

开源PolarDB总体架构 和企业级特性



钉钉扫码进群沟通交流

01 PolarDB总体架构设计

- 云原生架构
- HTAP架构
- 三节点高可用架构

02 PolarDB企业级特性

- 高性能
- 高可用
- 安全

03 PolarDB开源社区

- 开源策略
- 开源进展

01

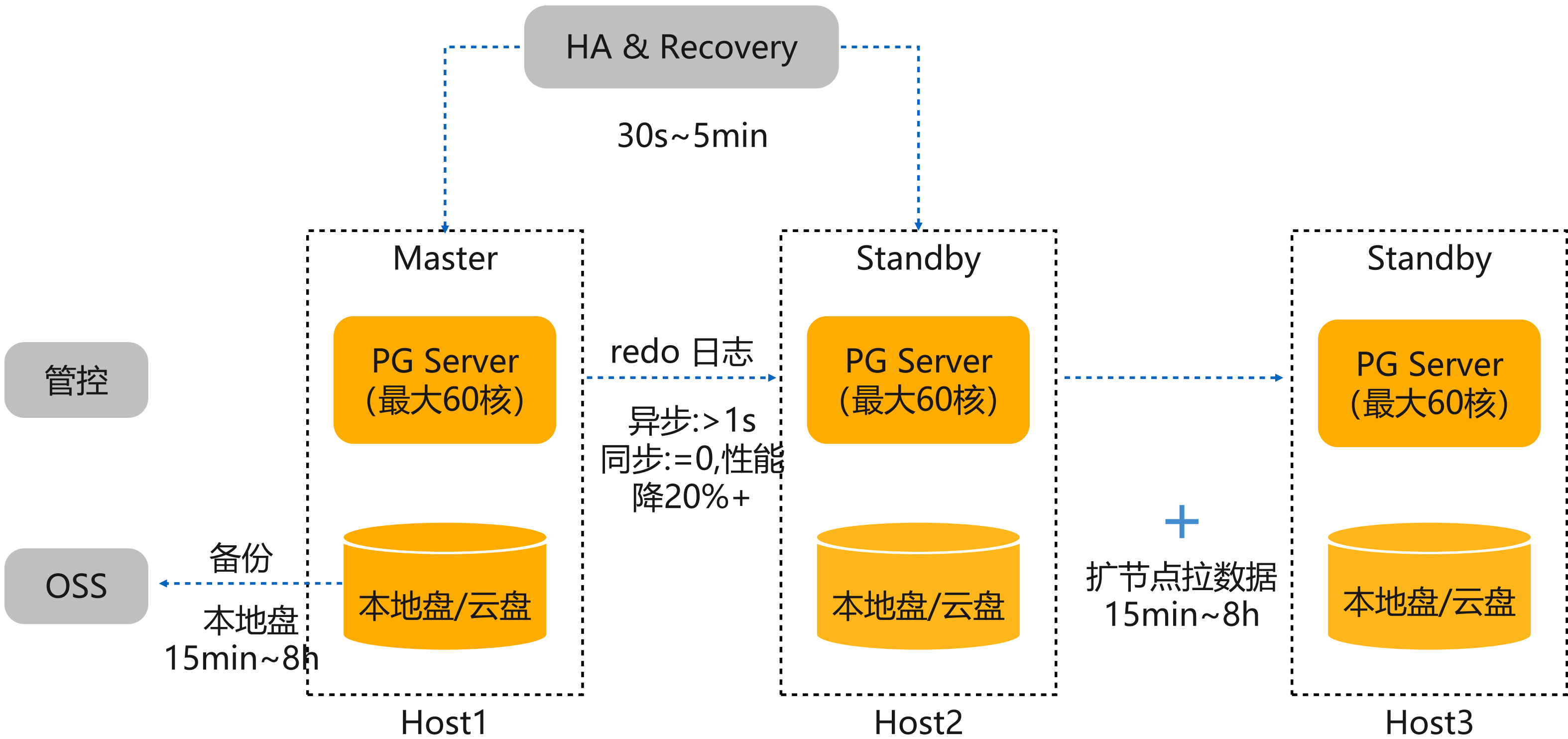
PolarDB总体架构

PolarDB总体架构 - 云原生架构



传统数据库的痛点

- 扩展性差：加节点小时级
- 可靠性差：RPO != 0
- 可用性差：RTO 30s-5min
- 成本高：
 - 存储成本随节点数线性增加
 - 预占资源



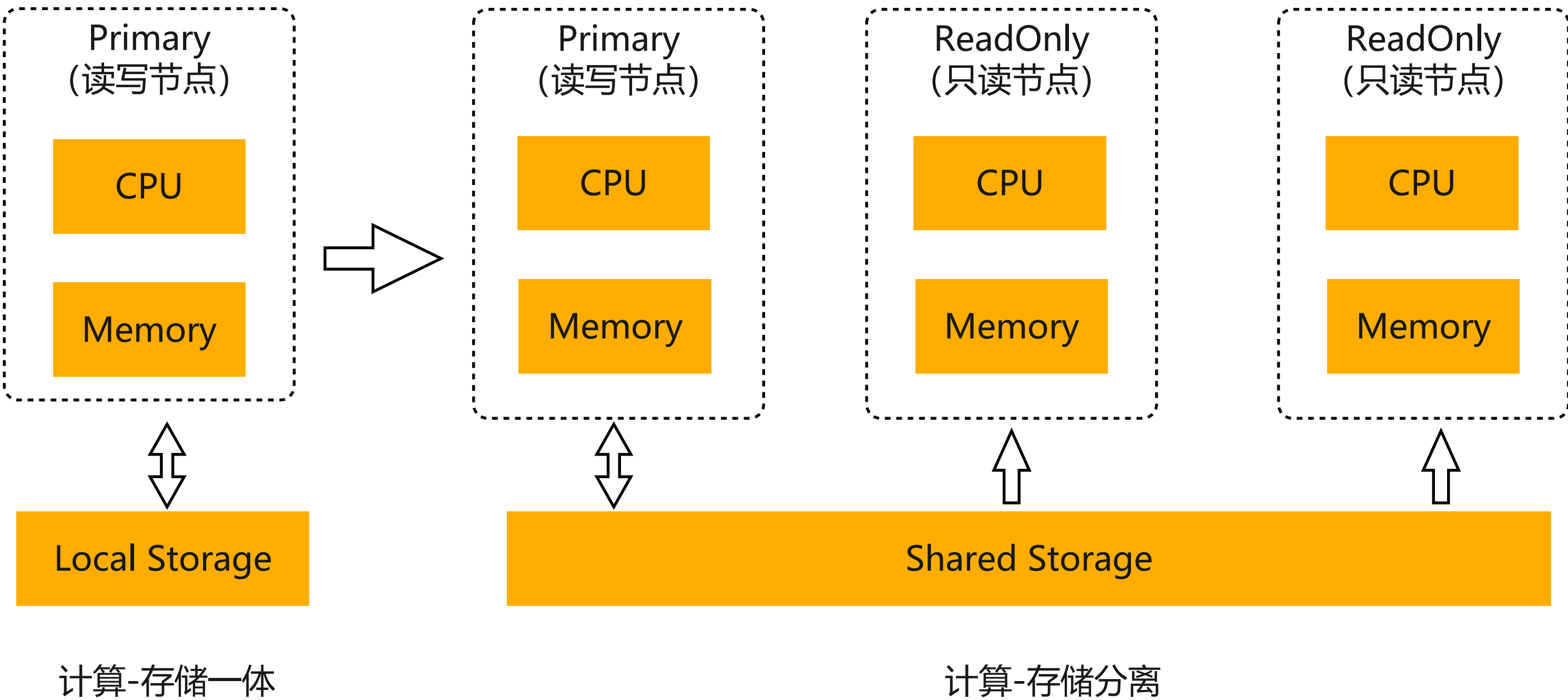
PolarDB总体架构 - 云原生架构



计算存储分离

架构优势

- 扩展性：存储计算分离，极致弹性
- 成本：共享一份数据，存储成本低
- 易用性：具备分布式的优势和单机体感
- 可靠性：三副本、秒级备份



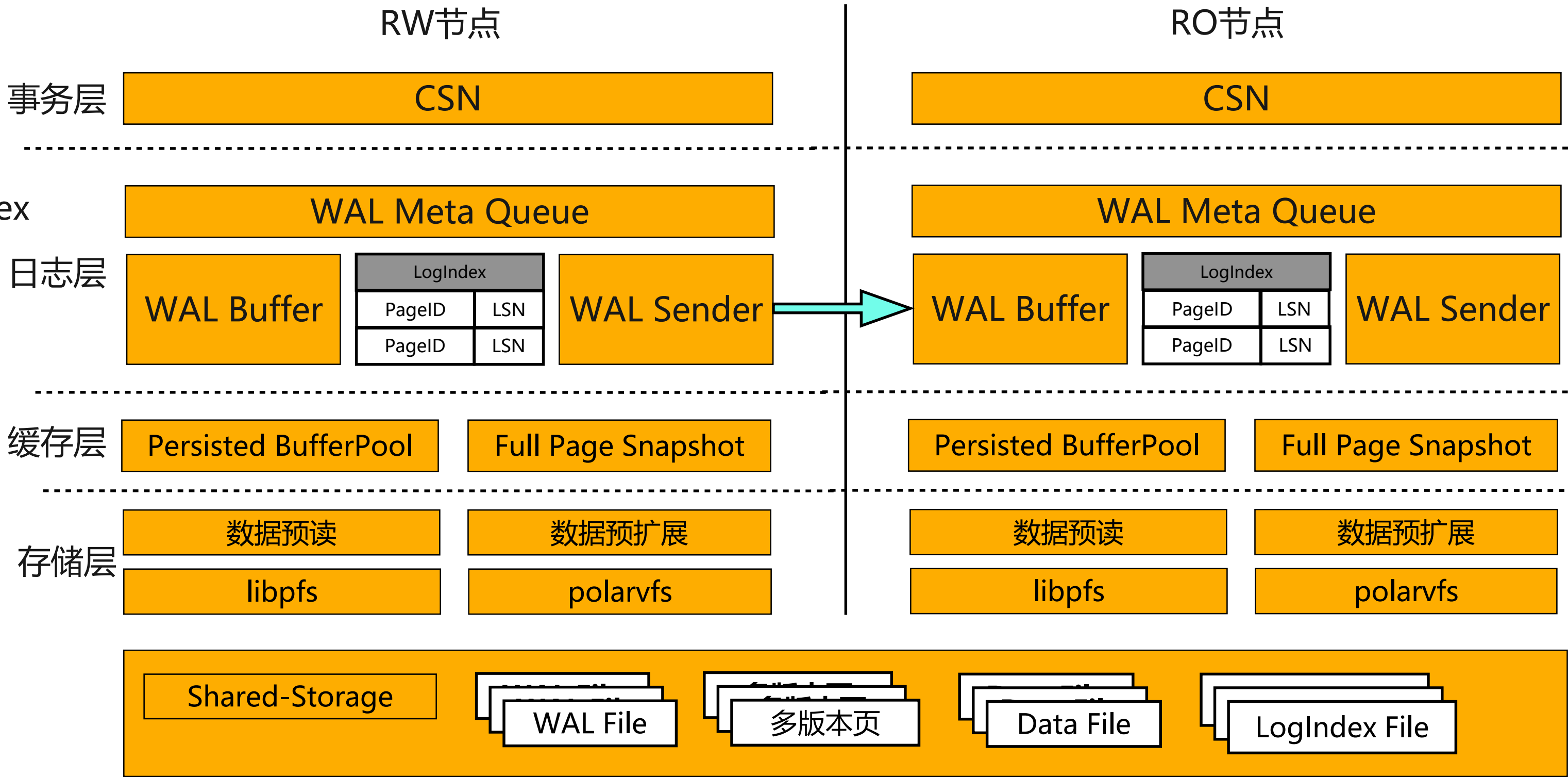
PolarDB总体架构 - 云原生架构



计算存储分离的模块栈

模块栈 - 关键技术点

- 事务层：CSN快照
- 日志层：复制WAL Meta、Lazy回放，并行回放，LogIndex
- 缓存层：常驻BufferPool、多版本页面
- 存储层：DirectIO、数据预读、预扩展、PolarVFS



PolarDB总体架构 - HTAP架构



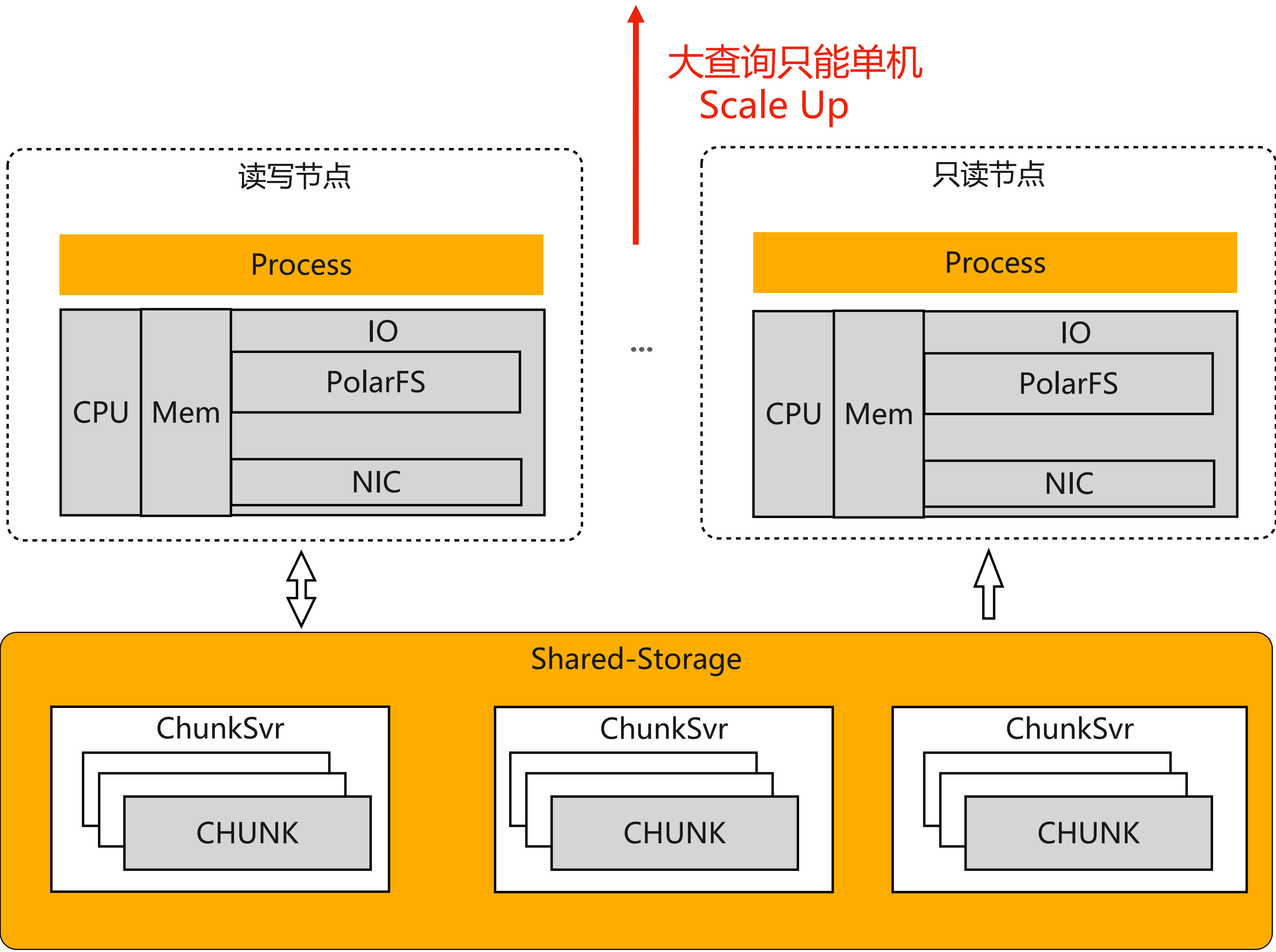
传统解决方案的痛点

计算存储分离架构处理AP查询的挑战:

- ❑ 无法发挥多个计算节点的CPU/MEM, 不能Scale Out
- ❑ 无法发挥共享存储池的大吞吐能力

业界HTAP解决方案:

- ◆ TP和AP在存储/计算上都分离
 - TP数据导入到AP系统, 有延迟, 时效性不高
 - 增加冗余AP存储, 成本增加
 - 增加AP系统, 运维难度增加
- ◆ TP和AP在存储和计算资源都共享
 - AP查询、TP查询或多或少相互影响
 - AP查询比重增大时, 无法快速弹性scale out



PolarDB总体架构 - HTAP架构



基于共享存储的分布式计算引擎 - 业界首创

✓ 一体化存储 (毫秒级数据新鲜度)

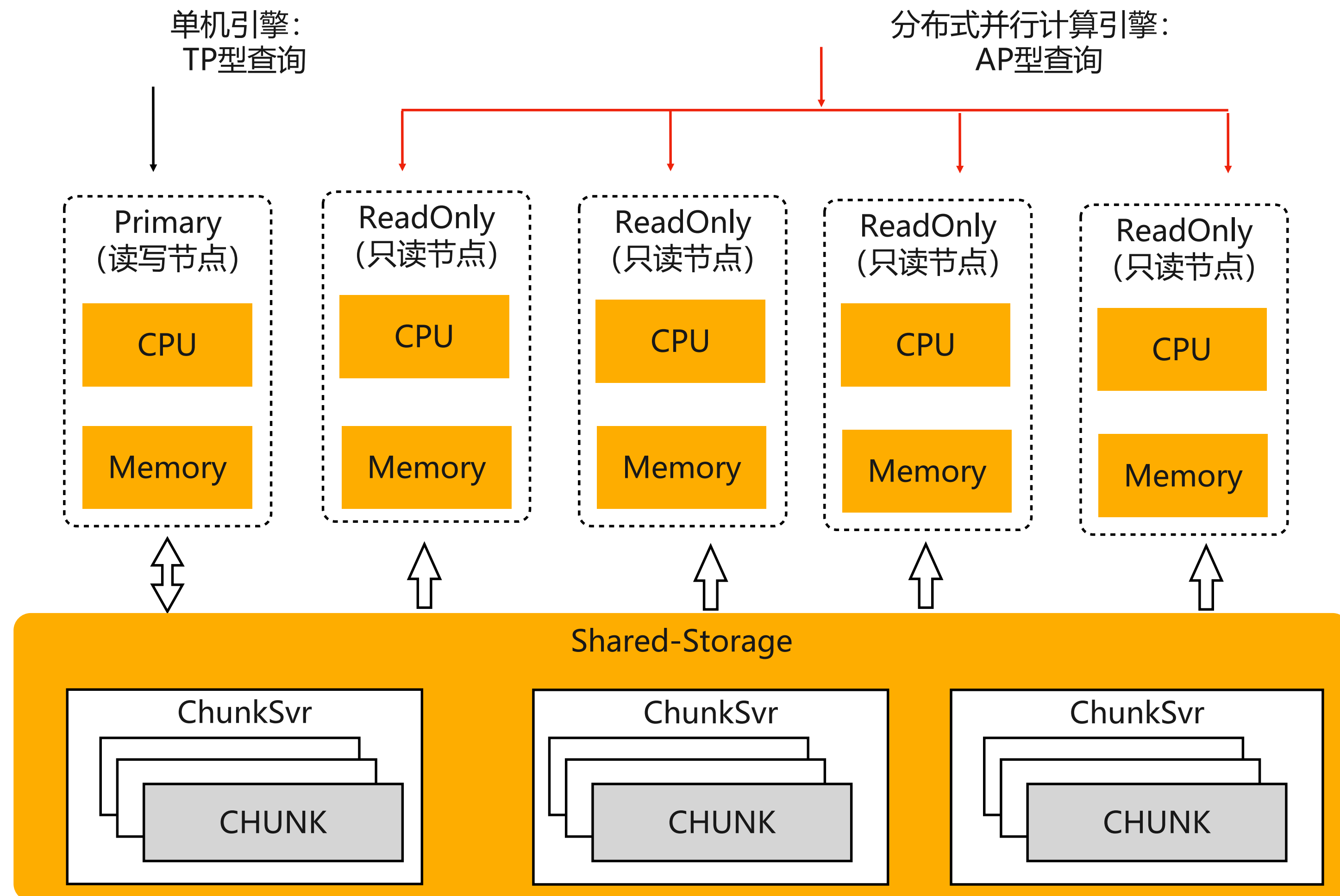
- TP/AP共享一套存储数据, 减少存储成本, 提高查询时效

✓ TP/AP物理隔离 (杜绝CPU/MEM相互影响) :

- 单机执行: 在RW/RO节点上, 处理高并发TP查询
- 分布式MPP执行: 在RO节点, 处理复杂AP查询

✓ Serverless弹性扩展

- 任何一个RO节点均可发起MPP查询
- ScaleUp: SQL级别弹性调整单机执行并行度
- ScaleOut: SQL级别弹性调整MPP执行节点范围



PolarDB总体架构 - HTAP架构



基于共享存储的分布式计算引擎 - 模块栈

分布式执行引擎：

- 分布式优化器
- 分布式执行器
- 事务一致性
- SQL全兼容

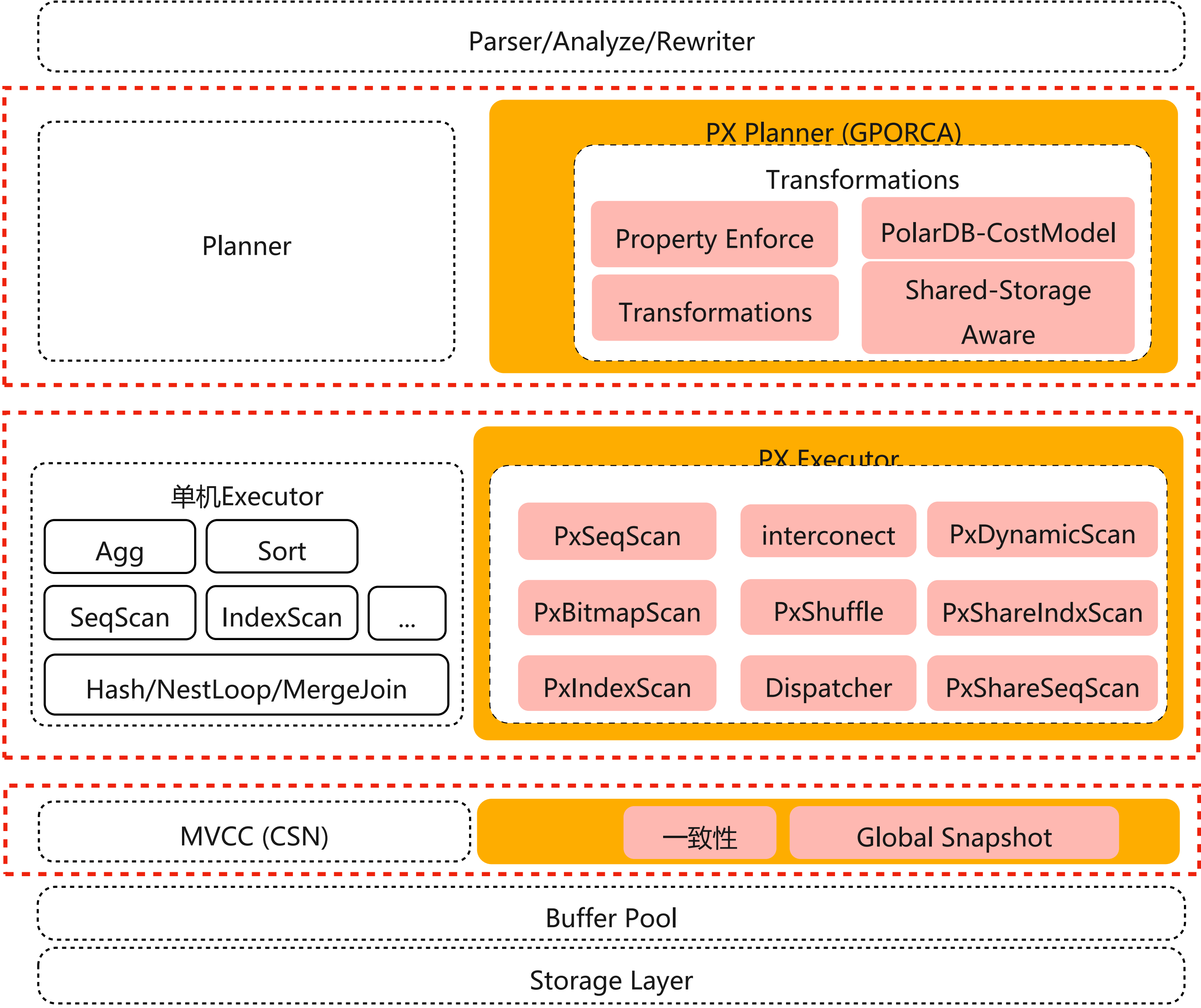
优化器

执行器

事务层

Buffer层

存储层



PolarDB总体架构 – 三节点高可用架构



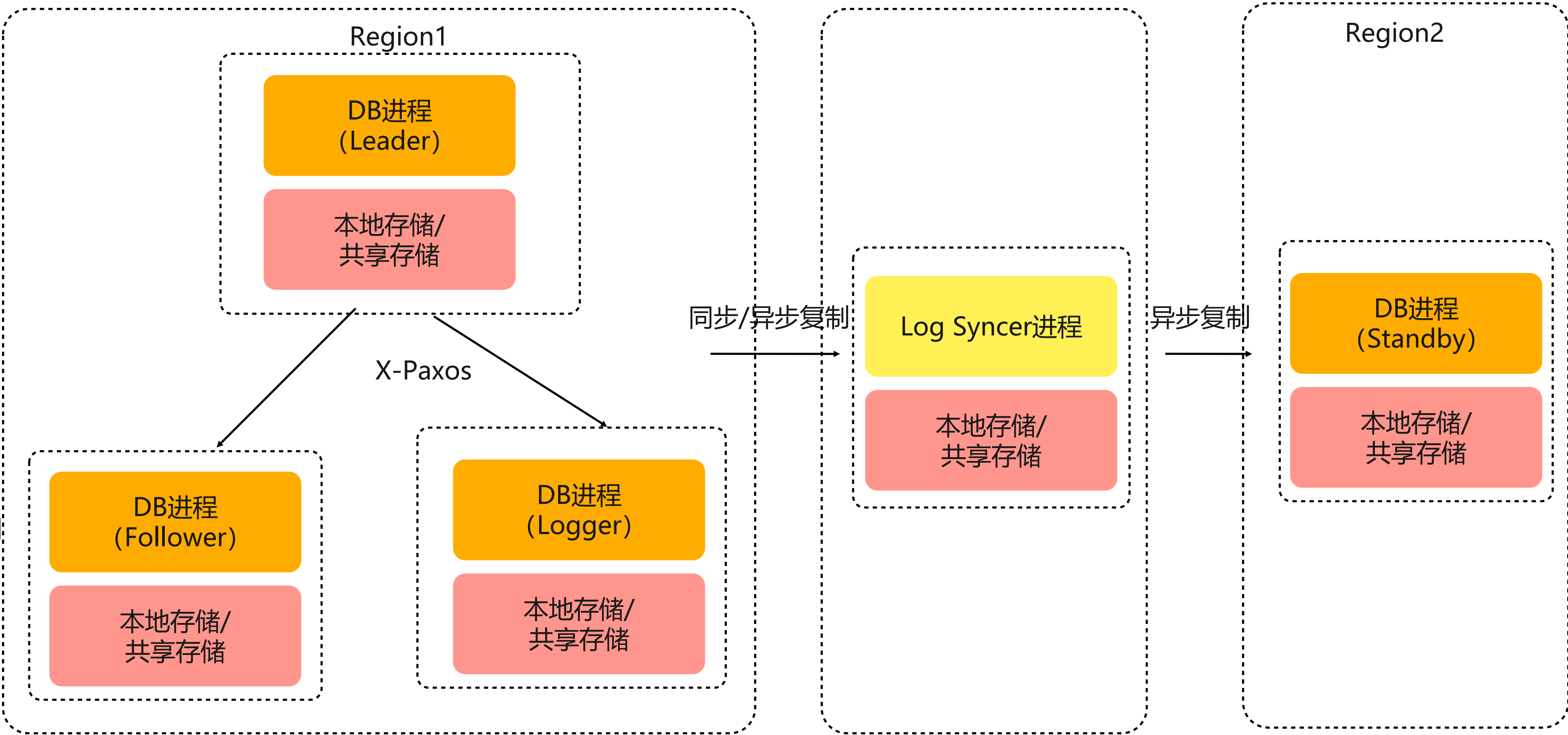
基于X-Paxos的流复制

Region内:

- low RTO, zero RPO
- 自动failover
- 集群变更
- 日志节点

跨Region:

- 异步复制
- 本地存储和共享存储
- 兼容原生流复制和逻辑复制



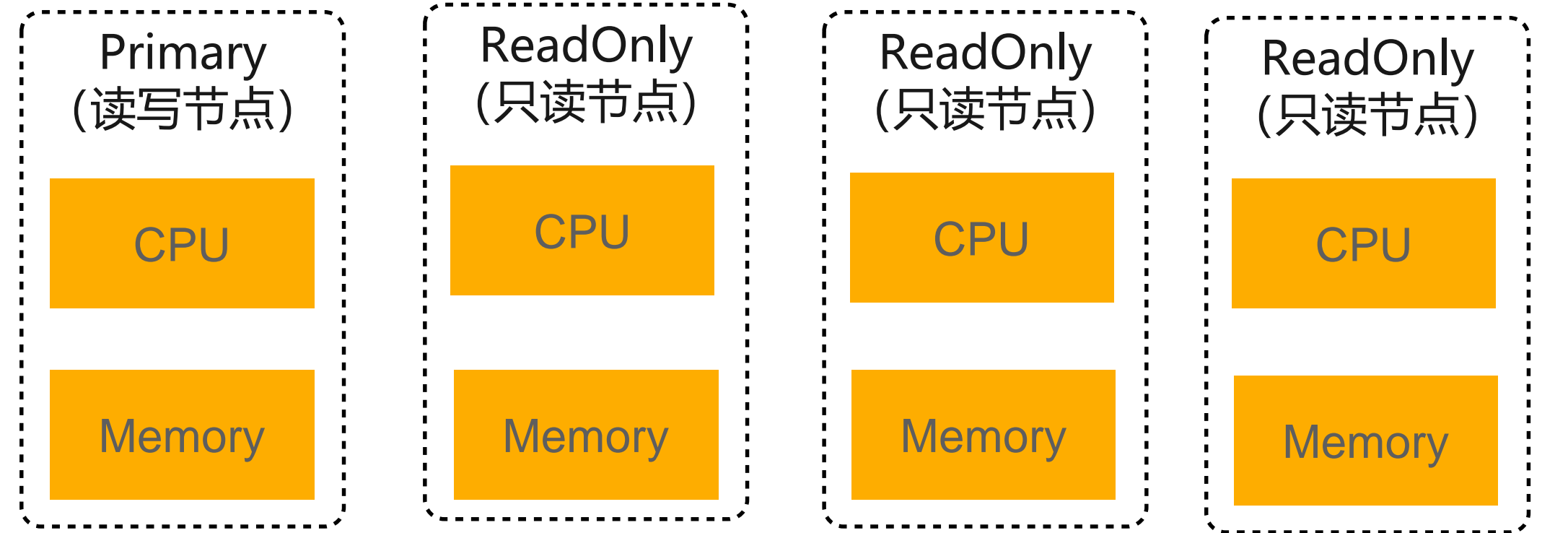
PolarDB总体架构 - 架构之间的关系

3个子架构是正交的关系



单机引擎：
TP型查询

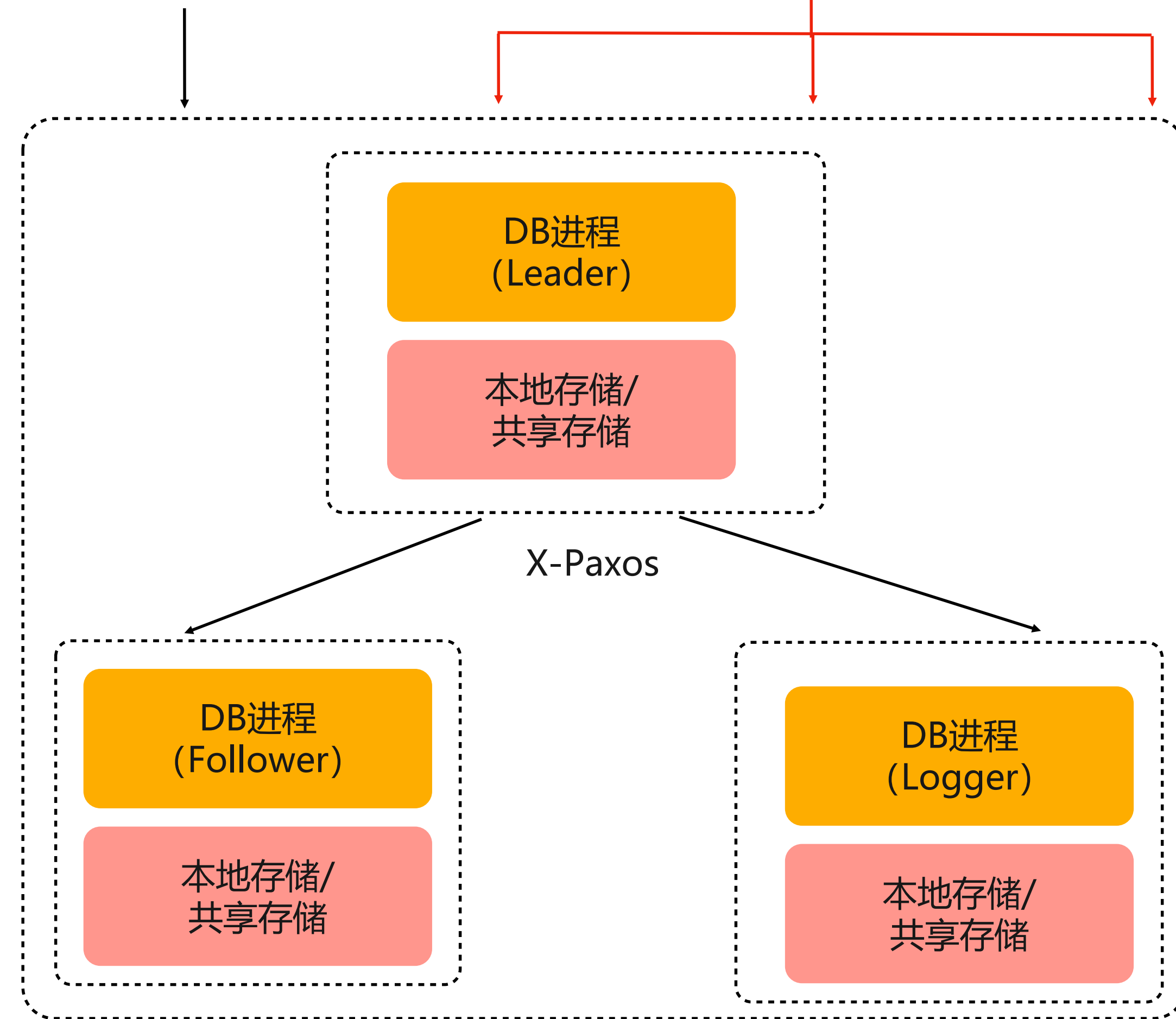
分布式并行计算引擎：
AP型查询



云原生架构+HTAP架构

单机引擎：
TP型查询

分布式并行计算引擎：
AP型查询



X-Paxos架构+HTAP架构

02 PolarDB企业级特性

架构

云原生架构
HTAP架构
X-Paxos高可用架构

高性能

CSN Snapshot
WAL Pipeline
预读/预扩展
RelSizeCache
CLOG优化
FullPageWrite优化

高可用

DataMax
Online Promote
延迟回放
并行回放

常驻 BufferPool
Replication Slot
算子级内存控制

安全

TDE 透明加密

AES 128位
AES 256位
国密 SM4

03

PolarDB开源社区

PolarDB 开源社区 - 开源策略



PolarDB开源策略：

1. 100%兼容PostgreSQL
2. 内部代码全部开源

PolarDB内核

PolarDB分布式文件系统

PolarDB云管控

<div>PolarDB-for-PostgreSQLPublic</div> <div>The default branch of PolarDB switched to “main” on 20210901, which supports compute-storage architecture. The “POLARDB_11_STABLE” is the stable branch which is based on PostgreSQL branch in the past switched to “distributed” branch, which supports distributed architecture.</div> <div>C ☆ 2,112 🍴 308 ⌚ 64 🔗 1 Updated 20 hours ago</div>	<div>PolarDB-Stack-StoragePublic</div> <div>Go ☆ 10 🍴 Apache-2.0 🍴 6 ⌚ 0 🔗 0 Updated on 29 Oct</div>
<div>PolarDB-Stack-OperatorPublic</div> <div>PolarDB Stack is a DBaaS implementation for PolarDB-for-Postgres, as an operator creates and manages PolarDB/PostgreSQL clusters running in Kubernetes. It provides re-construct, failover switch, high-available capabilities for each clusters.</div> <div>Go ☆ 11 🍴 Apache-2.0 🍴 5 ⌚ 0 🔗 0 Updated 6 days ago</div>	<div>PolarDB-Stack-WorkflowPublic</div> <div>Go ☆ 10 🍴 Apache-2.0 🍴 5 ⌚ 0 🔗 0 Updated on 25 Oct</div>
<div>PolarDB-FileSystemPublic</div> <div>C++ ☆ 63 🍴 Apache-2.0 🍴 23 ⌚ 0 🔗 1 Updated 15 days ago</div>	<div>PolarDB-Stack-DaemonPublic</div> <div>PolarStack-Daemon is a daemon process in DBaaS PolarStack. It runs on all hosts and is responsible for status collection, db log clear, db engine images availability collection, host network state collection, the basic host information and status for db cluster creating/migrating/state recognition</div> <div>Go ☆ 10 🍴 Apache-2.0 🍴 3 ⌚ 0 🔗 0 Updated on 25 Oct</div>
<div>PolarDB-NodeAgentPublic</div> <div>db-monitor is a light-weight and flexible agent for data collection, which supports performance monitoring from hosts and instances. It is a plug-in process running on physical machines or virtual machines, collecting performance data every second and real-time logs of all the instances (both containerized in VMs and containerized in...)</div> <div>Go ☆ 7 🍴 Apache-2.0 🍴 4 ⌚ 0 🔗 0 Updated 17 days ago</div>	<div>PolarDB-Stack-CommonPublic</div> <div>Go ☆ 8 🍴 Apache-2.0 🍴 5 ⌚ 0 🔗 0 Updated on 25 Oct</div>

<https://apsaradb.github.io/PolarDB-for-PostgreSQL/zh/>

PolarDB 开源社区 – 开源进展



PolarDB提供丰富的文档和视频资料：

- 整体架构文档
- 核心功能文档
- 快速入门文档
- 每周直播讲解PolarDB内核原理



快速上手

- 介绍
- 实例部署：基于阿里云云服务
- 实例部署：基于单机本地存储
 - 1. 安装 Docker
 - 2. 下载 PolarDB 源代码
 - 3. 环境准备
 - 方式 1：基于 CentOS7 的 Docker 开发镜像
 - 方式 2：基于 CentOS7 操作系统从零开始
 - 4. 编译并搭建实例
 - 本地单节点实例
 - 本地多节点实例
 - 本地多节点带备库实例
 - 本地多节点 HTAP 实例
 - 5. 检查和测试
 - 实例检查
 - 一键执行全量回归测试
- 实例部署：基于 NBD 共享存储
- 实例部署：基于 Ceph 共享存储
- 实例部署：基于 PolarDB Stack 共享存储

Logindex

背景介绍

PolarDB 采用了共享存储—写多读架构，读写节点 RW 和多个只读节点 RO 共享同一份存储，读写节点可以读写共享存储中的数据；只读节点仅能各自通过回放日志，从共享存储中读取数据，而不能写入，只读节点 RO 通过内存同步来维护数据的一致性。此外，只读节点可同时对外提供服务用于实现读写分离与负载均衡，在读写节点异常 crash 时，可将只读节点提升为读写节点，保证集群的高可用。基本架构图如下所示：

PolarDB整体架构简介

下面会从两个方面来解读PolarDB的架构，分别是：存储计算分离架构、HTAP架构。

存储计算分离架构

PolarDB是存储计算分离的设计，存储集群和计算集群可以分别独立

- 1. 当计算能力不够时，可以单独扩展计算集群。
- 2. 当存储容量不够时，可以单独扩展存储集群。

PolarDB for PostgreSQL

阿里云自主研发的云原生数据库产品

[快速上手](#)[架构解读](#)

HTAP 能力

基于 Shared-Storage 的分布式并行执行
加速在 OLTP 场景下的 OLAP 查

仅保存近12个月的直播回放

《PolarDB-for-PostgreS...

2021/12/03 42分34秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

《开源大咖说》-5-Polar...

2021/11/04 33分40秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

详细数据

《PolarDB-for-PostgreS...

2021/11/04 33分31秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

《开源大咖说》-4-分布...

2021/10/14 44分2秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

《开源大咖说》-3-分布...

2021/09/30 63分12秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

《PolarDB for PostgreSQL...

2021/09/24 69分49秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

《开源大咖说》-2-数据...

2021/09/23 55分59秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

《PolarDB for PostgreSQL...

2021/09/17 40分13秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

《开源大咖说》第1期

2021/09/16 61分16秒 北侠(北侠)

[分享](#)[统计](#)[更多](#)

观看回放：
https://developer.aliyun.com/topic/PolarDB_release

钉钉扫描下方二维码，获取更多
新品发布会资料，直播信息。
或搜索钉钉群：30675445



开源PolarDB企业级架构重磅发布