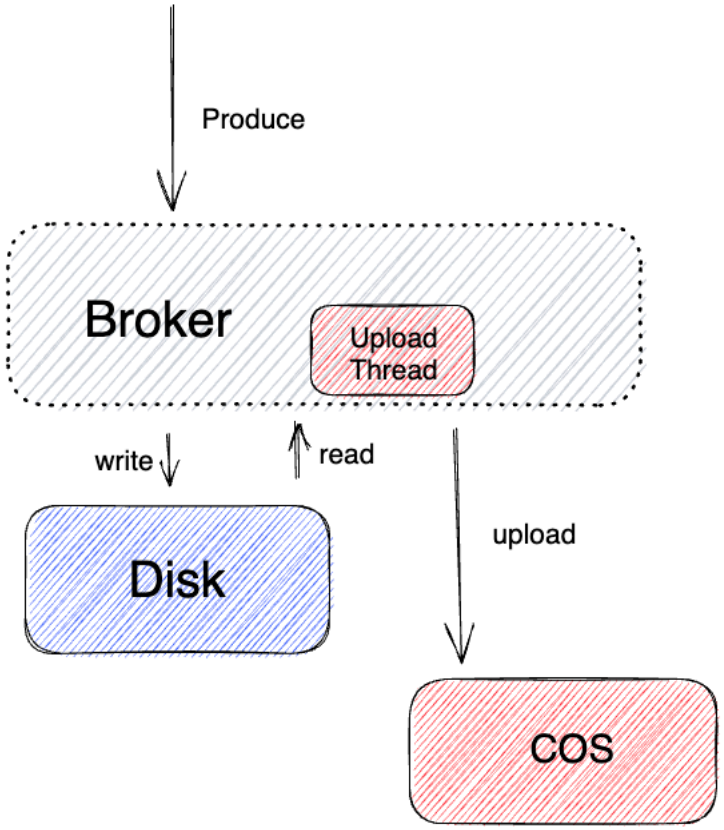
核心思路： 有状态计算层 => 无状态计算层



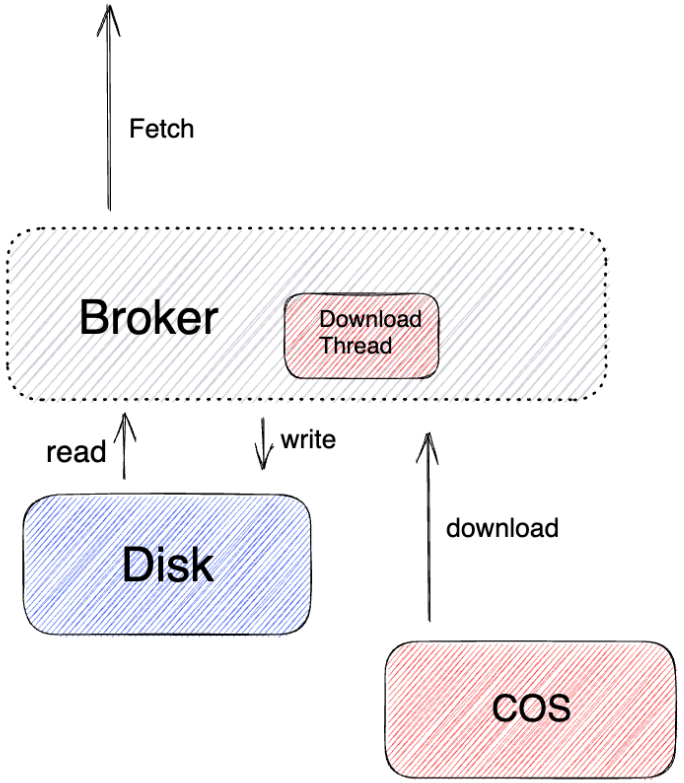
独立的计算层和存储层

Kafka 分级存储架构

分级存储读写流程

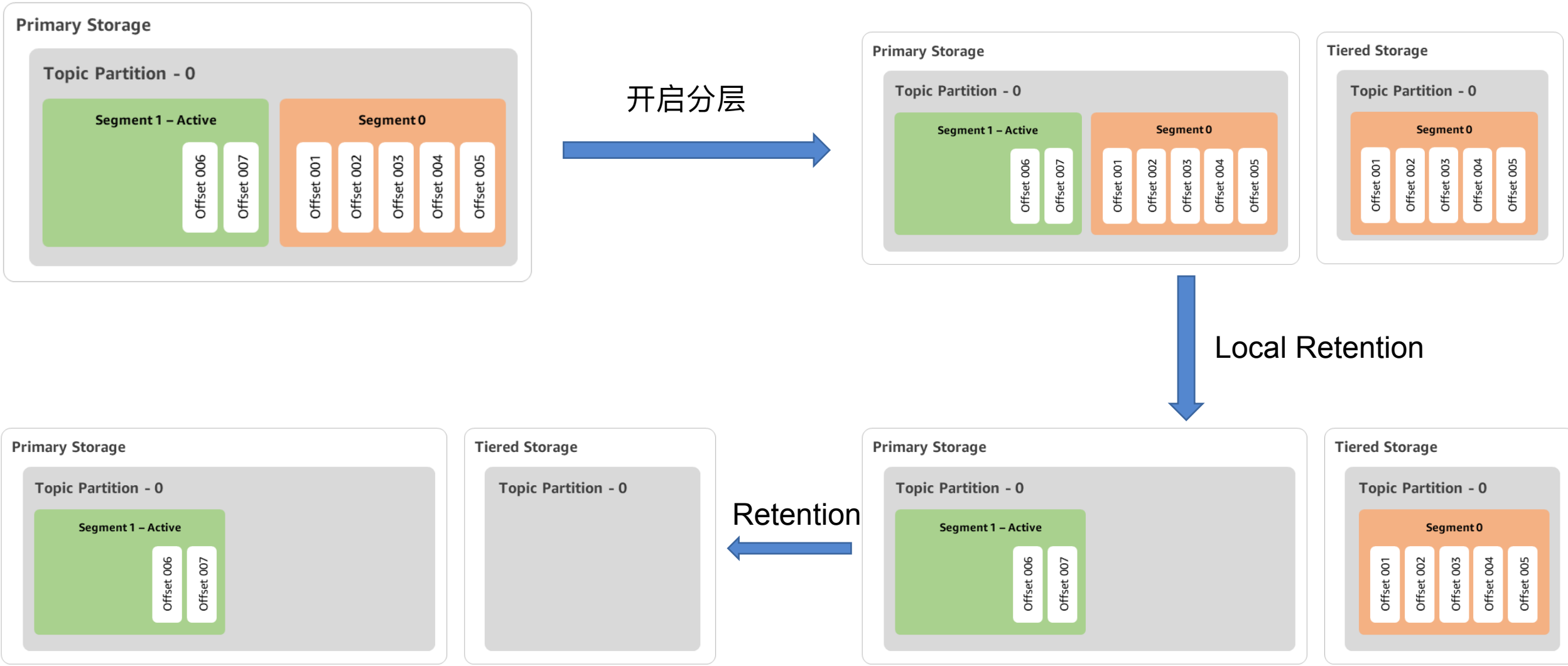


分层生产流程

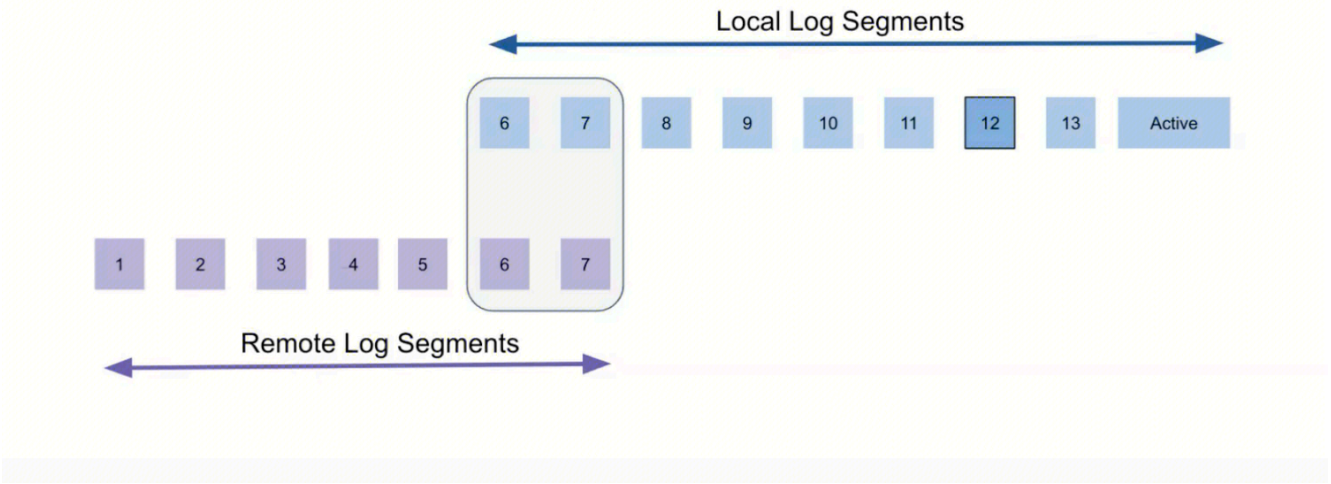


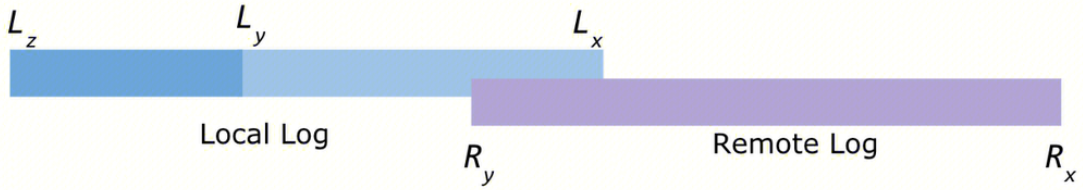
分层消费流程

数据生命周期


Primary Storage
Topic Partition - 0
Segment 1 – Active
Offset 006
Offset 007
Segment 0
Offset 001
Offset 002
Offset 003
Offset 004
Offset 005
开启分层
Primary Storage
Topic Partition - 0
Segment 1 – Active
Offset 006
Offset 007
Segment 0
Offset 001
Offset 002
Offset 003
Offset 004
Offset 005
Tiered Storage
Topic Partition - 0
Segment 0
Offset 001
Offset 002
Offset 003
Offset 004
Offset 005
Local Retention
Primary Storage
Topic Partition - 0
Segment 1 – Active
Offset 006
Offset 007
Tiered Storage
Topic Partition - 0
Retention
Primary Storage
Topic Partition - 0
Segment 1 – Active
Offset 006
Offset 007
Tiered Storage
Topic Partition - 0
Segment 0
Offset 001
Offset 002
Offset 003
Offset 004
Offset 005

Offset 约束





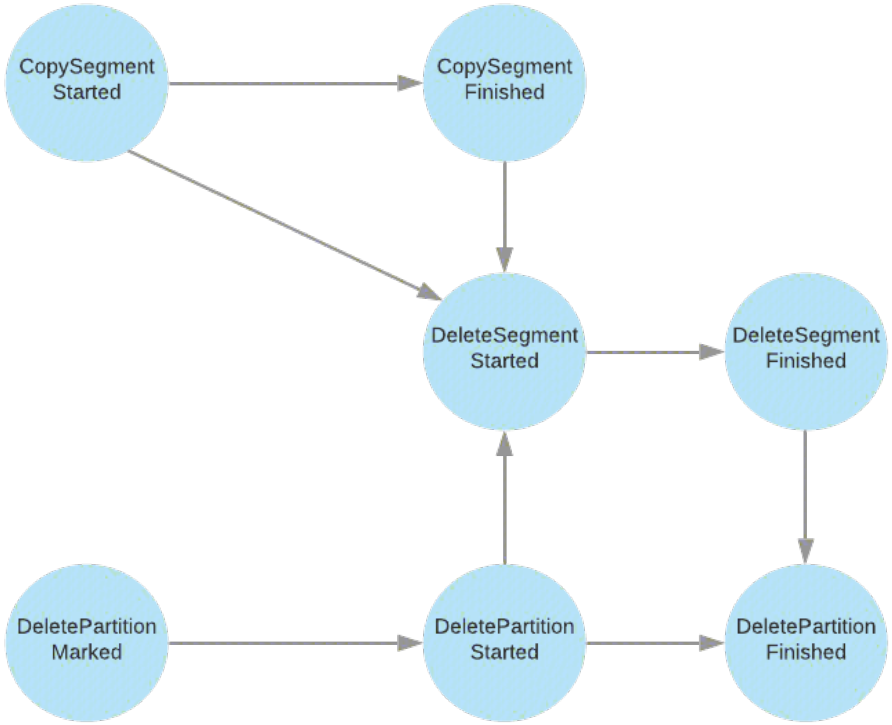
Lx = Local log start offset L_z = Local log end offset L_y = Last stable offset(LSO)

R_y = Remote log end offset R_x = Remote log start offset

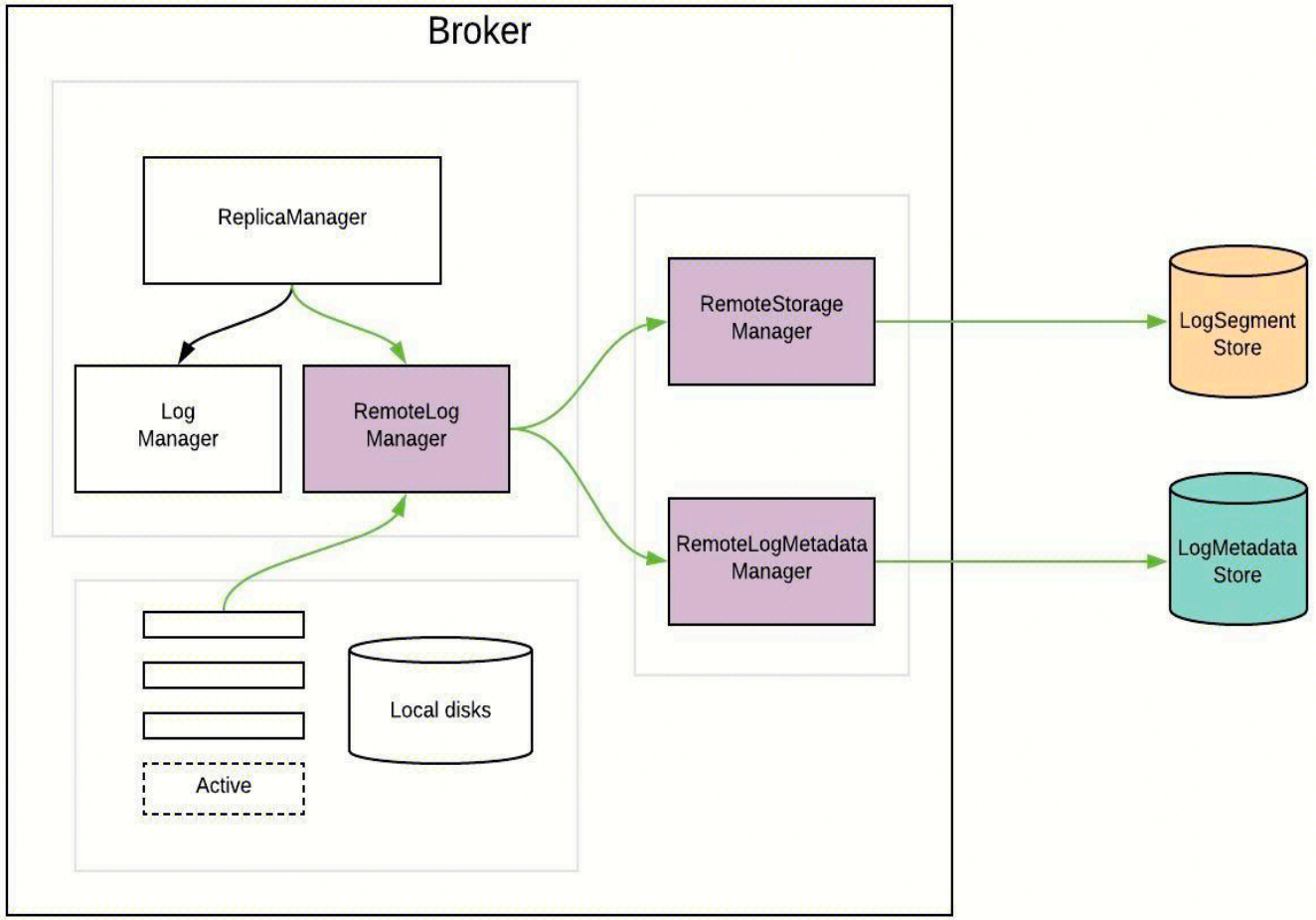
$L_z >= L_y >= Lx$ and $L_y >= R_y >= R_x$

Segment 状态流转

- Copy Segment
- Delete Segment
- Delete Partition



分级存储架构

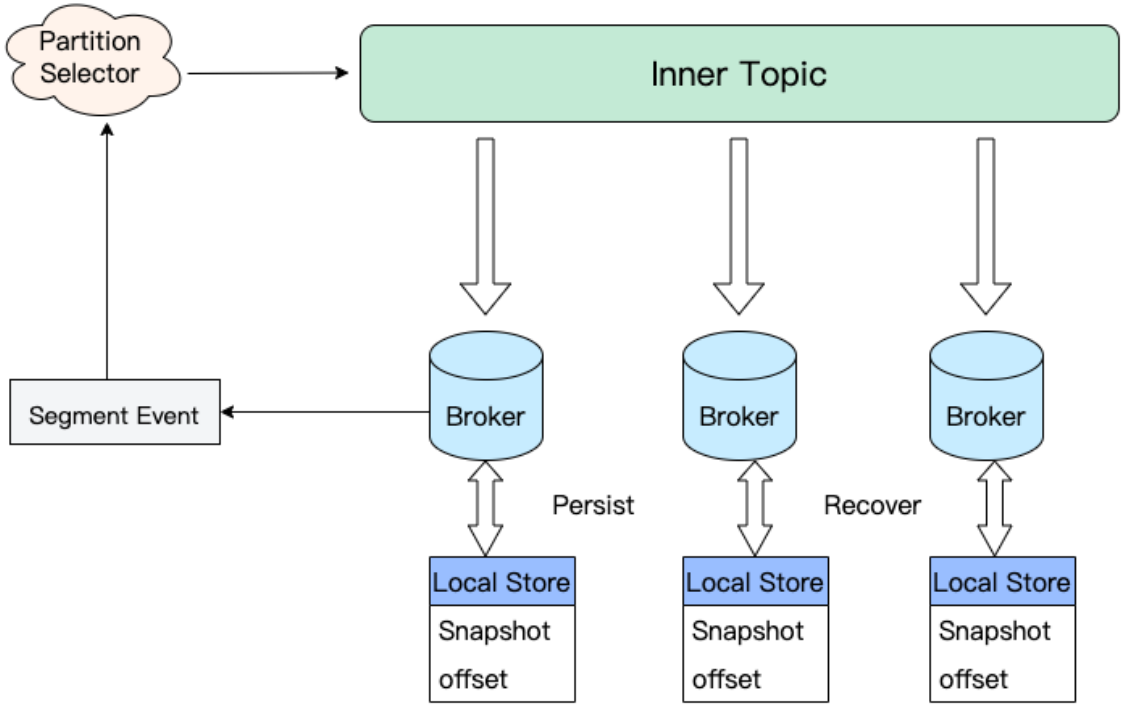


- RemoteStorageManager
  - 远程存储抽象类
  - 上传、下载、删除等接口

- RemoteLogMetadataManager
  - 元数据同步抽象类
  - 元数据初始化、更新、删除等接口

腾讯云的落地与实践

Segment 元数据管理

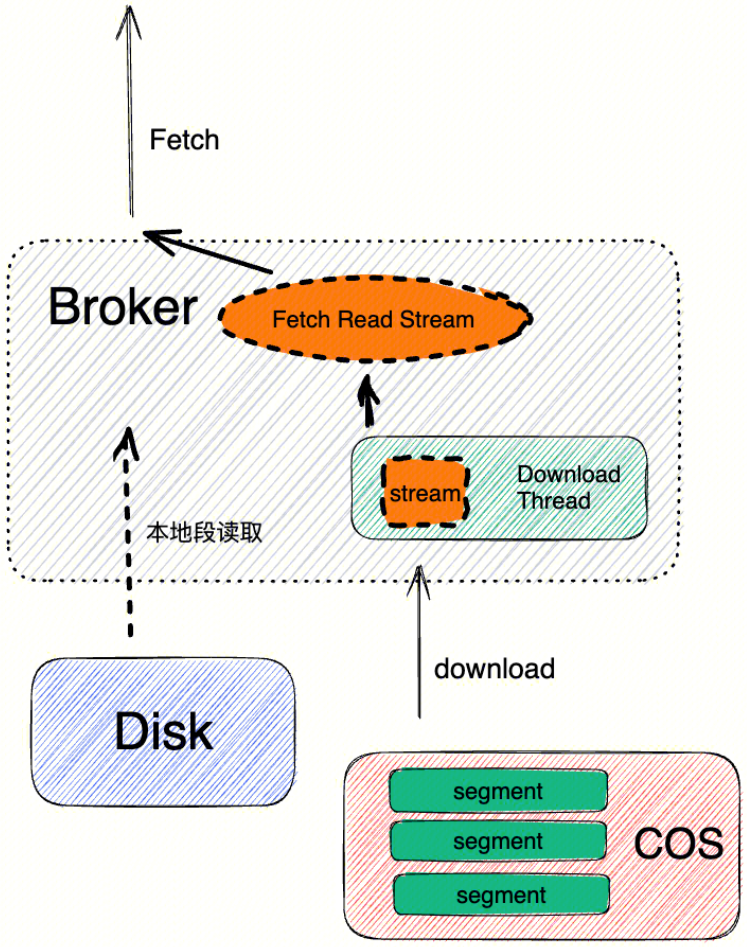


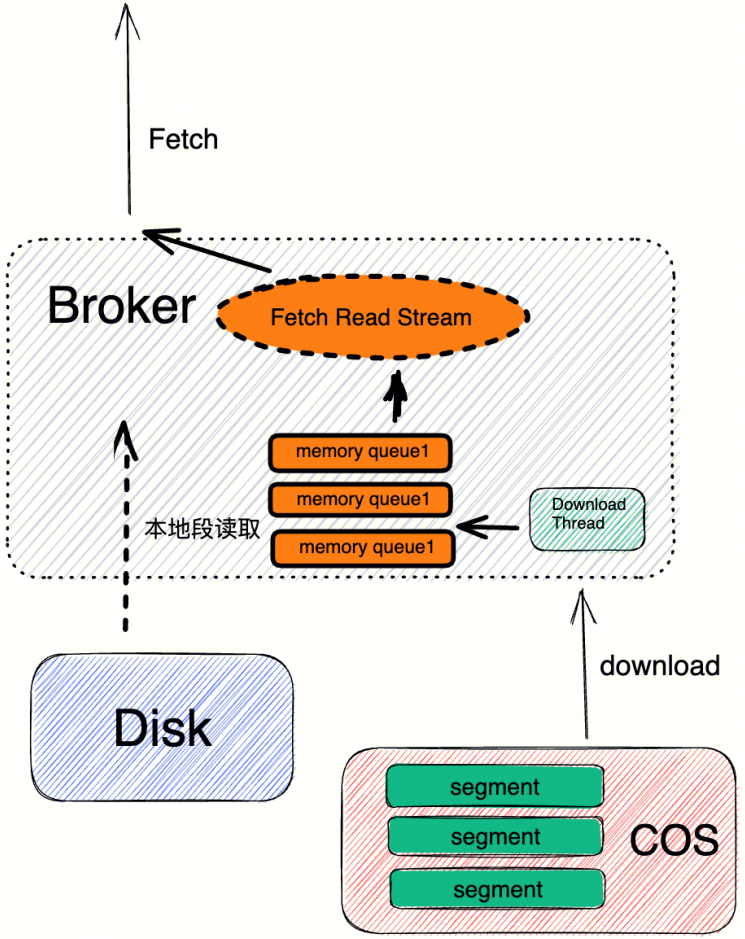
1. Inner Topic 作为 WAL 存储 Segment Event

2. Broker 消费 Inner Topic 构建 Remote Metadata StateMachine

3. Remote Metadata StateMachine 定期 Snapshot 到 broker 本地

消费性能

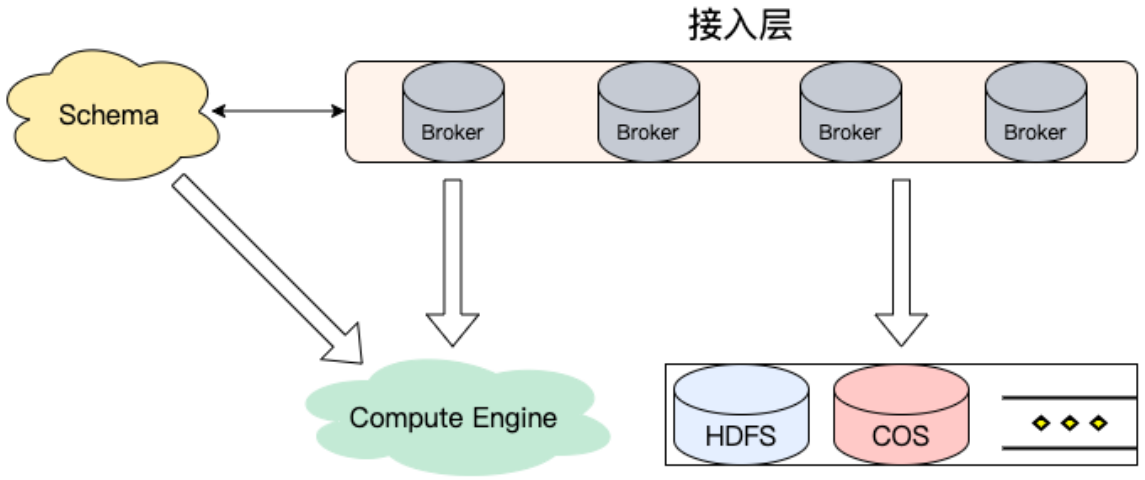

Fetch
Broker
Fetch Read Stream
stream
Download Thread
本地段读取
download
Disk
segment
segment
segment
COS


Fetch
Broker
Fetch Read Stream
memory queue1
memory queue1
memory queue1
Download Thread
本地段读取
download
Disk
segment
segment
segment
COS

隔离性

- 硬盘
  - 独立的 IO 盘，减少磁盘 IO 影响
- CPU
  - 核心线程进行绑核
  - 线程隔离
- 带宽
  - 上传/下载限流
  - 上传/下载任务并行度控制
- 内存
  - 使用堆外内存
  - ByteBuffer 复用
- 回滚
  - 暂停分层上传能力
  - 按需暂停分层数据下载能力
  - 运营系统自动扩容云盘
  - 支持 Topic/集群维度回滚

展望



- Schema
  - 消息格式存储(Protobuf、Json)
- 接入层
  - 流量接入，无状态可横向扩展
- Compute Engine
  - 格式转换计算层，如：行列格式转换(parquet)
  - Parquet 直接对接 Hudi/DeltaLake
  - 云 Api 获取文件
- 存储层
  - 多模存储，数据分级
  - 软硬件结合，探索新的存储系统

OpenEuler

感谢聆听！

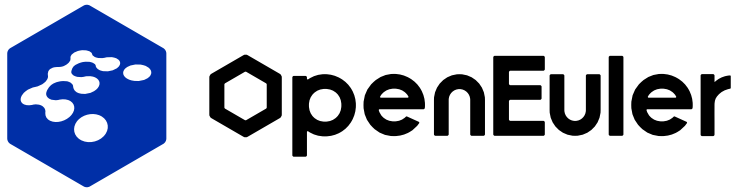
openEuler sig 用户群

腾讯云中间件
官方公众号

鲁仕林

浙江 宁波

扫一扫上面的二维码图案，加我为朋友。


OpenEuler