## MySQL必会必知企业面试题

**此文档来面试题源于老男孩教育MySQL企业实战课程之企业必会面试题，答案来自于：鸡血哥提供参考资料，以及笔者课后复习视频整理而来,特此整理文档分享给大家, 好东西内部分享,严禁外传。**

**声明：如果觉得文档排版不行,建议自行整理为自己喜欢的格式阅读。**

### 第一天MySQL

#### 01 什么是MySQL多实例，如何配置MySQL多实例？

简单的说，就是在服务器上安装一套MySQL程序，启多个不同的服务端口（如:3306,3307），通过不同的端口对外提供服务，这就是多实例。

**如何配置MySQL多实例：**

（1）我在工作中喜欢多配置文件、多启动程序部署方案。

区别：配置文件：serverid 端口，socket文件路径，初始化指定不同的配置文件。

（2）单一配置文件、单一启动程序多实例部署方案。缺点是：耦合性太高，一个配置文件，不好管理。工作开发和运维的统一原则是：降低耦合度。

#### 02 如何加强MySQL数据的安全，请你给出可行的思路？

1、为MySQL超级用户root设置较复杂密码

2、登录不要在命令行带密码（如果不小心带了,history删除， 在当前用户的家目录下有.mysql\_history,要删除）

3、在做备份的时候，脚本必须带密码,那么要授权700 属主是root或者mysql

4、删除默认的test库

5、删除无用的用户,只保留root@localhost。如果要求严格可以删除root用户,创建其他管理用户,例如admin

6、不要一个用户管所有的库,尽量专库专用户。

7、授权最好是单库，单用户、授权用户对应的主机不要用%。权限不要给all,最小化授权，从库只给selete。

**如果是一台lamp**

1、授权一定是localhost

2、不要监听外网IP,只监听内网，通过socket的方式连接。

3、如果是SQL注入，在nginx或者php来防止

4、phpmyadmin一定要额外加验证密码，不要对外，必须用host主机，如果ip无法直接访到。

5、或者vpn连接,在访问。

6、对内程序员禁止访问生产数据库。

7、不对外的从库可以访问，但是仅限于select权限。

8、不要让开发人员获取到正式的密码。

9、程序在发布的时候，由核心人员或dba使用脚本来密码覆盖掉。

10、生产隐患百分之50来自内部安全。

#### 03 企业中MySQL root密码忘了怎么解决，多实例密码忘了又如何解决？

**单实例：**

1、首先停止MySQL

[root@db02 ~]# /etc/init.d/mysqld stop

Shutting down MySQL. SUCCESS!

2、使用**--skip-grant-tables**启动mysql，忽略授权登录验证。

mysqld\_safe --skip-grant-tables --user=mysql &

提示：在启动时加--skip-grant-tables参数，表示忽略授权表验证。

3、无需密码即可登录MySQL

[root@db02 ~]# mysql

4、修改root密码为新密码

mysql> update mysql.user set password=password('xuliangwei') where user='root' and host='localhost';

Query OK, 1 row affected (0.00 sec)

Rows matched: 1 Changed: 1 Warnings: 0

mysql> flush privileges;

Query OK, 0 rows affected (0.00 sec)

5、重启服务在登录

[root@db02 ~]# mysqladmin -uroot -pxuliangwei shutdown

[root@db02 ~]# /etc/init.d/mysqld start

Starting MySQL.. SUCCESS!

**提示：启动时加--skip-grant-tables参数启动登录修改完密码后一定要重启再对外提供服务。**

**多实例修改密码：**

1、关闭mysql

killall mysqld

2、启动时--skip-grant-tables参数

mysqld\_safe --defaults-file=/data/3306/my.cnf --skip-grant-table &

mysql -uroot -S /data/3306/mysql.sock #登录时空密码

3、修改密码的方法

mysql> update mysql.user set password=password('xuliangwei') where user='root' and host='localhost';

Query OK, 0 rows affected (0.00 sec)

Rows matched: 1 Changed: 0 Warnings: 0

mysql> flush privileges;

Query OK, 0 rows affected (0.00 sec)

4、重启服务用新密码登录

改完密码后，停止数据库，以正常启动方式重新启动并登录

killall mysqld

单实例：/etc/init.d/mysqld restart

多实例：/data/3306/mysql restart

#### 04 MySQL库授权oldboy用户管理bbs库的所有表，172.16.1.0整个网段访问，密码是123456，请给出详细命令(不低于2种方法)?

**面试官你好,我在工作中有三种方法进行授权**

授权方法一：先创建用户,然后在grant授权方法

mysql> create user xuliangwei@'localhost' identified by '123'; #创建用户并给密码

mysql> grant all on bbs.\* to xuliangwei@'localhost';

mysql> show grants for xuliangwei@'localhost'; #查看权限

授权方法二：使用grant直接授权,我会用172.16.1.%方式。

mysql> grant all on bbs.\* to xuliangwei@'localhost' identified by '123'; #授权用户

Query OK, 0 rows affected (0.00 sec)

授权方法三：如果有vlan可以更细的授权方法,%位置还可以用子网掩码的方式来授权。

mysql> flush privileges; #刷新权限表

Query OK, 0 rows affected (0.00 sec)

mysql> show grants for xuliangwei@localhost\G; #查看授权权限

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Grants for xuliangwei@localhost: GRANT ALL PRIVILEGES ON `xuliangwei`.\* TO 'xuliangwei'@'localhost'

### 第二天MySQL

#### 05 delete from test和truncate table test区别？

delete from test逻辑删除 是语句删除 一行一行删除，比较慢可恢复

truncate table test 物理删除 删除比较快 是不可逆的 无法恢复

#### 06 MySQL 的SQL语句如何优化？

面试官你好：

如果SQL语句慢了,会登录MySQL 查看 processlist；执行两下或者三下,查看同样的语句是否存在，如果存在我认为有慢的嫌疑，临时抓慢查询。

配置文件配置参数,查过若干秒查询,不走索引查询,增加慢查询日志,每天把慢查询日志收集，通过写脚本来切割mysqladmin flush，切割完毕通过mysqlsla分析,分析完每天发给我们公司的DBA，如果没有DBA就发送我和核心开发,抄送给CTO。

抓到慢查询进行优化,创建索引, explian

改善产品，改善架构的方式。

#### 07 网站打开慢，请给出排查方法，如果是因为数据库慢导致，如何排查并解决，请分析并举例？

1、我们的网站是一个集群,用户打开网站慢是单个用户慢还是大面积慢，如果是单个可能是用户问题，这个时候我们可以自己登录，或者让我们运营人员登录下。

2、如果确定是服务器问题，我会先负载均衡器到web，然后排查web服务器到存储，然后web到数据库，这些服务器接点我们都需要排查。vmstat iotop等

3、我们可以首先在负载均衡器上先访问下我的web服务器，看慢不慢,如果访问web服务器如果慢，我们就访问下静态的web服务器内容，我在服务器上放一个index.html,我在负载均衡器上访问看看，如果负载均衡上访问不慢，那在看动态不带数据库的程序例如php.info。如果慢的话代表php0fastcgi有瓶颈了，如果不慢,就访问php带数据库，如果慢就是php和数据库这段慢，还可以在负载均衡器上访问图片、附件、通过curl和wget，如果慢就存储压力大了（如果有CDN那就不存在了）。 如果是mysql数据库慢怎么办------请看上面的mysql的sql语句如何优化。

### 第三天MySQL

#### 08 MySQL Sleep 线程过多？

1、开发前段优化

2、配置文件中设置链接时间，减少等待链接超时时间

1、配置文件里修改

[mysqld]

interactive\_timeout = 120 ##此参数设置后wait\_timeout自动生效。

wait\_timeout = 120

2、其他方法：

1、php程序中，不用持久链接，即使用mysql\_connect而不是pconnect（java调整连接池）

2、php程序执行完毕，应该显示调用mysql\_close

3、逐步分析mysql的SQL查询及慢查询，找到查询过慢的SQL，优化

#### 09 key\_buffer\_size参数作用，如何修改这个参数不重启数据库就可以生效？

此参数为MyISAM引擎索引缓存的参数。

MyISAM只缓存索引，innodb既缓存索引又缓存buffer。

set global key\_buffer\_size = 1024\*1024\*32 临时生效，

修改配置文件：/etc/my.cnf

key\_buffer\_size = 32M

mysql> set global key\_buffer\_size = 1024\*1024\*32；

Query OK, 0 rows affected (0.01 sec)

mysql> show variables like '%key\_buffer%';

+-----------------+----------+

| Variable\_name | Value |

+-----------------+----------+

| key\_buffer\_size | 33554432 |

+-----------------+----------+

1 row in set (0.00 sec)

#### 10 MySQL binlog的工作模式有哪些？各有什么特点，企业中如何选择？

1.行模式：日志中会列出每一行数据被修改的形式，然后再slave端对相同的数据进行修改

优点：记录详细的执行细节，update1W条它会记录1W条，日志很大

缺点：性能不太好

2.statement level（默认）

优点：节约IO，磁盘空间 记录语句模式，数据库插入什么就记录什么，binlog比较小，对函数，触发器这样的语句复制时会出现问题，造成数据不一致。

缺点：可能在主从复制丢语句。

3.Mixed

它为前面2个的结合，它会分析语句决定用哪种模式来记录语句。在做主从同步的时候如果功能要求很多可以使用Mixed

企业中根据业务需求来进行选择。比如说更新1W条语句，它就不会以行级模式来记录

#### 11 工作中数据库被误操作执行了一个删除的SQL语句，你如何完整恢复丢失的数据？

**提示：注意解答问题的高度、宽度、实战性。**

增量恢复过程：

进入数据库，关闭数据库访问写入

1.把全备单独拷贝到一个目录下，把增量的bin-log日志移走到相同目录下

2.汇总增量bin-log日志用命令指定库拆分bin-log,如下：

mysqlbinlog -d oldboy mysql-bin.000005<bin.sql

用vi编辑bin.sql把其中错误语句删除

把sql\_log\_bin关掉

5.先把全量备份恢复回去，再把bin.sql恢复回去

6.数据恢复完成，检查数据完整性，然后再开启MySQL访问功能

**如果不允许停库的话**

如果不允许停库的话

1、停止一个从库，然后再主库刷新binlog,把mysql-bin.000014恢复成bin.sql(去掉drop语句)

2.把全备bak\_2014-11-06.sql及10点前的增量bin.sql恢复到从库。

3、切换到从库提供服务。

数据丢多少？10：10分刷新BINLOG以后的数据。mysql-bin.000015

4、把mysql-bin.000015解析为sql，恢复到从库。

以上方案还是不会主键冲突问题。

============================

1、停止一个从库，然后再主库刷新binlog,把mysql-bin.000014恢复成bin.sql(去掉drop语句)

2.把全备bak\_2014-11-06.sql及10点前的增量bin.sql恢复到从库。

3、数据丢多少？10：10分刷新BINLOG以后的数据。mysql-bin.000015

4、停止主库，快速把mysql-bin.000015解析为sql，恢复到从库。

切换到从库提供服务

### 第四天MySQL

#### 12 企业中MySQLDUMP备份时使用了-A -B参数，请问此时如何进行单表单库恢复？

找一台临时备用机器恢复，然后把需要的导出来，恢复到正式的

#### 13 详细MySQL主从同步的原理及配置主从的完整步骤。

MySQL的主从同步是异步的同步，在同步的时候，左边是主库，右边是从库，我们在上面做个标记，一般我们称master为主库，slave为从库，我以一个主一个从为例来给大家说。

首先，同步从库是需要2个线程来完成，其中一个线程叫做sql线程，另一个叫做io线程，在主库上是由一个线程负责完成主从同步，这个线程叫做io线程，要想完成主从同步，主库必须开启一样东西，开启什么啊？开启bin-log ,主从同步就是依靠bin -log来实现的,几个0忘记了，5个0 ，

当用户请求我主数据库的时候，如果是有增删改的时候，它会把我的数据写到数据库里面，然后把更新的结果写到bin-log里面，这个没问题吧？

那么主从同步时由谁找谁的呢，是由从库找主库的，我们要在从库上change master等等，其中这个地方有包含master的主机，用户，密码，端口和文件，master\_pos，在从库上要配的这些东西可以连主库，从binlog的哪个位置连主库，这个要在从库上执行；这里面有个做同步的过程我还没说，我要在从库开启开关之前，要做一个事情，我要在同步的位置点，把同步位置点之前的所有数据要灌到从库来，然后再主库备份时候的点，就是从库changemaster之后要指定的点，这个changemaster 的点就是我们后面那条命令。你在从库执行了这条命令了，在changemaster就不用指定这个点了，因为你从库已经执行完了，那么change完了我们怎么同步呢，要打开一个开关，start slave，只要打开这个开关，就开启同步了。

开启同步之后，首先是由从库的io 向主库发起请求，那主库接收到这个请求，就会验证这个请求合不合法，看你的用户名，密码，IP地址，端口，合不合法，如果全都正确就允许你连接了。

然后io线程，从库在连接的时候会告诉主库，我要从你主库的哪个位置哪个点给我发binlog，那主库的io线程收到了请求，比如我从4269开始发，他就从下面找找找，找到4269这个位置给他发binlog，发多少呢，从那个位置往下，一直到当前的情况，会一直给从库发，那从库的IO收到了这个log日志，它把这个log日志写到一个地点，这个地点在从库上叫做relay-log中继日志。它把中级日志写到这个里面了，他就不管了，它就结束了，然后从库还有一个文件交租master.info，这个文件记录什么东西呢，就是记录change master的信息，也就是说IO进程，取到的日志扔到这个里面，它把当前取到的日志的位置点更新到master.info，接下来它在向主库的iO请求，说什么呢。说我要你上一次的位置点，往下到现在的binlog，然后主库的io把这些log再给他发过来，它再把日志写到log里面，再更新master.info，就是它当前的位置点到哪了，这个masterinfo跟他是一致的

那你说这样也没有同步啊，其实从库在第一次IO线程把binlog扔到relaylog里面，这个SQL线程就已经发现了，哥们想骗我不可能，IO的小尾巴一桥它都知道，所以IO一往里面写relaylog，SQL 就发现了，-log发现了它怎么做呢，它时时的把relaylog的语句经过转换写到从库数据文件里，然后IO不断往里写，sql不断往里读并转换写到数据库，那IO呢它相当于，只要扔完了它就像主库要，主库有就发给他，它的链接就会一直保持建立状态，这就是主从同步的原理。

大家知道为什么是异步的把，它往里面一扔它就不管了。我们讲过同步和异步的例子了把。

**最后我们总结一下：**

1.同步的时候从库由2个线程来完成io 和sql，主库由1个线程来完成io

2. 要在从库上配置连接主库的ip，用户名，帐号，密码，链接的文件的位置以及pos点

3.在开启开关之前，要确保主从库 基于某一个点是一致的，这这个文件位置点以前从库是有完整数据的，

4.我changemaster连主库，我从库有吗？所有要在从库开关之前，要在主库专门建立从库的帐号

5.主库要打开binlog开关，否则没法实现同步。

6.从库打开开关的过程，就是让他2个线程工作的过程。

#### 14 生产场景不停不锁主库，不夜里操作，如何白天从容批量创建从库？

**提示：注意解答问题的高度和宽度**

创建定时任务，通过musqldump进行全备时使用参数—master-data=2进行备份

之后通过获得的位置点进行创建主从同步。

#### 15 MySQL出现同步延迟有哪些原因？如何解决？

**提示：以经历的生产案例框架解答。**

1.从库太多导致复制延迟

优化：建议从库数量3-5个为宜

2.从库硬件比主库硬件差

优化：提升硬件性能

3.慢SQL语句过多

优化：SQL语句执行时间太长，需要优化SQL语句

4.主从复制的设计问题

优化：主从复制单线程，可以通过多线程IO方案解决；另外MySQL5.6.3支持多线程IO复制。

5.主从库之间的网络延迟

优化：尽量链路短，提升端口带宽

6.主库读写压力大

优化：前端加buffer和缓存。主从延迟不同步：

不管有多延迟，只要不影响业务就没事

7、业务设计缺陷导致延迟影响业务

优化：从库没有数据改读主库

#### 16 企业生产MySQL集群架构如何设计备份方案？你是怎么做的？

**你的数据库是如何备份的（请答出200服务器以上规模企业数据库备份的解决方案）**

**提示：注意解答问题的高度和宽度**

解答：从库进行备份，停sql 线程，dump

单机备份：

1.首先开启binlog

2.每天夜里0点使用mysqldump备份 -A -B 多个库 --master-data=2 -x -F，ls -lrt|tail 查看最后binlog名字。

亮点：分库分表备份，分表分库恢复，总的恢复。

3.数据备份到本地，定时rsync推送（包括binlog分钟级或秒级别）到备份服务器。本地保留3-7天数据

4.再推送的时候通过md5sum做指纹及flag，在备份服务器做校验。把备份的结果发给你和老大的邮箱。

5.甚至定期一周拉到办公室备份，模拟还原(user 表排除)到测试环境，以确定备份是有效的。

6、数据量小于50G数据，mysqldump足矣，如果数据量大于50G，xtrabackup物理工具。还有从库直接打包的方案。

主从环境：

如果是单机环境，如果大于1台服务器一定要做主从复制。

1、主从的从本身就是备份，而且实时备份。主从同步的功能本身就是解决物理故障宕机的实时备份方案。缺点：不能防止逻辑故障数据损失。（例如：delete）

2、从库备份：建议

1)选一个不对外提供业务的从库做备份、

2)开启binlog

3)备份时可以锁表也可以停止sql\_thread.不停io\_thread,让日志不断的同步，但是不应用到从库，备份期间，如果主库挂了，那么，从库备份加上备份后的binlog就是完整的主库内容。

4)数据量小于50G，mysqldump足矣，如果数据量大于50G，xtrabackup物理工具。或者停止sql\_thread,cp或打包的方案

5)我的工作中是一主5从，3从对外LVS集群提供服务，一个从定时任务，开发用，后台用，最后一个专门做备份啥也不干，最大限度保持和主一致。

如果主库宕机，SSH补全，直接提升为主。

积极主动挖出来更深的技术。

主库宕机的恢复，从库宕机的恢复，单库单表的恢复，主从延迟监控，解决。MMM

DRBD-HA， MHA ，amoeba,开发实现读写分离。

#### 17 MySQL主从复制故障如何解决？如何监控主从复制是否故障?

**提示：注意解答问题的高度和宽度**

先stop slave set global sql\_slave\_skip\_counter 然后再 start slave

三个参数：2个yes ，1个延时，这3个也不准，主库写时间戳，从库查看

出故障解决办法：

忽略可忽略错误，在my.cnf下面的[mysqld]模块下slave-skip-errors = 1032,1062

设置跳过错误条数mysql> set global sql\_slave\_skip\_counter =1;

1.看IO 和SQL 和延时

2.在主上面写一个时间戳，在从库上进行查看比对

#### 18 MySQL如何实现双向互为主从复制A<==>B?

1.两库都需要开启binlog;从库要开启log-slave-updates;

2.CHANGE MASTER

3.两边都要创建用户和密码用来相互连接，授权对方访问

4.都要做全备；

5.都要start slave.

6.要在配置文件里面配置表的自增，偏移

#### 19 MySQL如何实现级联同步A-->B-->C？

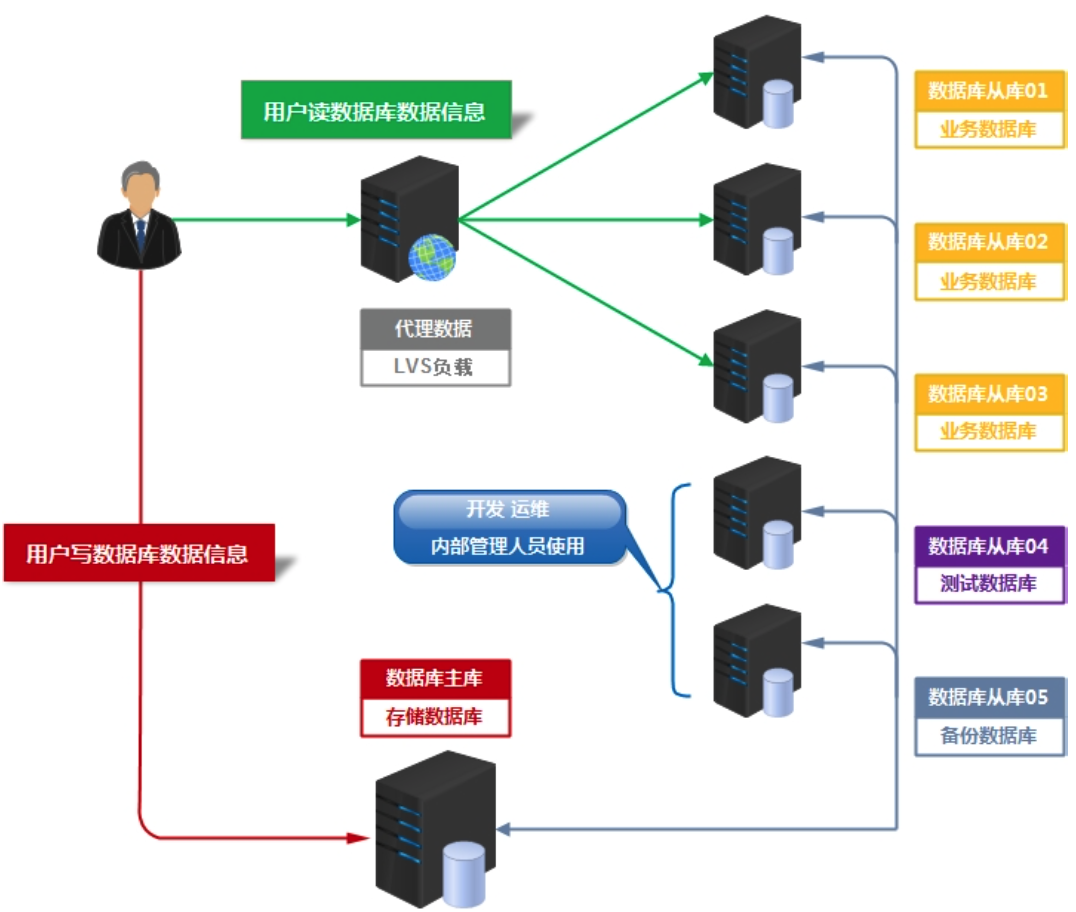
1.A、B、C里面开启 binlog开关,B还要开启log-slave-updates

2.A授权让B连接它，B上面授权让C连接它

3.B、C开启start slave和change MASTER

#### 20 MySQL数据库如何实现读写分离，你的公司是怎么实现的

**注意：请答出200服务器以上规模企业数据库的解决方案**

****

一主五从、主写从读、

三台用lvs做负载均衡，提供读服务。

一台从做后台查询、日志分析和内部人员连接

一台只做备份，开启binlog日志功能。如果写入数据就从master同步到五台slave上。怎么让他自动找读和写，可以用mysql-prokcess、omba等软件，但是这些软件都不太稳定。我们是从开发层面实现的。如果有select形式的语句就连接vip从库，如果有create、update、delete这些语句的时候就连接主库。

#### 21 生产场景,一主多从环境,从库宕机,请问你如何恢复？

从库宕机，由其他从库代替即可，我做了负载均衡

#### 22 生产场景,一主多从环境,主库宕机,请问你如何恢复（类似MHA高可用原理）

**注意：高度和宽度**

事先规定好哪个硬件比较好的从，啥都不干，等着接替，或者做成慢同步，如果没来得及做那就选，看谁跑得快就用谁，把那个不快的跟快的做一个binlog补全，不用说细节，说着思路，这个细节很多，如果需要回头我可以给你个文档。

先在所有的从库上show一下看有没有应用完毕，在应用完毕的情况下，看谁的master。info最快，就提升为主库，然后在前端改一下host分发，业务就跟过来了，从库上如果有慢同步的就优先用慢同步的。

主库宕机有两种情况：

1.如果主库可以SSH连接，BINLOG数据没丢，要把主库的BINLOG补全到所有从库。

a.提升S1为M1的操作

1)调配置read-only .授权用户select,变为增删改查，开启binlog;

如果做了双主同步，就直接切即可，啥都不用做。

2)rm -f master.info relay-log\*

3)登录数据库reset master

4)重启数据库。

b.从库到执行changemaster命令；

CHANGE MASTER TO ,MASTER\_LOG\_FILE='mysql-bin.000004',MASTER\_LOG\_POS=

2.如果主库连不上：

a.半同步从库提升为主库，半同步数据，补全到 从库

半同步从库提升主库操作间1-a步骤，所有从库CHANGE MASTER同1-b。

b. S1啥也不干的方法：

提升啥也不干的S1为主库M1 ，操作见1-a步骤，所有从库CHANGE MASTER 同1-b

c.主库宕机 ，没有实现指定主库。

选主的方法：

1）登入所有从库，分别查看

master.info

确保更新完毕，看看4个从库哪个最快，经过测试没有延迟的情况pos差距很小，甚至是一致的

2）登入看谁更新的最快；3）提升主

选相同文件选个POS 最大的最为主库

选个文件及位置点最大的为主库，补全所有从库（通过中继日志）所有其他从库和当前准备为主的数据一致。

从库继续changemaster

### 第五天MySQL

#### 23 什么是数据库的事务,事务有哪些特性？

简单说,事务就是指逻辑上的一组SQL操作,组成这组操作的各个SQL语句,执行时要么全成功要么全失败

1.原子性(Atomicity)

事务是一个不可分割的单位,事务中的所有SQL等操作要么都发生,要么都不发生

2.一致性(Consistency)

事务发生前和发生后,数据的完整性必须保持一致

3.隔离性(Isolation)

当并发访问数据库时,一个正在执行的事务在执行完毕前,对于其它的会话是不可见的,对于并发事务之间的数据时相互隔离的。

4.持久性(Durability)

一个事务一旦被提交,它对数据库中的数据改变就是永久性的,如果出了错误,事务也不允许撤销,只能通过“补偿性事务”

#### 24 解释下有关数据库的ACID是什么意思？

ACID， 指[数据库](http://www.linuxeye.com/database/)事 务正确执行的四个基本要素的缩写.包含:原子性(Atomicity)、一致性（Consistency）、隔离性（Isolation）、持 久性（Durability）。一个支持事务（Transaction）的数据库系统，必需要具有这四种特性，否则在事务过程（Transaction processing）当中无法保证数据的正确性,交易过程极可能达不到交易方的要求.

1原子性(Atomicity)

事务是一个不可分割的单位,事务中的所有SQL等操作要么都发生,要么都不发生

2一致性(Consistency)

事务发生前和发生后,数据的完整性必须保持一致

3隔离性(Isolation)

当并发访问数据库时,一个正在执行的事务在执行完毕前,对于其它的会话是不可见的,对于并发事务之间的数据时相互隔离的。

4持久性(Durability)

一个事务一旦被提交,它对数据库中的数据改变就是永久性的,如果出了错误,事务也不允许撤销,只能通过“补偿性事务”

#### 25 MySQL有哪些常用引擎？企业中如何批量更改表的引擎？

常用的有MyISAM与Innodb,memory,NDB等

创建后引擎的更改，5.0以上：

MySQL里面修改

ALTER TABLE oldboy ENGINE = INNODB;

ALTER TABLE oldboy ENGINE = MyISAM;

独立命令：

mysql\_convert\_table\_format --user=root --password=oldboy123 --socket=/data/3306/mysql.sock --engine=MyISAM oldboy t2

#### 26 MyISAM与Innodb数据库引擎有什么特点与区别，企业中如何选择？

**MyISAM特点**

1.不支持事务（事务室指逻辑上的一组操作，组成这组操作的各个单元，要么全成功要么全失败）

2.表计锁定，数据更新时锁定整个表：其锁表机制是表级锁定，这虽然可以让锁定的实现成本很小但是也同时大大降低了其并发性能。

3.读写互相阻塞：不仅会在写入的时候阻塞读取，MyISAM还会在读取的时候阻塞写入，但读本身并不会阻塞另外的读。

4.只会缓存索引：MyISAM可以通过key\_buffer\_size缓存索引，以大大提高访问性能减少磁盘IO，但是这个缓存区只会缓存索引，而不会缓存数据。

[root@mysql ~]# grep key\_buffer /data/3306/my.cnf

key\_buffer\_size = 16M

5.读取速度较快。占用资源相对少。

6.不支持外键约束，但支持全文索引。

7.MyISAM引擎是MySQL5.5.5前缺省的存储引擎

适合的业务：

不需要事务支持的业务

读多或者写多

数据修改相对较少的业务

硬件资源比较差的机器

**InnoDB引擎特点：**

1.支持事务：支持4个事务隔离级别，支持多版本读

2.行级锁定（更新时一般是锁定当前行）：通过索引实现

3.读写阻塞与事务隔离级别相关

4.具有非常高效的缓存特性：能缓存索引，也能缓存数据。

5.整个表和主键以Cluster方式存储，组成一颗平衡树。

6.支持分区，表空间，类似Oracle数据库

7.支持外键约束，5.5以前不支持全文索引，以后支持了。

适合的业务：

需要事务支持的业务

#### 27 如何调整生产线中MySQL数据库的字符集,例如:从UTF8改成GBK，请给出完整步骤。

导出sed替换在导进来

1）mysqldump数据备份（数据量大先导出表结构，sed替换），导出数据。

2）在备份中把字符集改了给成GBK。例如：sed 替换

3）修改系统配置调整字符集生效。

4）mysql 把表结构还原，再把数据还原。

5）开发把程序改成对应的字符集。

#### 28 请描述MySQL里中文数据乱码的背后原理，以及工作中如何防止数据库数据乱码？

**中文数据乱码背后原理**

1）环境本身不对，插入了错误数据，很难解决。

2）环境对的时候插入了正确数据，客户端环境破坏了，查看乱码了。

**如何防止数据库乱码**

1、Linux客户端字符集。（UTF8）

2、Linux服务端字符集。（LANG="zh\_CN.UTF-8"）

3、数据库客户端字符集。（UTF8）

4、数据库服务端字符集。

5、具体数据库的字符集。

6、表的字符集。

7、PHP/Java程序字符集。

#### 29 企业生产MySQL如何优化？

<http://www.xuliangwei.com/xubusi/213.html>

MySQL数据库的优化–运维架构师必会高薪技能，笔者近六年来一线城市工作实战经验

#### 30 MySQL集群和高可用方案有哪些，再生产场景你都用过哪些方案？

**提示：注意解答问题的高度和宽度**

主从复制：我在工作中曾经用过一主多从（主库提供主，从库提供从），我也用过双主多从，我也用过级联M-M-S ，工作的时候VIP在一个机器上面，一个宕机可以飘到另外一个上面去。我们也没什么高可用，就是一主多从

集群就是主从复制，然后加上读写分离（amoeba ,程序来做到，MySQL-proxy）,然后高可用这块是keepalive+主从复制（互为主从，keepalive来回切），keepalive+drdb,mysqlmmm,mha,pxc。

1.主从复制、读写分离、keepalive+主从复制、MHA、MySQL-proxy、MMM等

2.我们公司就是用主从复制来实现MySQL集群和高可用，读写分离这块我们是开发通过程序来实现的，然后MySQL集群前端加缓存，数据库1主5从，单独拿一台从库来做慢同步，用于主库宕机的备用机，其他从库做负载均衡，定期检查监控主从复制，开启慢查询，定时任务切换，用mysqlsla语句进行读取，把结果定时发给核心运维总监，核心开发人员，及CTO的邮箱

#### 31 你维护的企业里MySQL服务架构能说下么？

一主五从，3个用LVS做负载均衡，一个做备份，一个给开发用，程序来做读写分离，给2个VIP，一个是主一个是从，程序来连，怎么授权？三种方法：忽略授权表同步，分别授权，用户是一个，最好是忽略授权表和read only，往多说。

我们公司的MySQL服务架构，架构前面加缓存，数据库读写分离一主5从，从库做负载均衡，拿一个从库专门做主库的慢同步。用于主宕机了直接顶上使用，我做的最多的就是SQL语句调优，就是开启慢查询，然后定时任务切割，在用mysqlsla语句进行读取，并把结果定时发给核心运维总监，核心开发人员及CTO的邮箱。

#### 32 如何分表分库备份及批量恢复（口述脚本实现过程）?

<http://www.xuliangwei.com/xubusi/252.html>

企业实战Shell-MySQL分库分表备份脚本