**探索MySQL高可用架构之MHA(概念和架构篇)**

|  |
| --- |
| 本文出自51CTO博客博主[走不完的路,看不完的书!](http://51power.blog.51cto.com/) ，如有任何问题请进入博主页面互动讨论。  博文地址：<http://51power.blog.51cto.com/3549599/1664138> |

**什么是高可用性?**

很多公司的服务都是24小时\*365天不间断的。比如Call Center。这就要求高可用性。再比如购物网站，必须随时都可以交易。那么当购物网的server挂了一个的时候，不能对业务产生任何影响。这就是高可用性。

**如何处理failover?**

解释failover，意思就是当服务器down掉，或者出现错误的时候，可以自动的切换到其他待命的服务器，不影响服务器上App的运行。

**以MySQL为例,什么样的架构才能保证其高可用性呢?**

**MySQL replication with manual failover**

同步数据是采用MySQL replication的方法，在MySQL分表分块到主从已经解释。简单的说就是从库根据主库的日志来做相应的处理，保证数据的一致。通常还配合MySQL Proxy或Amoeba等进行读写分离减少服务器压力。

manual failover，显然当Master挂掉时，利用本方式是需要手动来处理failover，一般来说是将slave更改为server。

**Master-Master with MMM manager(Multi-Master Replication Manager)**

同步数据的方式是Multi-Master Replication Manager，在MySQL分表分块到主从解释，多主多从的设置，是一个loop环形，每个DB既是前一个DB的Slave又是后一个的Master。优势就在于，一个Master挂掉，也还可以继续DB操作。每个DB都可以进行读写，分散压力。

**Heartbeat/SAN**

处理failover的方式是Heartbeat，Heartbeat可以看成是一组程序，监控管理各个node间连接的网络。当node出现错误时，自动启动其他node开始服务。Heartbeat必须解决的一个问题就是split brain，在网络中的一个node down掉后，每个node都会认为其他node down掉并尝试开始服务，因为产生数据冲突。

**通过SAN来共享数据**

SAN：Storage Area Network，是一种LAN来处理大数据量的传输，提供了计算机和存储系统之间的数据传输。各个计算机组成的集群可以通过SAN共享存储。

**Heartbeat/DRBD**

处理failover的方式依旧是Heartbeat。

同步数据使用DRBD：Distributed Replicated Block Device(DRBD)是一个用软件实现的、无共享的、服务器之间镜像块设备内容的存储复制解决方案。和SAN网络不同，它并不共享存储，而是通过服务器之间的网络复制数据。

**MySQL Cluster**

MySQL Cluster也是由各个DB node组成一个cluster，在这个cluster中由网络连接。可以自由的增减node的个数来对应数据库压力。

**MySQL高可用性大杀器之MHA**

MHA(Master High Availability)目前在MySQL高可用方面是一个相对成熟的解决方案，它由日本DeNA公司youshimaton(现就职于 Facebook公司)开发，是一套优秀的作为MySQL高可用性环境下故障切换和主从提升的高可用软件。在MySQL故障切换过程中，MHA能做到在 0~30秒之内自动完成数据库的故障切换操作，并且在进行故障切换的过程中，MHA能在最大程度上保证数据的一致性，以达到真正意义上的高可用。

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。

目前MHA主要支持一主多从的架构，要搭建MHA,要求一个复制集群中必须最少有三台数据库服务器，一主二从，即一台充当master，一台充当备用 master，另外一台充当从库，因为至少需要三台服务器，出于机器成本的考虑，淘宝也在该基础上进行了改造，目前淘宝TMHA已经支持一主一从。

官方介绍：https://code.google.com/p/mysql-master-ha/

**本次架构实现功能**

a.一主库，三个从库(其中1个为备主)，实现ABBB复制

b.使用Atlas实现读写分离，主库和备主库接收写操作，从库接收读操作

c.使用Mha实现现有架构的高可用

d.使用keepalived实现vip的漂移

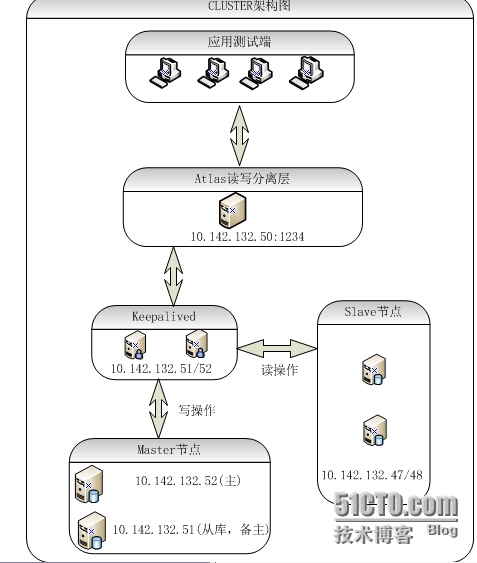
e.手工编写shell，修复Mha的不足

e1.修复当AB故障切换一次后，mha-manager会自动退出

e2.修复原主库，出问题后，修复后不能自动加入现有AB集群

e3.关于relay log的清除

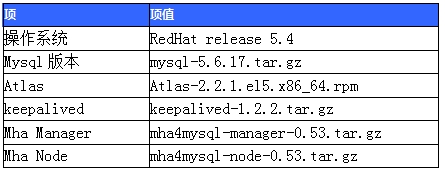
**本次实现架构图**



本次架构主机划分



软件版本



安装路径



【编辑推荐】

[防止服务器宕机时MySQL数据丢失的几种方案](http://database.51cto.com/art/201506/479889.htm)

[101个MySQL开源数据库调试和优化技巧](http://database.51cto.com/art/201506/481254.htm)

[如果对MySQL还停留在这个印象，就out了](http://database.51cto.com/art/201506/481965.htm)

[就这么简单：秒杀应用的MySQL数据库优化](http://database.51cto.com/art/201506/482015.htm)

[MySQL优化: Slave延迟很大的优化方法总结](http://database.51cto.com/art/201507/482365.htm)