# 概述

# 架构

# 存储引擎

## 内存管理

## 内存数据落盘技术

# SQL引擎

## SQL请求执行流程

## DML语言处理

## DDL语言处理

## 查询改写

## 执行计划

## 执行计划缓存

# SQL调优

## 调优方法

## 分区

## 索引

## 局部索引与全局索引

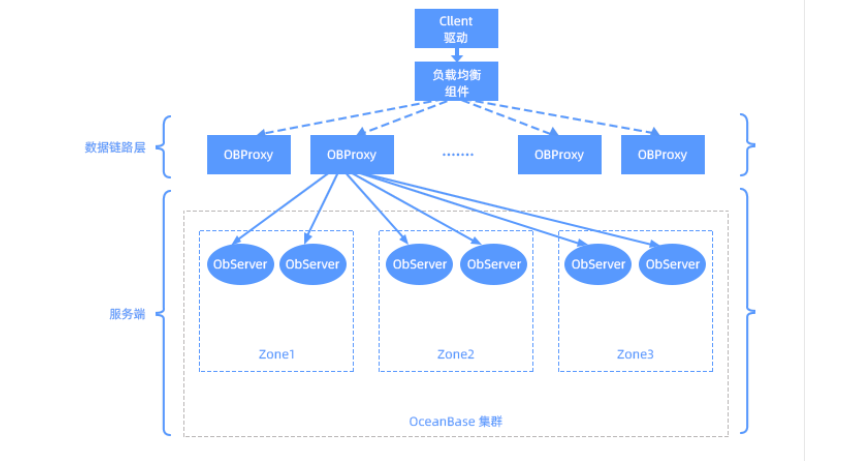
## Hint

## SQL执行性能监控

# OBProxy路由

## 背景

OceanBase在部署完成后，应用可以采用OB提供的客户端、MySQL客户端或者其他语言的客户端来访问OceanBase，OceanBase以服务的形式提供给应用访问。OceanBase集群一般包含多个Zone，每个Zone中又包含一台或多台OBServer，集群中的每个OBServer都可以接收用户的连接并处理请求。  
 • 实现一个OBProxy来接受来自应用的请求，并转发给OBServer，然后OBServer将数据返回给OBProxy，OBProxy将数据转发给应用客户端。



OBProxy作为OceanBase的高性能且易于运维的反向代理服务器，具有防连  
接闪断、OBServer宕机或升级不影响客户端正常请求、兼容所有MySQL客户端、支持热升级和多集群功能。

## 功能

### 路由

#### SQLParser

SQLParser：轻量的sql解析，判断出客户端的sql请求所涉及的表的主副本在哪台机器上，将请求路由至主副本所在的机器上。

#### LDC路由

LDC路由：主要对于读写分离的场景，根据observer和obproxy配置的region  
（区域）和LDC（逻辑机房），将请求发送给本地的副本（后面proxy路由策略  
会详细说明）。

#### 读写分离部署

读写分离部署：对于读写分离的场景，OBProxy会把请求优先发送到本地的只读副本。

#### 黑名单

黑名单：OBProxy在路由过程中，如果发现OBServer不可用，则把该server加入到黑名单。

### 连接管理

在observer宕机/升级/重启时，客户端与OBProxy的连接不会断开， OBProxy可以迅速切换到正常的server上，对应用透明。

• OBProxy支持用户通过同一个OBProxy访问多个OceanBase集群  
 • Server session对于每个client session独占  
 • 同一个client session对应server session状态保持相同(session变量同步)

OBProxy与OB集群（OBServer）保持长连接，客户端一般通过连接池的方式连接到OBProxy。

## 部署

## 策略

根据不同的配置，Obproxy进行综合的路由排序：  
 1、LDC配置：  
 1）本地：同城同机房（IDC相同）  
 2）同城：同城不同机房（IDC不同，Region相同）  
 3）异地：不同的地域（Region不同）  
 2、Observer 状态：常态vs正在合并  
 3、租户的Zone类型：读写型vs只读型  
 4、路由精准度：优先精准度高的  
 OBproxy中有目标partition的路由信息（PS）  
 OBproxy中没有Partition的路由信息，只有租户的路由信息（TS）

写请求：  
 - 写请求路由到ReadWrite zone的主副本  
 读请求：  
 - 强一致性读  
 - 弱一致性读  
 ⚫ 主备均衡路由策略（默认）  
 ⚫ 备优先读策略  
 ⚫ 读写分离策略

## 执行流程

## 使用限制

## 使用和运维

## 常见问题

# 分布式事务

## 全局快照及分布式一致性读

## 分布式两阶段提交

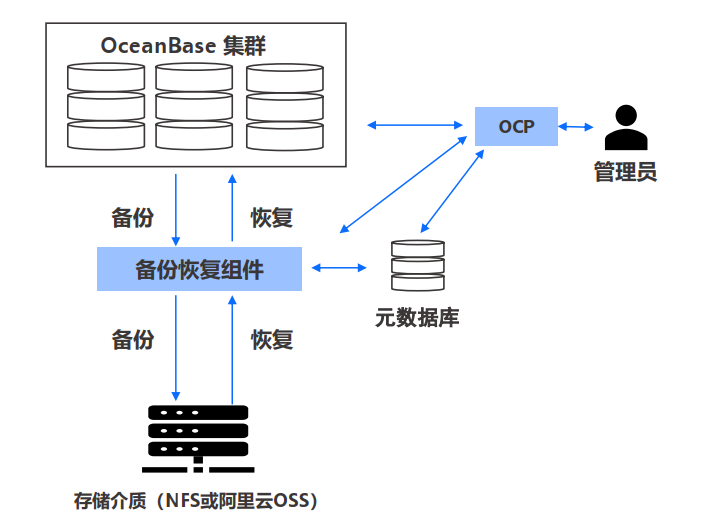
# 数据分布

# 复制/一致性

# 备份恢复

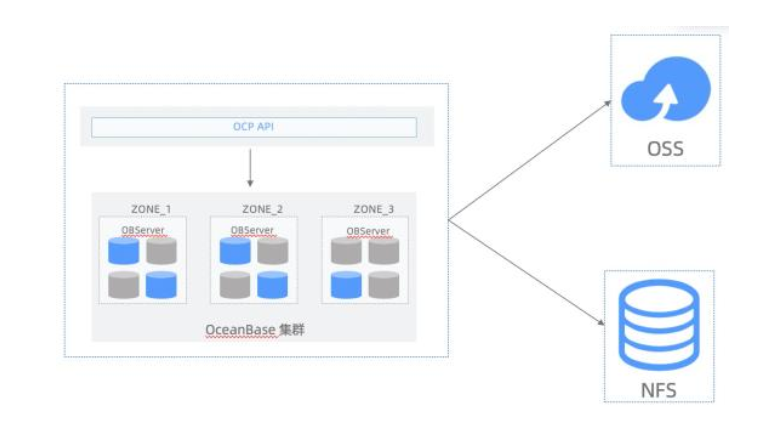
## 架构

### 逻辑备份/恢复方案



OceanBase支持全量备份和增量备份，全量备份是对存储层的基线数据进行备份，增量备份是通过redo-log备份， OceanBase支持在线实时的全量和增量备份，对业务无感知。  
 备份支持2种介质，一种是阿里云OSS存储，另外一种是普通的NFS。 NFS需要一个公共目录，每个OB Server都可以访问， NFS服务器为每台OB Server创建子目录。  
 备份性能可以达到网卡的上限，约1G左右。恢复性能，一般也可以达到500兆左右。  
 备份恢复最小粒度是租户，让备份和恢复更加灵活。  
 备份恢复数据方面，支持逻辑数据（比如用户权限、 表定义、系统变量、用户信息、视图信息等）和物理数据。  
 OceanBase的备份恢复，完全能够满足企业的日常需求。

### 物理备份/恢复方案



## 步骤

## 查看任务状态

## 常见问题

# 容错/故障切换

# 兼容性

# 扩展性

# 高并发

# 高可用

# 数据安全

# 数据压缩

# 数据迁移

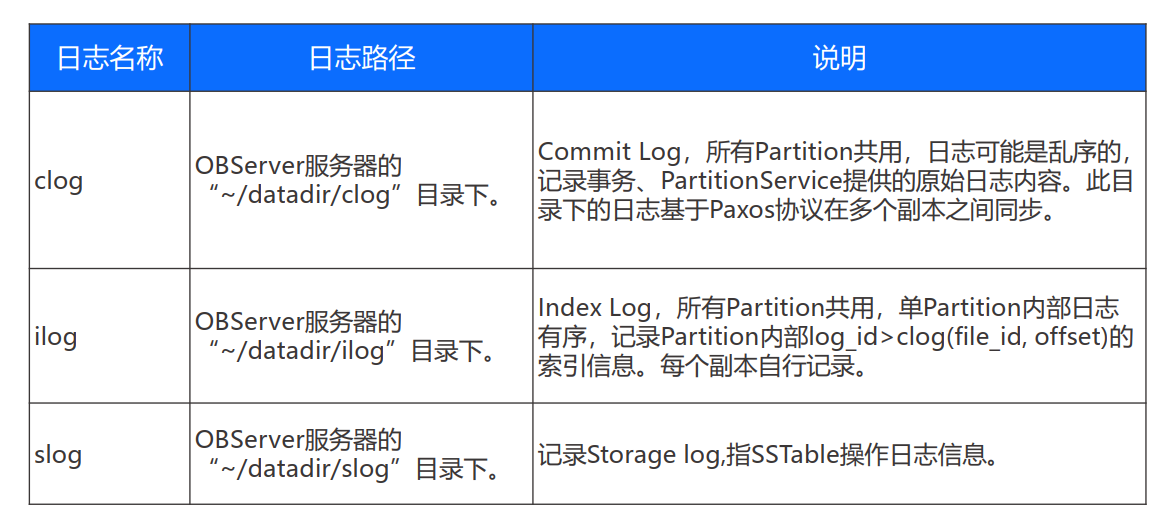
# 运维/监控告警

## 用户权限管理

数据库用户管理操作包括新建用户、删除用户、修改密码、修改用户名、锁定用户、用户授权和撤销授权等。  
 用户分为两类：系统租户下的用户，一般租户下的用户.  
 创建用户时，如果当前会话的租户为系统租户，则新建的用户为系统租户用户，反之为一般租户下的用户。  
 • 创建用户：create user ‘my\_user’identified by ‘my\_password’  
 • 删除用户：drop user ‘my\_user’  
 • 给用户相应权限：grant [用户权限] on [资源对象] to username  
 • 收回权限：revoke [用户权限] on [资源对象] to username  
 • 查看用户：show grants for username

## 日志查询

Observer日志： OceanBase在运行过程中会自动生成日志。维护工程师通过查看和分析日志，可以了解OceanBase的启动和运行状态  
 • /home/admin/oceanbase/log  
 ➢ 事务/存储日志： OceanBase在事务执行过程中，会持久化事务/存储日志  
 • /data/log1/集群名/



### clog

clog是指广义的 Commit Log，代表整个事务的所有日志信息。

事务的日志包括： redo log, prepare log, commit log, abort log, clear log等  
 • redo log记录了事务的具体操作，比如某行数据的某个字段从A修改为B  
 • prepare log记录了事务的prepare状态  
 • commit log表示这个事务成功commit，并记录commit信息，比如事务的全局版本号  
 • clear log用于通知事务清理事务上下文  
 • abort log表示这个事务被回滚所有的事务日志信息，不用于用户查看和定位系统问题

### ilog

### slog

## 日常运维操作

## 数据库监控

# 故障排查

## 常见异常处理

## 灾难恢复