

mysqldump全备加增备

1. 全备：

`mysqldump --lock-all-tables --flush-logs --master-data=2 -u root -p test > backup.sql` (适用于MyISAM存储引擎)

`mysqldump --single-transaction --flush-logs --master-data=2 -u root -p test > backup.sql` (适用于InnoDB存储引擎)

`--flush-logs`为结束当前日志，生成新日志文件；

`--master-data=2` 选项将会在输出SQL中记录下完全备份后新日志文件的名称

其中参数`--master-data=[0|1|2]`

0: 不记录

1: 记录为CHANGE MASTER语句

2: 记录为注释的CHANGE MASTER语句

Usage: `mysqldump [OPTIONS] database [tables]`

OR `mysqldump [OPTIONS] --databases [OPTIONS] DB1 [DB2 DB3...]`

OR `mysqldump [OPTIONS] --all-databases [OPTIONS]`

For more options, use `mysqldump --help`

```
/usr/local/mysql/bin/mysqldump -uroot -p --quick --events --all-databases --flush-logs --delete-master-logs --single-transaction  
>
```

加上 `-delete-master-logs` 则清除以前的日志，以释放空间。但是如果服务器配置为镜像的复制主服务器，用MySQLdump `-delete-master-logs` 删掉MySQL二进制日志很危险，因为从服务器可能还没有完全处理该二进制日志的内容。在这种情况下，使用 `PURGE MASTER LOGS`更为安全。

跨主机备份

使用下面的命令可以将host1上的sourceDb复制到host2的targetDb，前提是host2主机上已经创建targetDb数据库：

`-C` 指示主机间的数据传输使用数据压缩

`mysqldump --host=host1 --opt sourceDb | mysql --host=host2 -C targetDb`

1. 恢复完全备份

`mysql -u root -p < backup_sunday_1_PM.sql`

2. 恢复增量备份

`mysqlbinlog MySQL-bin.000002 ... | MySQL -u root -p`注意此次恢复过程亦会写入日志文件，如果数据量很大，建议先关闭日志功能

参考：<http://www.cnblogs.com/kevingrace/p/6114979.html>