**Mycat-cluster组件设计方案简述**

--QING\_\_\_\_

Mycat-cluster为Mycat分布式集群基础组件,可以让Mycat存储系统的各个节点server互相协同,目前Mycat-cluster支持的特性如下:

1) 集群中Mycat节点服务状态跟踪:可以方便的查看集群中各个节点的server状态,以及所有节点之间的网络关系.当节点状态异常时,可以获得通知消息等.

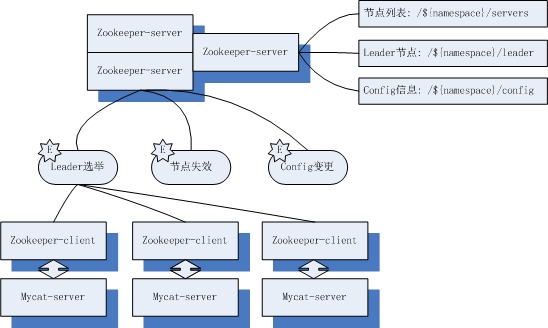
2) Leader选举,以及Follower节点的双向通讯: leader和Follower之前需要进行其他数据通信和消息发送.

3) 集群节点动态管理,比如动态的新增节点/移除节点等.

4) 支持optional配置文件的集群统一管理: 对于非核心配置文件,可以统一管理,对于配置文件变更,可以有效的同步到的所有节点上,并动态加载.

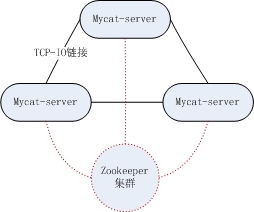
**一.整体设计**

系统服务的分布式管理,可以使用使用zookeeper作为底层支持,zookeeper作为集群服务管理是比较合适的选择.可能用户的环境并没有zookeeper支持,我们在将来会支持基于IO的cluster管理.



F1. 逻辑组件结构图

每个Mycat节点启动时,都可以选择是否挂载启动zookeeper客户端,zk客户端的主要作用,就是和zookeeper集群通讯,用来参与leader选举,以及响应leader失效/节点变更等事件.



F2. Cluster简析

集群中每个Mycat节点都会和zookeeper集群保持一个TCP长链接.此外每个mycat节点之间也需要建立一个有向的TCP链接..因为网络环境的复杂性,可能导致mycat与zookeeper之间的网络不通,但是每个mycat之间的网络却是通常的;我们需要确保在zookeeper集群短时间异常的情况下,不能过度的干扰服务的支撑能力.

**二. 配置**

mycat.cluster=on //”on” : 开启cluster,”off”关闭

mycat.cluster.zookeeper.connectString=127.0.0.1:2181 //zookeeper集群的链接地址.

mycat.cluster.zookeeper.sessionTimout=30000 //zookeeper

mycat.cluster.namespace=db/sns //当前应用的分组ID