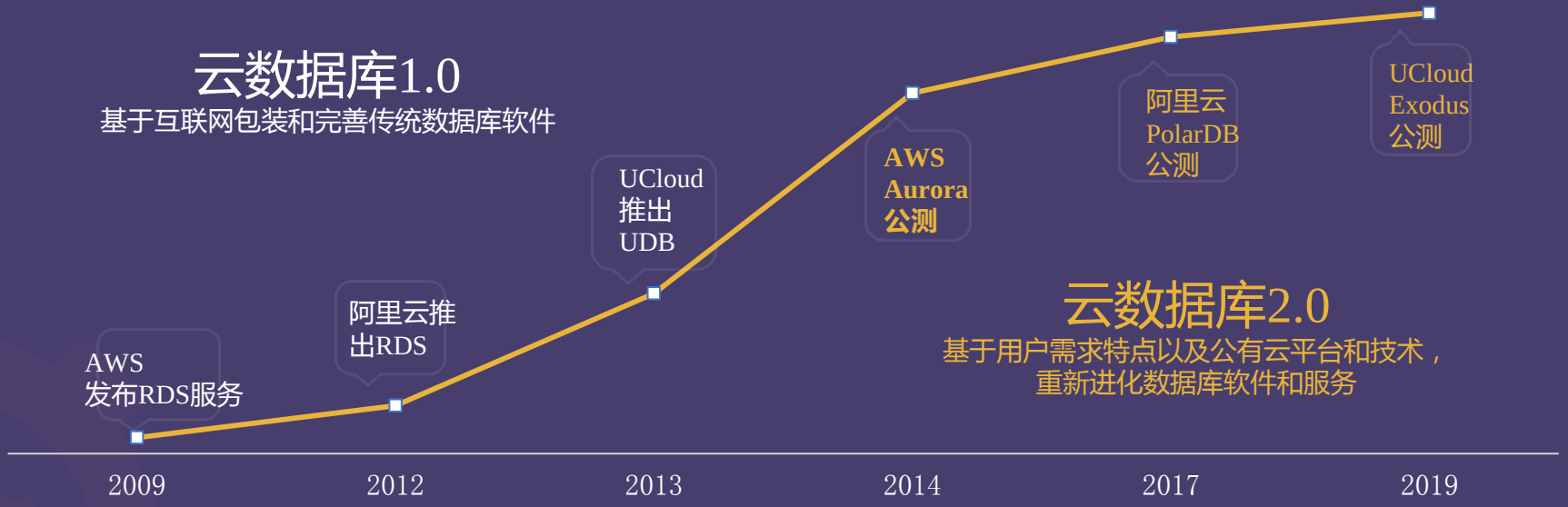
UCLOUD

悟有所值 Ucan 下午茶 Ucan Afternoon Tea

新一代公有云分布式数据库
UCloud Exodus

刘坚君

UCloud资深数据库研发工程师



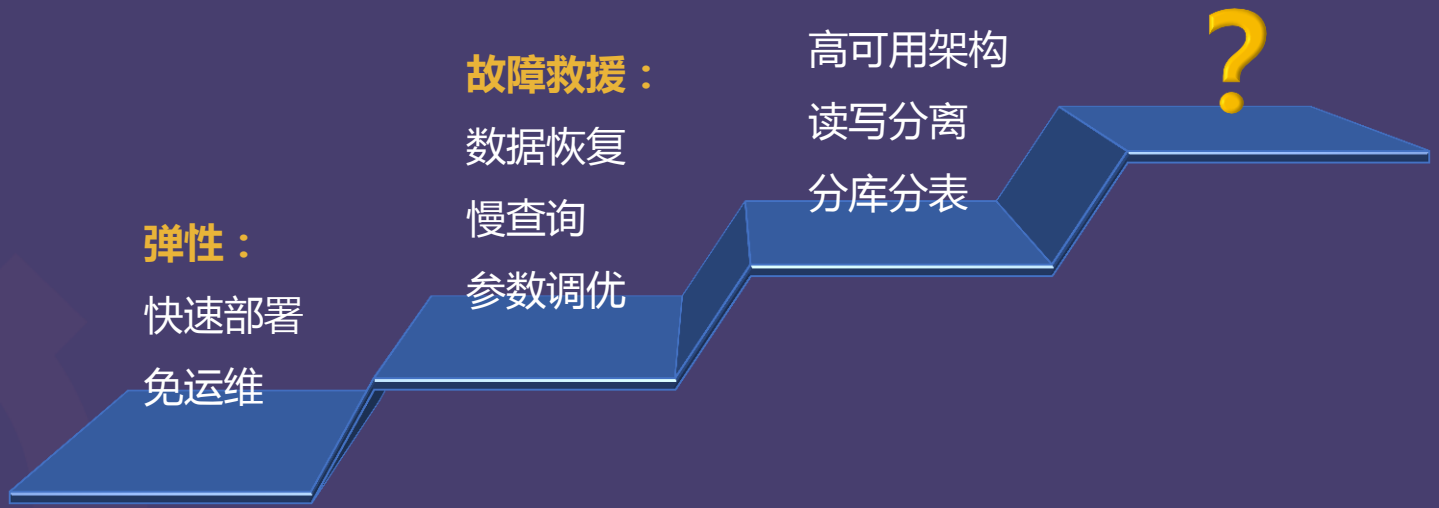
公有云数据库发展-关键时间节点

悟有所值
Ucan Afternoon Tea

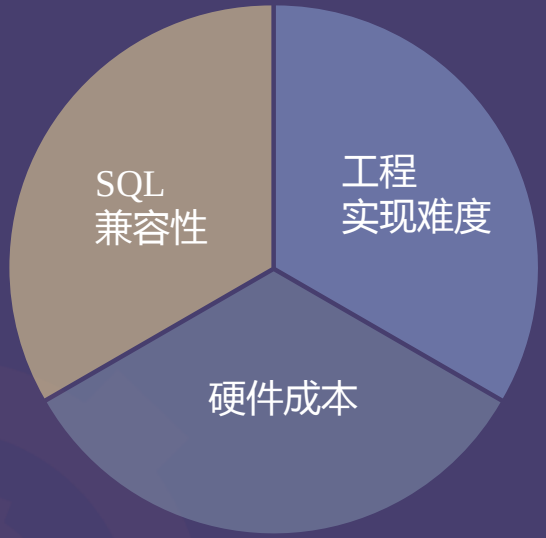
目录

1. 云数据库1.0的三个问题

云数据库1.0的用户价值



问题1：容量和性能

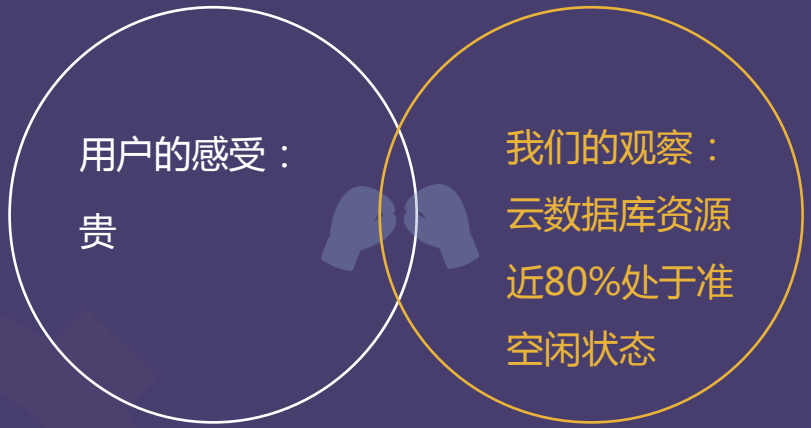


垂直升级

中间件+MySQL

NewSQL(Spanner/F1 OceanBase等)

问题2：用户成本



原因：

1.数据库是高频产品：新项目的开发、测试、上线、老项目的扩容

2.云数据库并非按需使用和计费

3.刚上线的新业务，峰值无法预估

4.衰退期的老项目无法缩容

问题3：运营成本

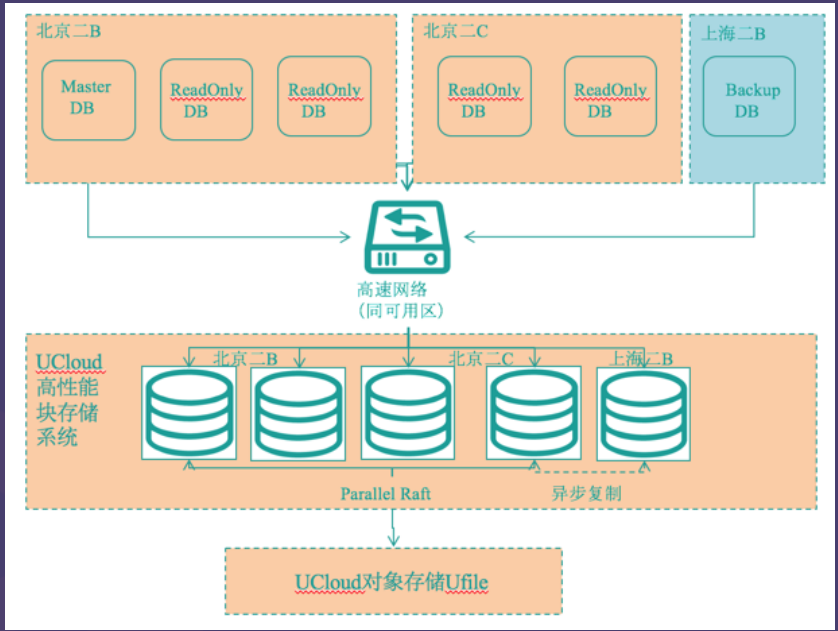
硬件成本：
机型选型
/TCO

人力成本：
分库分表&业务改造/支持
故障救援

运营风险：
大数量备份/
容灾失败

2. UCloud Exodus 的解决之道

UCloud Exodus 架构图



1. 计算和存储分离。计算层只做SQL解析、事务处理、数据修改等计算操作，真正的存储下放到了底层的分布式存储（UCloud UDisk）

2. 计算层和存储层分别扩展

3. 数据实时更新到对象存储

4. **设计指标：**十万级写QPS、百万级读QPS、100TB容量，MySQL 100%兼容

云数据库1.0三大问题迎刃而解

悟有所值 Ucan Afternoon Tea

容量和性能

1.容量达100TB，写入延迟接近于本地SSD，单点写入达到10W QPS

2.针对大部分业务数据库读多写少的特点，增加只读实例实现读性能大幅提升

3.MySQL 100%兼容

4.低硬件成本

租用成本

1.存储容量和计算能力按需扩容， 根据业务需求按需付费

2.多个业务的开发、测试环境可共用一套Exodus实例

3.新业务按量逐步扩容，老业务按使用量逐步缩容

运营成本

资源成本：计算机型和存储机型分开选型，成本大幅降低

人力成本：MySQL 100%兼容，无需分库分表、业务改造

运营风险：数据实时备份、存储三副本冗余、计算节点秒级迁移

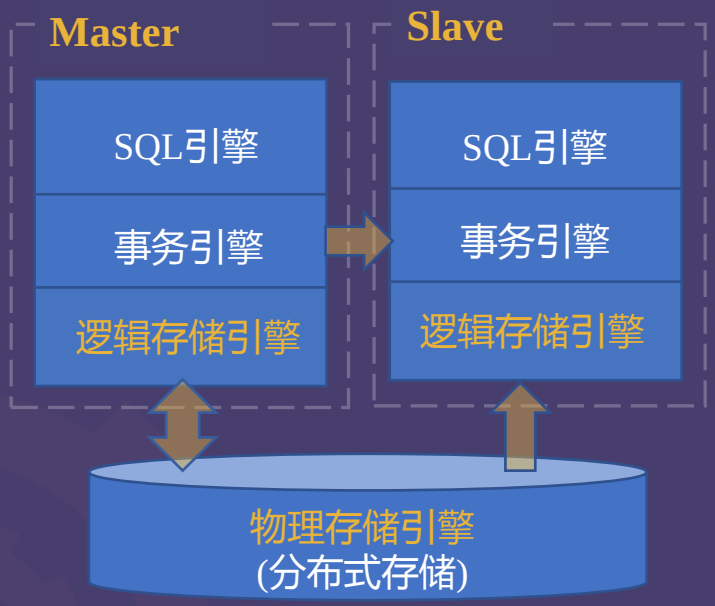
更大的难题

UCloud Exodus至少和业内发布的三款云数据库2.0产品，在架构上是相似的。

而作为一家中立无巨头背书的云厂商，如果只是做一个产品和服务能力跟大厂差不多的产品，在市场上没有太大机会

那么，UCloud Exodus和竞品相比，还能够有什么区别（**更大的价值点**）？

计算和存储分离架构



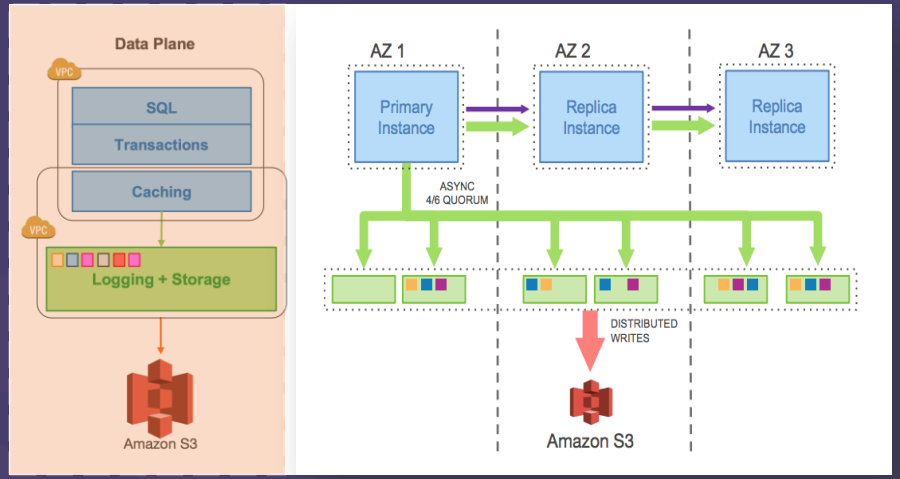
计算和存储分离，其思路是把数据库的计算层（SQL+事务）和存储层（数据持久化）解耦，打开，各自独立扩展。

要实现高性能，需解决两个问题：

1. **IO路径的问题：**IO由本地IO变成网络IO，延迟增加吞吐量降低，IO如何优化

2. **主从数据同步的问题：**读性能的增加依赖于从节点的数量，主从数据如何高效同步，将是平衡可用性和数据一致性的重要问题

AWS Aurora：Shared-Storage



IO路径的问题：

1.数据写入时（计算层到存储层），只写Redo Log不写page，存储节点根据 Redo Log重演出 page；同时避免Binlog、DoubleWrite 写入，最终单事务平均IO为单机 MySQL 的 1/7.7

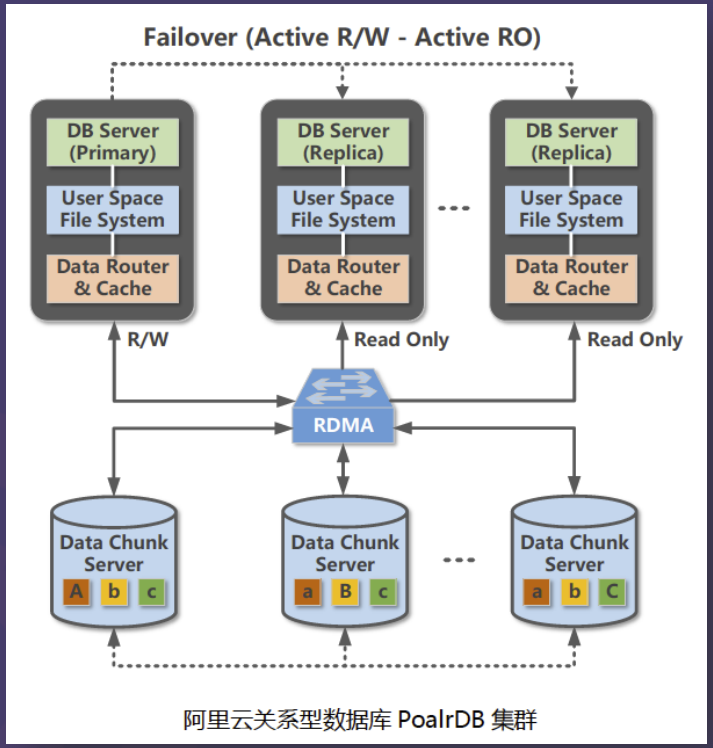
主从同步的问题：

将 Redo Log 写入到存储层的同时， 同步 Redo Log到从 节点， 从节点根据 Redo Log更新缓存 page，实现数据最终一致

Shared-storage:

存储层负责page重演、多副本一致性保证、故障恢复、备份还原等操作，包含大量数据库业务逻辑，是一个专门为数据库定制的分布式存储引擎

Aliyun PolarDB：Shared-Disk



IO路径问题：

采用最新软硬件技术（RDMA/NVMe/SPDK），对数据库 IO 路径做极致优化

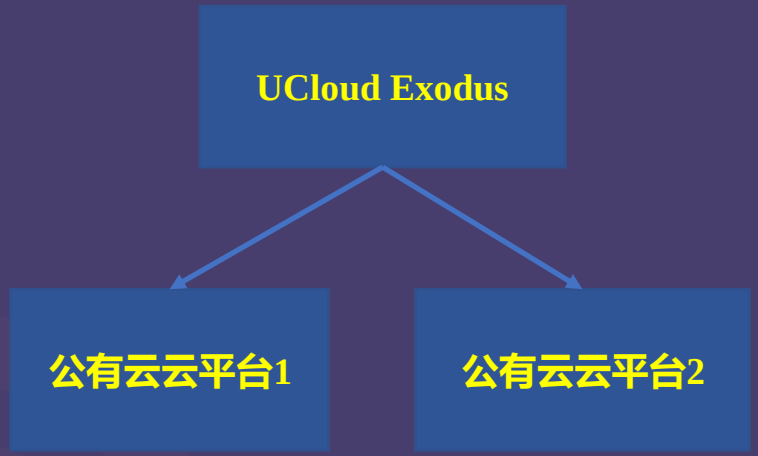
主从同步问题：

事务提交时，同步 Redolog 地址到从节点，从节点从 PloarStore 加载 redolog 并刷新page缓存

Shared-Disk：

从目前公开的论文的资料看，底层存储 PolarStore 并不针对数据库的特殊逻辑，其作用本质上为一个分布式Disk，故将其归类为 Shared-Disk 范畴

UCloud Exodus: Shared-ALL-Disk



IO路径的问题：更进一步

1.充分相信未来公有云的底层分布式存储，能提供低延迟（微秒级别）、高IOPS的产品。

2.以此为出发点，改造传统数据库内核，打造适配大部分公有云平台的，大容量高性能数据库。

主从同步的问题：

改造传统数据库内核， 实现高性能的主从同步，增强Exodus的读处理能力。

进一步后的产品形态

功能完备

通用

开源

进一步后的巨大价值

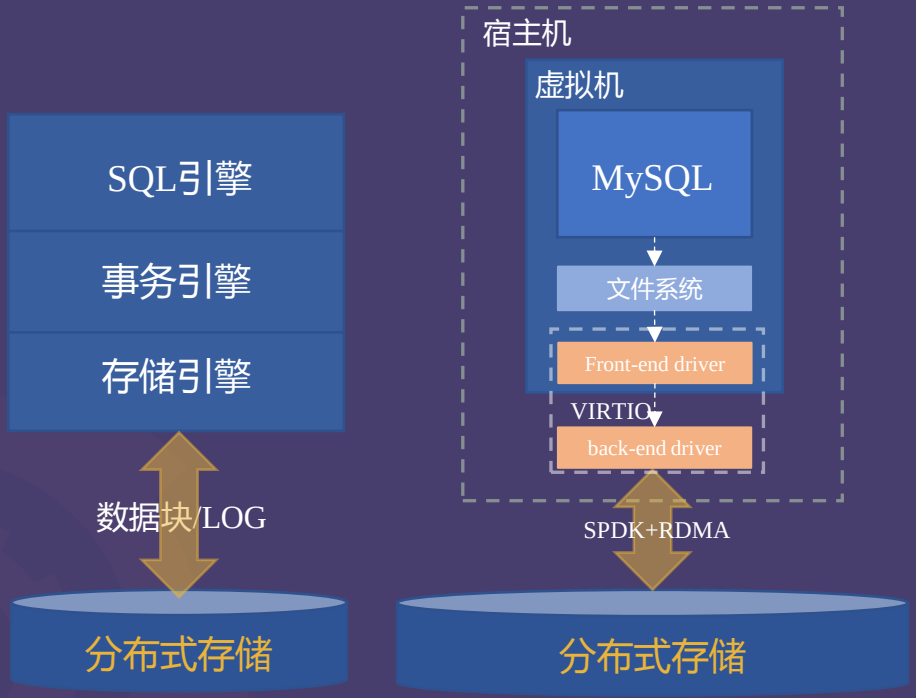
悟有所值 Ucan Afternoon Tea

和行业做朋友

和用户做朋友

和时间做朋友

技术方案Part1: IO路径问题

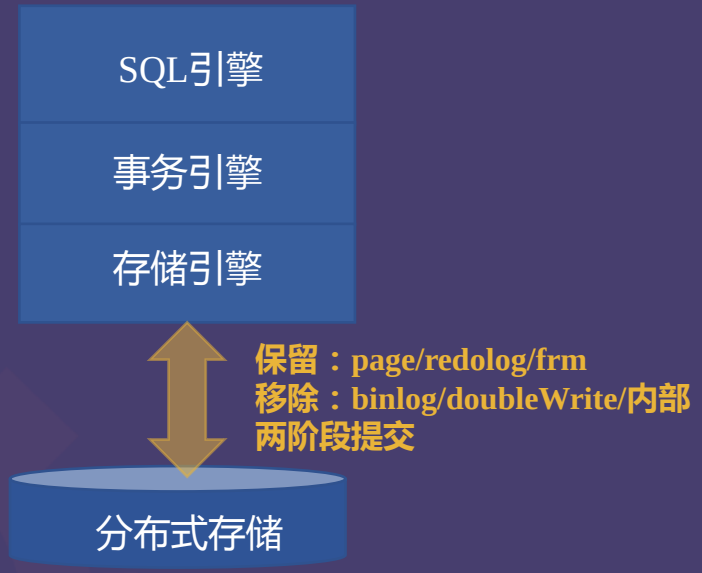


方案：基于MySQL+InnoDB，直接复用公有云分布式存储产品（如UCloud 块存储产品 UDisk ）

我们判断，未来云平台的底层的分布式存储产品，在Io路径上将实现极致优化，主流云平台底层分布式存储将实现微秒级延迟，百万级IOPS，足以支持高性能业务（如数据库）

以 UDisk 为例，我们通过 虚拟机 -> VIRTIO->宿主机 ->SPDK+RDMA ->chunk server的技术方案，已经测得100us的IO延迟，100W+ IOPS

技术方案Part2: IO路径问题

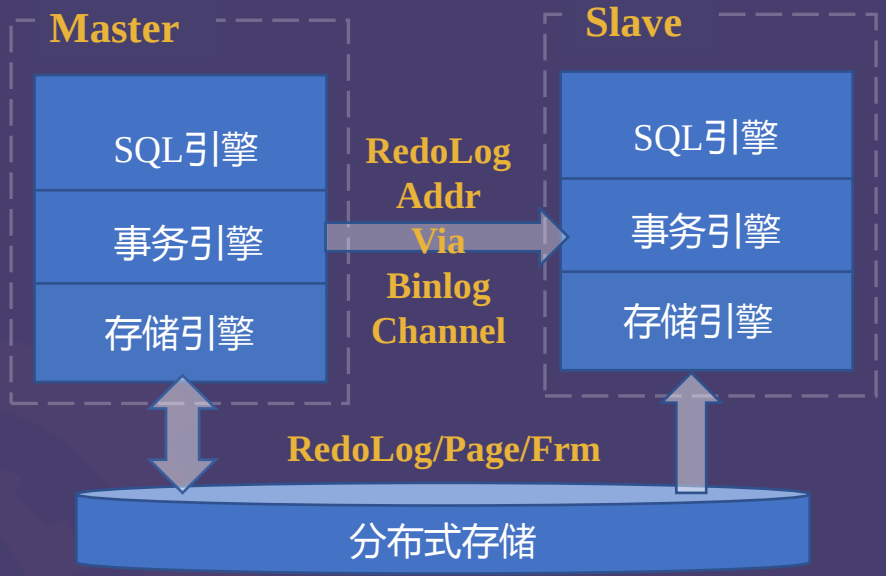


把难题交给队友，押宝公有云产品的生长能力，不意味着我们在Io路径优化上毫无作为。

具体做法：

1. 移除Binlog， 主从同步采用redolog复制，下游系统数据同步根据归档日志（由redolog归档而来）反向生成binlog然后同步；
2. Binlog移除后，事务提交时内部两阶段提交亦可省去
3. 基于分布式存储的原子写能力， 有效去DoubleWrite

技术方案Part3: 主从同步问题



1. **借鉴：**和大部分2.0数据库一样，主从数据同步采用Redolog。主从之间同步Redolog Addr，从收到Addr之后再去底层存储拉取

2. **创新：**利用MySQL原有的机制，设计了一种简单可靠的Redolog Addr同步机制

UCLOUD

悟有所值 Ucan 下午茶 Ucan Afternoon Tea

关注“UCloud技术公告牌”，更多分享与交流