**日志**文件：

事务日志：事务性的存储引擎

错误日志：默认开启 Error log

服务器启动和关闭中的信息

服务器运行中的错误信息

时间调度运行一个时间时产生的信息

在**从服务器**上启动**从服务器进程**时产生的信息

show global variables like ‘error’

修改必须写在配置文件中并重启

show global variableslike ‘%log%’;

中log\_warning 默认也启动

二进制日志：Binary log 默认启动

主要记录DML和DDL语句，一些影响内容或对象的语句

在复制恢复等中作用很大

max\_binlog\_size 默认最大日志大小1g flush-logs滚动日志

但如果仅仅记录语句的话例如datetime会造成结果不一样，所以最后结有两种方式：1 基于行（产生精确结果但极费资源） 2 基于语句 （结果不确定） 在show global variableslike ‘%log%’可以看到binlog\_format:**混合的**MIXED

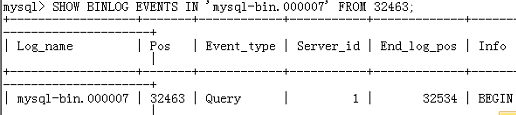
查看日志：mysqlbinlog（shell中） +名字

--start-positon

mysql中用SHOW MASTER STATUS查看正在使用的日志的位置

SHOW BINARY LOGS用来查看所有的日志，不能手工删除或者直接修改！

SHOW BINLOG EVENT [IN ‘logname’] [FROM pos] [LIMIT [offset,] row\_count]



使用**PURGE BINARY LOGS** 来删除二进制文件TO ‘XXX’ 到XXX之前的日志都删除了 或者BEFORE ‘datetime’

查询日志：默认不启动 general\_log **一般不启动** 否则太大

慢查询日志：**默认不启动 一般开启** slow\_query\_log 短暂的开启方法：set slow\_query\_log on

log\_output改变记录的位置（有可能用到）

中继日志：中继的服务器启用

**备份：**

备份类型：

热备：

温备：

冷备：

第二中分类类型：

逻辑备份：纯文本文件 mysqldump

优点：容易恢复 兼容性强 **独立于存储引擎**

不同平台恢复， 平台无关

缺点：

耗费更多的资源（cpu等） 备份结果文件比较大 浮点数丢失精度

物理备份：文件（数据文件，日志文件）

优点：备份简单，直接copy即可，速度快

缺点：结果会更大，跨平台取决于存储引擎

第三种分类类型：

完全备份：

增量备份：

差异备份：以上一次备份起始开始备份，占用空间较大 但恢复简单

一定要测试此次备份可用再删除之前的备份

**应该备份什么东西：**

**二进制文件：**

**事务日志**：对于innoDB这种需要备份事务日志 在RAID1或者RAID5两块磁盘分开保存两份

**代码：**存储过程，存储函数，触发器等等

**备份配置文件：**svn，git等 不仅仅包括my.cnf等

**数据**本身：

**自动执行脚本：**cron，关联到mysql的用户和组，管理脚本

**常见备份工具：**

mysqldump

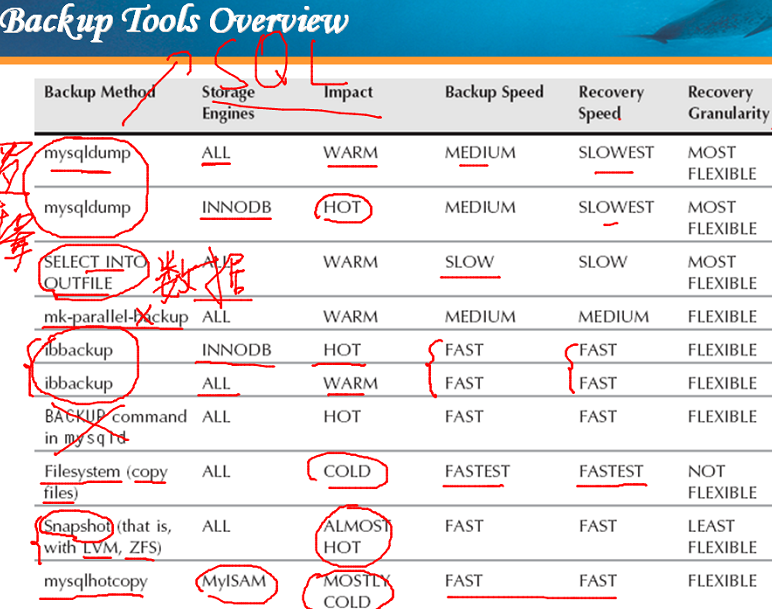
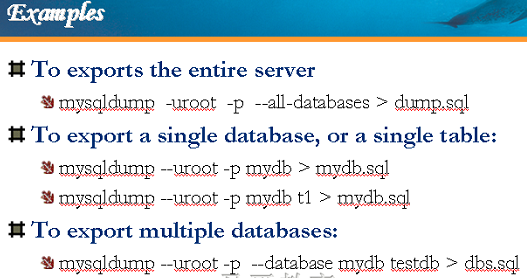
逻辑备份，恢复极慢但恢复之前处理效果较好，对innoDB热备，其他温备

ibbackup

买不起

完全备份是逻辑或者物理备份

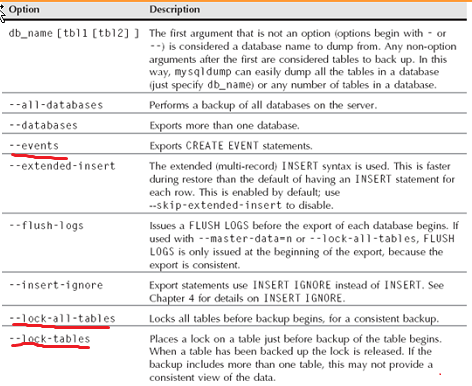
逻辑备份是



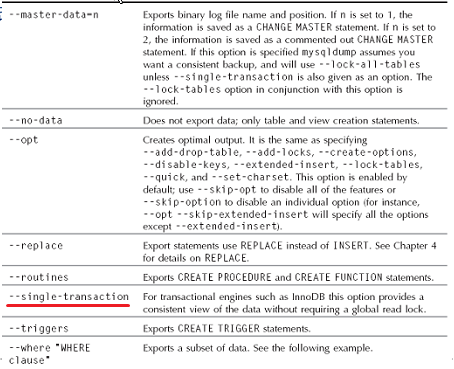
使用mysqldump备份：

innoDB支持热备

先锁表 再备份



备份一个库中的多张表不能分开一次锁定一个备份

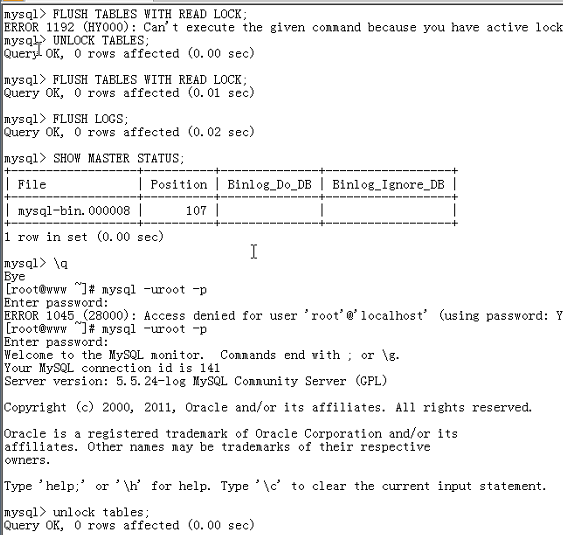


红线处是备份热备的关键

triggers是触发器

--no-date 只备份表结构不备份数据-

myisam备份必须先加锁，记录下来起始点 瑕疵接着 再dump 再unlock



mkdir /etc/backup

cd

vim bachtestdb.sh

MYDUMP=/usr/local/mysql/bin/mysqldump

MYSQL=/usr/local/mysql/bin/mysql

HOSTNAME=localhost

USERNAME=root

PASSWORD=admin

$MYSQL -u $USERNAME -h $HOSTNAME --password=”$PASSWORD” -e “FLUSH TABLES WITH READ LOCK”



$MYSQL -u $USERNAME -h $HOSTNAME --password=”$PASSWORD” -e “SHOW MASTER STATUS” > position

echo ‘date’>> position



但这样是不可能的 **进程结束锁将被自动释放**



指定=2为记录结束点

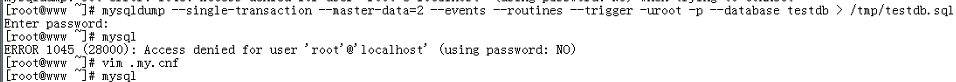
vim /root/testdb.sql就可以看到

事务性备份：热备

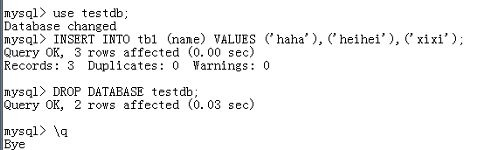


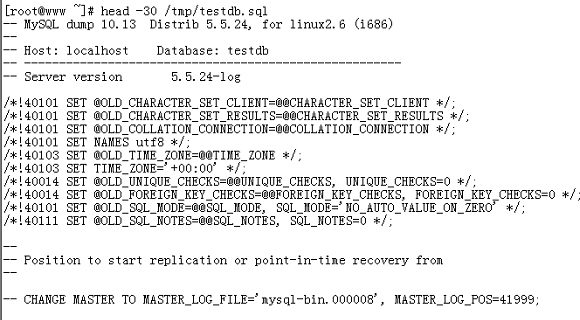
--single-transaction 备份 不需要--lock-all-tables了

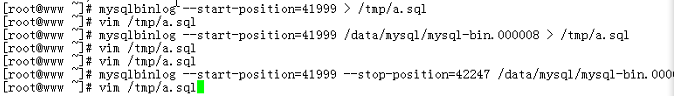
将二进制日志暂时关闭 ？为何？



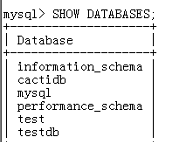






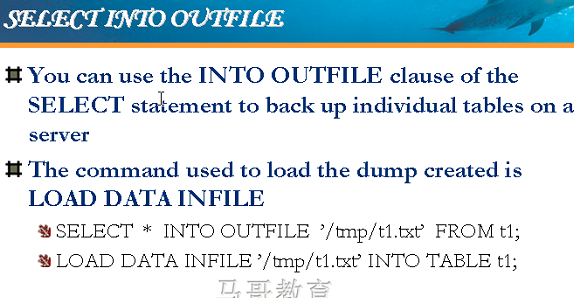


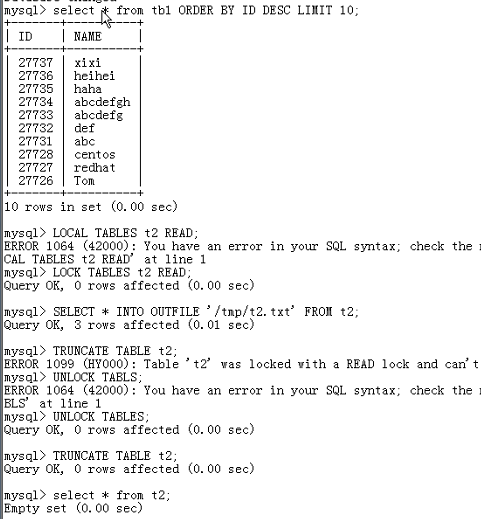


回来了

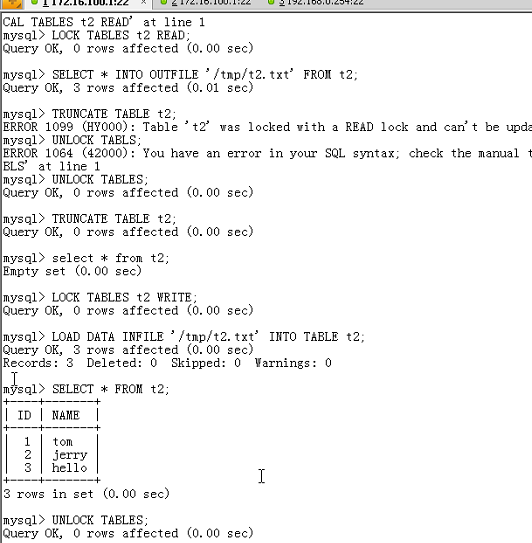
相当依赖于binlog

select方式：





来恢复



**还原：**