

Mycat 现状对比分析

文档信息

文档编写人	姬文刚	编写日期	2018-03-14
文档评审人		评审日期	

版本修订历史记录

版本号	作者	参与者	起止日期	备注
V1.0				

内容修订历史记录

章节	修改内容	修订人	修订日期	修订原因

一、性能方面

mycat 主机环境配置采用目前x86中间配置，1G net(QPS直接影响因素为主机性能、网络等硬件设施，这里取中值)

TPS/QPS 决定因素大概有这么几个方面

1、mycatserver的集群数量 奇数3\5 集群越大qps越高，但消耗的是硬件成本。

2、mycatserver单点线程数(32\64\128\256) 越大qps越高，但同时带来线程池成本，框架范围不轻动。

3、数据库集群的集群分布情况 一主多从，多主多从，集群分布多qps越高。但是根据读写分离频率，会影响整体主从性能，框架范围不轻动。

4、节点数量增加越多，qps越高，节点带来系统复杂性，框架范围不轻动。

5、物理单库(每个节点) 设置有效索引及有效的全局序列会提升qps。 可优化。

针对以上情况，结合一些验证数据，

大概结论：1亿数据 qps大概在13000 左右。

二、核心功能方面

1、基本功能不列举

三、竞品对比(与 corba 与 rds)

劣势：

1、安全性方面不如rds（等保三级）

2、proxy方面不如rds

3、分布式事务支持不如rds

4、可扩展性不如rds

优势：

1、开源、社区活跃

- 2、读写分离、主从模式支持灵活，功能较强；
- 3、ER分片的创新；
- 4、重量级功能支持，功能丰富，集思广益；

四、方案方面

- 1、故障自动纳管问题；
 - 2、服务注册与管理目前没有，依赖zk目前对于zk的融合不是很好。对于服务的故障切换等变动不能有很好的反映与zk管理；
 - 3、依赖配置，目前依赖于手工操作，包括节点、读写分离、主从结构，没有友好的操作界面，服务对于以上动态变化信息没有有效反映与配置中；
 - 4、容器环境的支持目前尚存在一个bug，因自动切换造成服务为有状态服务，对于容器环境的支持不是很友好；
 - 5、弹性伸缩目前没有，无法根据监控性能情况自动调度资源；
- 针对以上情况可以深入的去实现加以改进
- 思路在于服务活动与配置中心的watch变化同步机制的建立；

五、代码方面

- 1、分布式事务 2pc实现； 对比3pc，对比维护守护副本两种方案；对比阿里X-DB paxos强一致；
- 2、高可用部分心跳检测与故障切换； 心跳这里最应该与配置中心实现watch监控同步机制，以灵活反映整个集群状态；

六、UCLLOUD DaaS 方面差异

- 1、版本15年上线后，需求基本不变，二期做了一次大的升级。二期升级部分包括自研与引入mycat开源新功能。 版本维护以项目需求为主，持续性与深度不强。mycat社区活跃度与持续度较高。

2、版本差异，ucloud项目有基于云平台的服务自注册与发现的CM 组件支持，mycat独立运行。 ucloud有基于服务调用与服务参数设置的ws 服务接口支持，mycat无ws 有webconsole监控配置。

3、故障纳管流程，ucloud项目基于项目需求屏蔽自动切换，转入管理平台运维流程，有一套自己的体系，mycat 基于技术层面做自动切换纳管。