

这周我们来说说MySQL数据库的高可用架构如何搭建，上一周我们已经聊了MySQL的主从复制架构如何搭建了，说白了，就是允许主库把数据复制到从库上去，然后允许我们的系统往主库写入数据，从从库读取数据，实现一个读写分离的模式。

那么读写分离的模式确定了，接着就可以来考虑一下数据库的高可用架构了，所谓的高可用就是说，如果数据库突然宕机了一台机器，比如说主库或者从库宕机了，那么数据库还能正常使用吗？

其实如果从库宕机了影响并不是很大，因为大不了就是让所有的读流量都从主库去读就可以了，但是如果主库宕机了呢？那就真的麻烦了，因为主库一旦宕机，你就没法写入数据了，从库毕竟是不允许写入的，只允许读取。

所以有没有一种办法，可以在主库宕机之后，就立马把从库切换为主库呢，然后所有人都对从库切换为主库去写入和读取呢？如果能实现这样的一个效果，那数据库不就实现高可用了吗？没错，就这么简单，这就是数据库的高可用架构。

一般生产环境里用于进行数据库高可用架构管理的工具是MHA，也就是Master High Availability Manager and Tools for MySQL，是日本人写的，用perl脚本写的一个工具，这个工具就是专门用于监控主库的状态，如果感觉不对劲，可以把从库切换为主库。

这个MHA自己也是需要单独部署的，分为两种节点，一个是Manager节点，一个是Node节点，Manager节点一般是单独部署一台机器的，Node节点一般是部署在每台MySQL机器上的，因为Node节点得通过解析各个MySQL的日志来进行一些操作。

Manager节点会通过探测集群里的Node节点去判断各个Node所在机器上的MySQL运行是否正常，如果发现某个Master故障了，就直接把他的一个Slave提升为Master，然后让其他Slave都挂到新的Master上去，完全透明。

其实这个原理是非常简单的，不过搭建的过程非常的复杂，因此我们可能要用本周一周的时间来一步一步讲解搭建过程，大家做好心理准备。

首先，大家最好是准备4台机器，其中一台机器装一个mysql作为master，另外两台机器都装mysql作为slave，然后在每个机器上都得部署一个MHA的node节点，然后用单独的最后一台机器装MHA的master节点，整体就这么一个结构。

首先，大家得确保4台机器之间都是免密码互相通信的，这个大家可以自行搜索一下，大量的方法可以做到，就是4台机器之间要不依靠密码可以直接ssh登录上去，因为这是MHA的perl脚本要用的。

接着大家就应该部署一个MySQL master和两个MySQL slave，搭建的过程就按照之前讲解的就行了，先装好MySQL，接着进行主从复制的搭建，全部按照之前的步骤走就行了，可以选择异步复制，当然也可以是半同步复制的。

这就是今天的内容，大家可以先自动准备好上述实验环境，然后本周接下来两天就会把剩余的MHA搭建步骤和实验步骤讲解完毕。

**End**

内部资源仅限自己学习  
www.pp1sunny.top