

1、原理

mha适合一主多从

主从同步，注意数据一致性。5.6以上版本Second behind master这个数值不靠谱，忽略了i/o延迟

该软件由两部分组成：MHA Manager（管理节点）和MHA Node（数据节点）。MHA Manager可以单独部署在一台独立的机器上管理多个master-slave集群，也可以部署在一台slave节点上。MHA Node运行在每台MySQL服务器上，MHA Manager会定时探测集群中的master节点，当master出现故障时，它可以自动将最新数据的slave提升为新的master，然后将所有其他的slave重新指向新的master。整个故障转移过程对应用程序完全透明。

在MHA自动故障切换过程中，MHA试图从宕机的主服务器上保存二进制日志，最大程度的保证数据的不丢失，但这并不总是可行的。例如，如果主服务器硬件故障或无法通过ssh访问，MHA没法保存二进制日志，只进行故障转移而丢失了最新的数据。使用MySQL 5.5的半同步复制，可以大大降低数据丢失的风险。MHA可以与半同步复制结合起来。如果只有一个slave已经收到了最新的二进制日志，MHA可以将最新的二进制日志应用于其他所有的slave服务器上，因此可以保证所有节点的数据一致性。

2、切换过程

1.从宕机的master中保存二进制文件

2.检测含有最新日志更新的slave

3.应用差异的中继日至（relay log）到其他的slave

4.应用从master中保存的二进制日至事件到其他的slave中

5.提升一个slave为master

6.使其他的slave指向最新的master进行复制。

3、manager工具

masterha\_check\_ssh 检查MHA的SSH配置状况

masterha\_check\_repl 检查MySQL复制状况

masterha\_manger 启动MHA

masterha\_check\_status 检测当前MHA运行状态

masterha\_master\_monitor 检测master是否宕机

masterha\_master\_switch 控制故障转移（自动或者手动）

masterha\_conf\_host 添加或删除配置的server信息

4、node工具

save\_binary\_logs 保存和复制master的二进制日志

apply\_diff\_relay\_logs 识别差异的中继日志事件并将其差异的事件应用于其他的slave

filter\_mysqlbinlog 去除不必要的ROLLBACK事件（MHA已不再使用这个工具）

purge\_relay\_logs 清除中继日志（不会阻塞SQL线程）

 install plugin rpl\_semi\_sync\_master soname ‘semisync\_master.so’;

 install plugin rpl\_semi\_sync\_slave soname ‘semisync\_slave.so’;

set global read\_only=1;