OceanBase 1.0

——分布式技术架构

署名：杨传辉（日照）
职称：资深专家

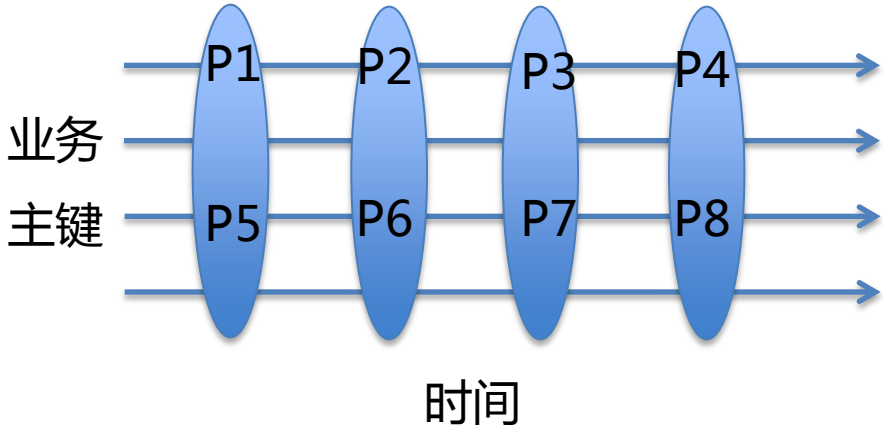
主办单位：杭州 Alibaba Group 阿里巴巴集团 战略合作伙伴：intel

目 录
content

一、OceanBase数据存储架构
二、OceanBase分布式查询
三、经验&思考

一、OceanBase数据存储架构（可扩展、高可用、低成本）

数据的分布式



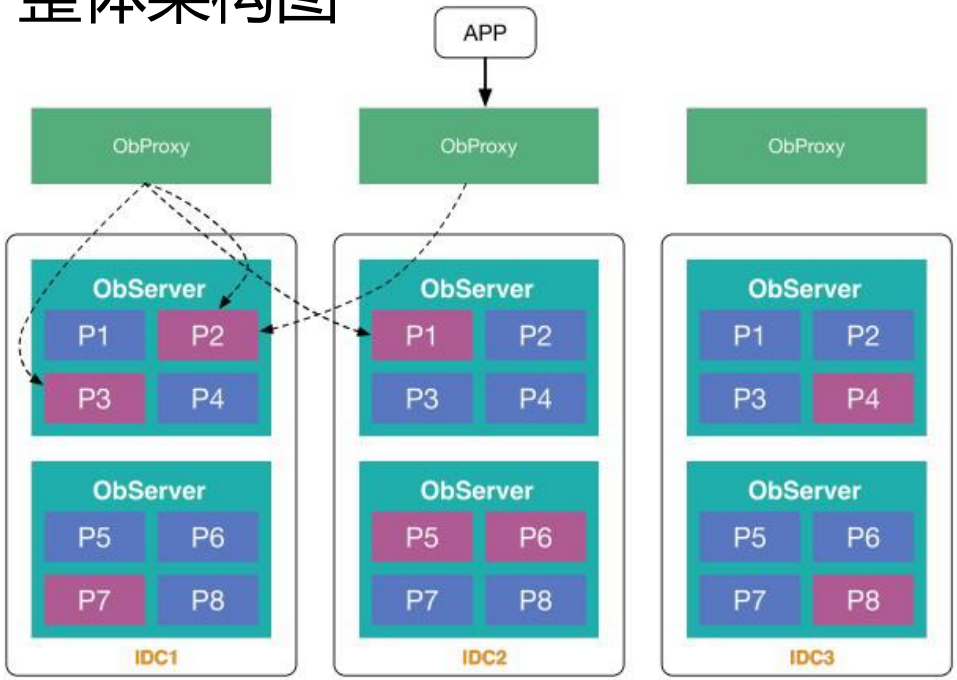
两级分区表：时间 + 业务主键

融合在线数据(OLTP)&历史数据(OLAP)

历史数据：更高压缩率，预计算

在线数据：内存计算

整体架构图



P1 ~ P8：多副本数据分片，跨机房

ObProxy：拥有百万级处理能力的代理服务

ObServer：OceanBase工作机，包含分区服务模块及总控服务模块

RootService：集成在ObServer，由系统自动选举一台机器提供总控服务

事务处理

1

单分区事务

MVCC：写不阻塞读
优化：日志同步异步化，内存结构优化

2

多分区事务

不跨ObServer：融合为单分区，执行单机事务
跨ObServer：两阶段提交（Two-Phase Commit）

3

分布式事务优化

Table Group：将同时操作的表格尽量调度到一台服务器
无协调者日志：所有参与者Prepare成功，事务即可返回

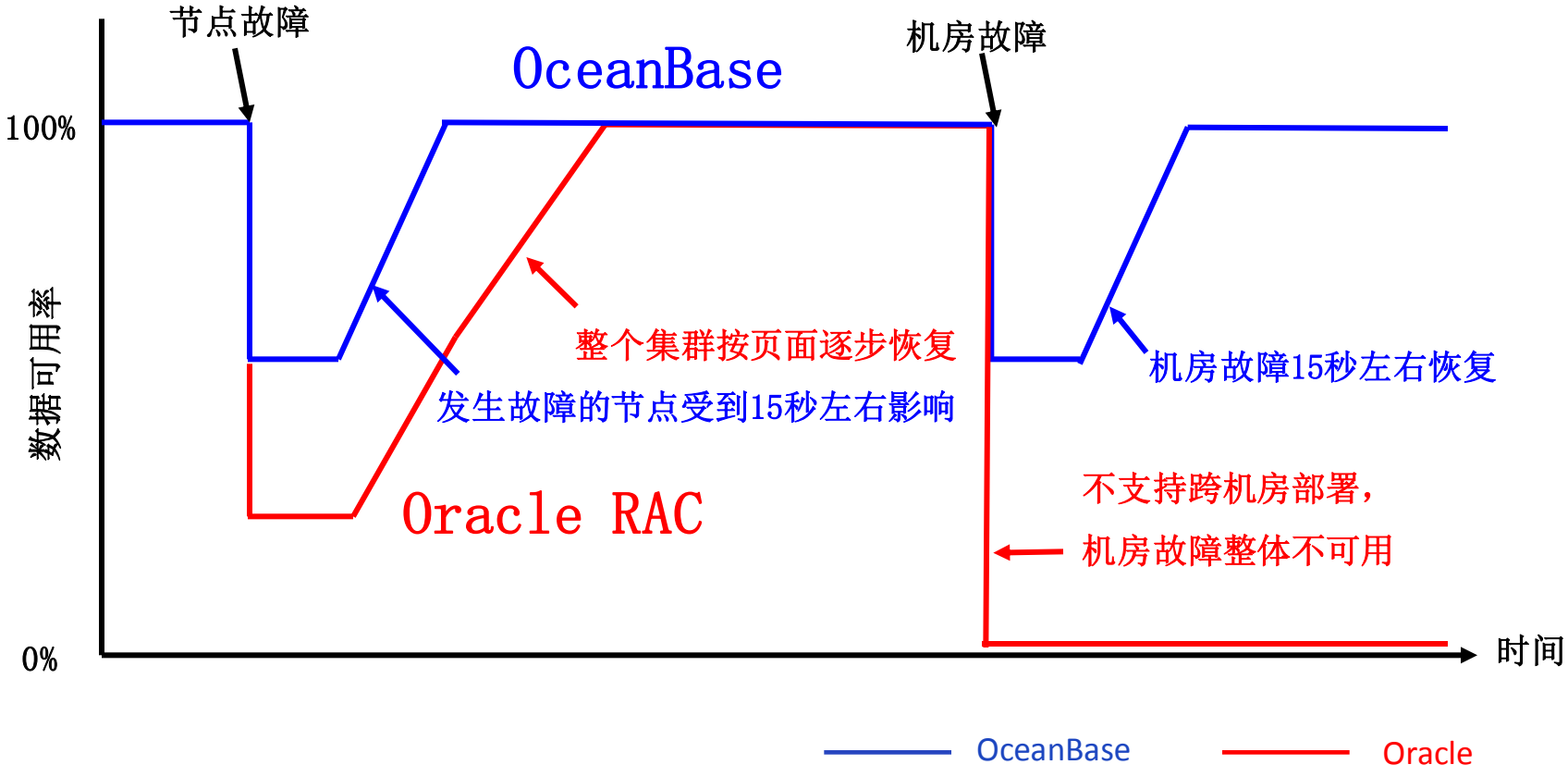
高可用原理

自动工作负载均衡

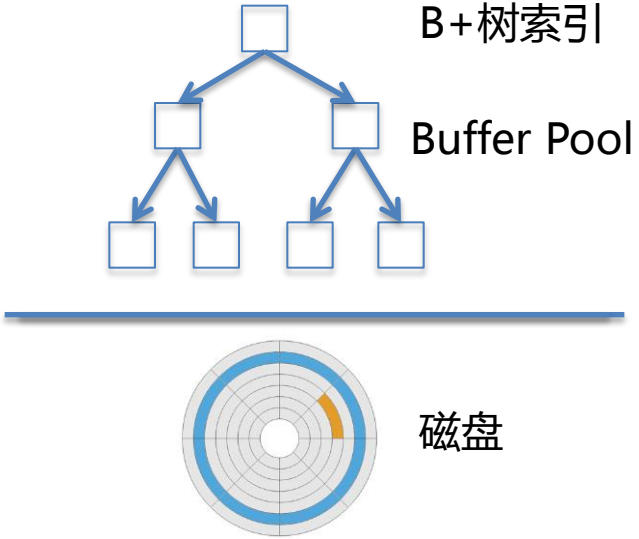
运行在Linux系统上的
ObServer

采用Paxos协议实现的
分布式选举和多库多活技术

分布式调度与管控

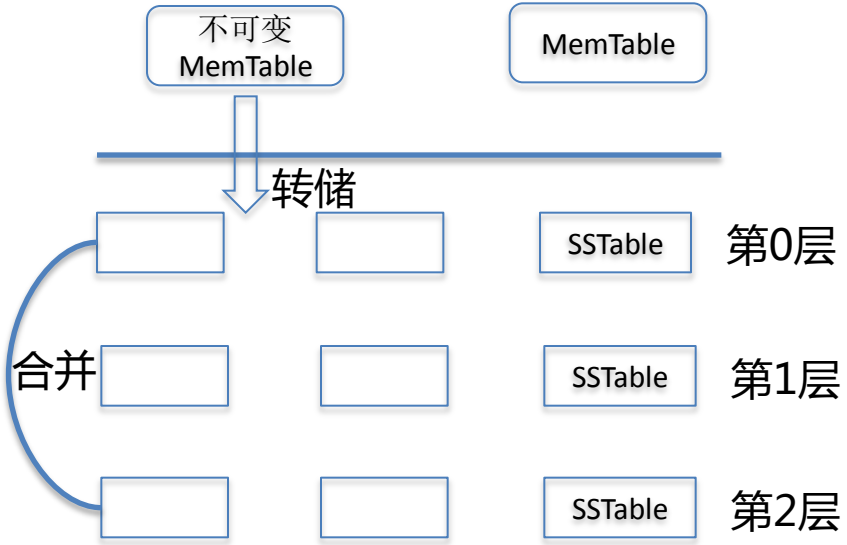

节点故障
机房故障
OceanBase
100%
数据可用率
整个集群按页面逐步恢复
发生故障的节点受到15秒左右影响
机房故障15秒左右恢复
Oracle RAC
不支持跨机房部署，
机房故障整体不可用
0%
时间
OceanBase
Oracle

数据库引擎



面向磁盘设计，B+树索引

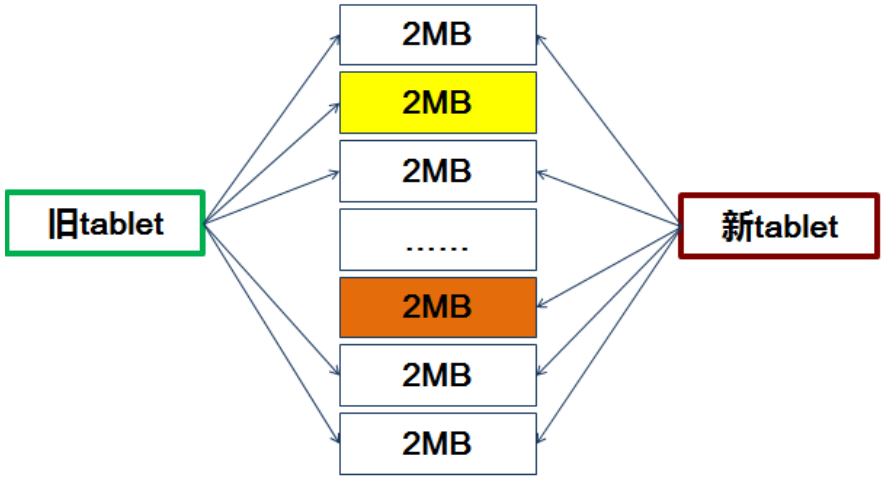
写入放大效应

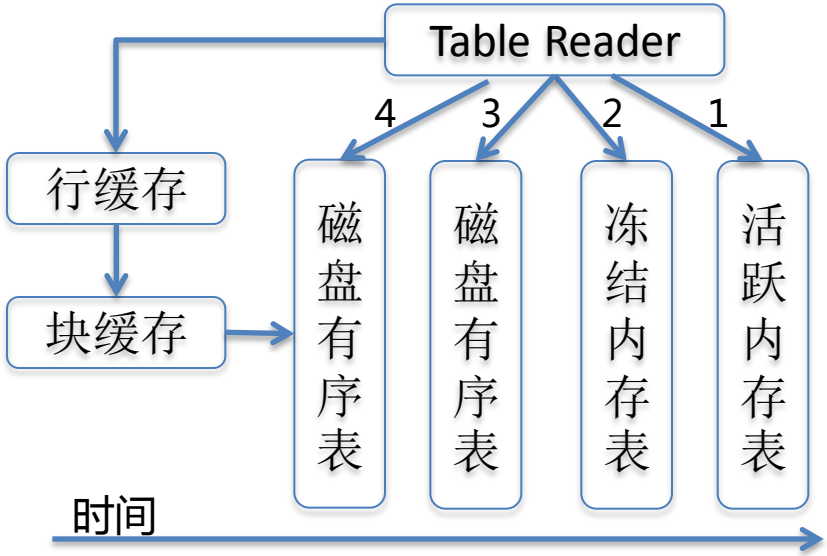


面向写优化

合并（compaction）代价很大

OceanBase存储引擎





数据按照2MB划分宏块

合并优化：只读写修改过的宏块

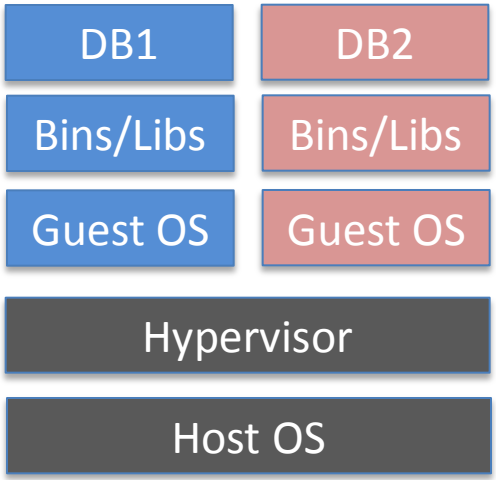
错峰合并：错开副本的合并时机

按时间线逆序读取，行内列索引过滤

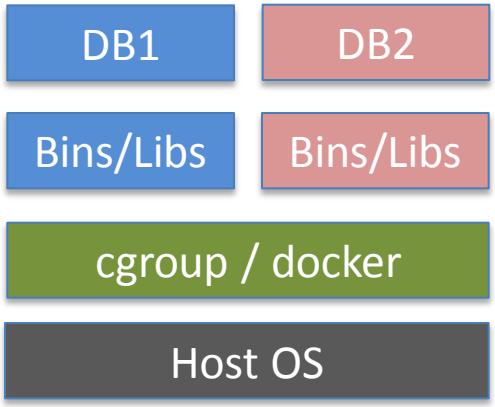
行缓存：行数据&布隆过滤器

单行读写：接近内存数据库

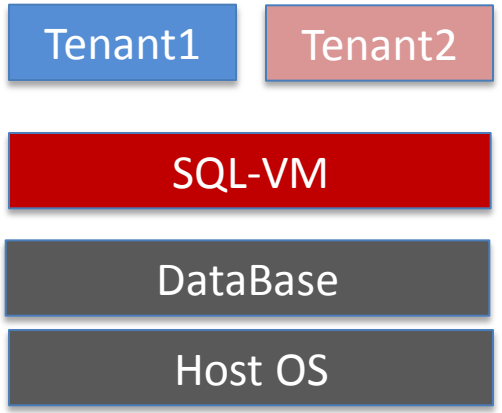
SQL-VM



虚拟机：重量级OS隔离



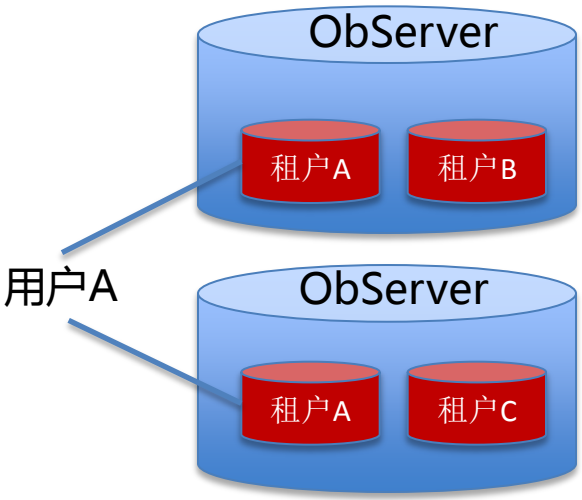
Cgroup：轻量级进程隔离



类似Oracle PDB/CDB

统一管理，降低成本

多租户原理



两级负载均衡

租户间均衡：租户均匀分散到ObServer

租户内均衡：同一个租户在多个ObServer尽可能均衡

小租户尽可能在一台ObServer

SQL-VM资源隔离

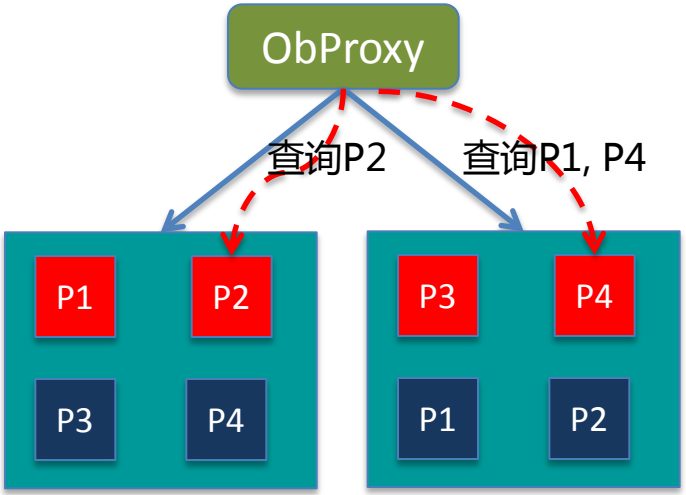
CPU隔离：基于时间片的主动调度器

IO隔离：基于deadline和优先级的调度算法

内存隔离：内存限制 + 公平挤占算法

二、OceanBase分布式查询

ObProxy



连接管理：客户端连接 ⇔ 服务端连接

SQL解析：轻量级Parser，长短边

分区位置缓存：访问失效时刷新

高性能异步框架，线程本地化

运维：热升级，全链路监控

分布式查询

1

本地作业

ObProxy路由到分区所在ObServer
单机查询

2

远程作业

ObProxy路由错误，且只读取单个分区
调度器把整个作业发送给远程机器执行

3

分布式作业

读取的数据位于多台ObServer
并行查询：任务拆分，结果合并，并发数限制等

三、经验&思考

1

高可用

强一致性+高可用 是未来云数据库的标配
云数据库的高可用必选Paxos

2

自动化

强一致是自动化的前提
尽可能减少人工干预：配置，SQL Hint, etc

3

成本

性能 !=成本
成本需综合考虑性能、压缩比、利用率、运维等各个因素
基线 + 增量存储引擎是个好东西！

2016 The Computing Conference

THANKS