数据库5

数据备份与恢复

物理备份（支持myisam存储引擎，不支持innodb（需要日志文件））

冷备：cp、tar

拷贝数据库目录

逻辑备份（支持innnodb存储引擎）：

完全备份：备份所有数据

增量备份：备份上一次备份之后产生的新数据

差异备份：备份完全备份之后产生的所有新数据

物理备份（需要两台主机操作系统一致，兼容性不高）

在192。168.4.21主机上操作

将192.168.4.21数据库文件备份发送到192.168.4.22主机

[root@node22 ~]# cp -r /var/lib/mysql /dbbak/mysql.bak

[root@node22 ~]# scp -r /dbbak/mysql.bak [root@192.168.4.22:/root](mailto:root@192.168.4.22:/root)

在192.168.4.22主机上操作

[root@node33 ~]# rm -rf /var/lib/mysql/\*

[root@node33 ~]# cp -r /root/mysql.bak/\* /var/lib/mysql/

[root@node33 ~]# chown -R mysql.mysql /var/lib/mysql(没有修改所有者和所属组，服务重启失败)

#实现数据库备份，两台主机的数据一致

逻辑备份

mysqldump 完全备份（完全备份，新增数据不会备份，备份表格加上了写锁，等数据恢复完毕，才能继续写入数据，有延迟）

mysqldump -uroot -p密码 库名 > l路径

恢复操作

mysql -uroot -p密码 库名 < 路径

表达方式

mysqldump -uroot -p密码 -A 或者(--all-databases )#导入所有库

mysqldump -uroot -p密码 数据库名 #单个库

mysqldump -uroot -p 密码 数据库名 表名 #单个表

mysqldump -uroot -p 密码 数据库1 数据库2 #多个库

在对单个表和单个库进行恢复时，需要在数据库命令行新建库，在恢复命令行加上新建库名

mysql>create database uerdb1;

[root@node33 ~]# mysql -uroot -p123456 userdb1 < /root/userdb.sql

实时备份

使用binlog日志(二进制文件日志)

日志文件默认最大存储空间为1G，存满会生成新的日志文件

修改配置文件

vim /etc/my.cnf

server\_id=100(0-255中任意数字)

log-bin

重启服务

mysql>show master status;

#查看binlog日志是否启用

生成新的日志文件(上一个日志文件未存满时，生成新的日志文件，数据记录在编号大的日志文件里面)

1. 重启服务systemctl restart mysqld
2. mysql>flush logs;

#刷新日志文件

1. [root@node22 ~]# mysqldump -uroot -p123456 --flush-logs userdb > /dbbak/db3.sql

#

1. [root@node22 ~]# mysql -uroot -p123456 -e "flush logs"

删除binlog日志(当日志文件已备份时，删除日志文件)

mysql> purge master logs to "node22-bin.000003";

#删除日志文件03之前的所有日志文件

mysql> system ls /var/lib/mysql

#在mysql命令行查看

mysql> reset master;

#删除所有日志文件，并生成新的日志文件

可修改日志存放路径

vim /etc/my.cnf

log-bin=路径

使用binlog日志恢复数据

mysql> show master status;

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

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |

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

| plj.000002 | 154 | | | |

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

position 偏移量

在数据库中执行成功的增删改，产生新的数据，存入日志文件中，偏移量会发生改变，

]#mysqlbinlog binlog日志文件 | mysql -u用户名 -p密码

例：

mysqlbinlog /root/plj.000002 | mysql -uroot -p123456

修改binlog日志文件记录sql命令的格式：

mysql>show variables like “%binlog%”;

#查看默认日志格式

binlog\_format

进入配置文件修改

vim /etc/my.cnf

binlog\_format=”mixed”

重启服务

mysql>insert into db6.t1 values(888);

mysql> system mysqlbinlog /opt/mysql/plj.000003 | grep -i insert

#查看日志文件中是否有insert命令

使用日志文件恢复部分数据

偏移量

时间

mysqlbinlog --start-positon=xxx --stop-position=xxx binlog日志文件 | mysql -uroot -p123456

#恢复部分数据（根据position）

实时热备

备份的时候不锁表(行锁)，仍可以对数据进行操作

安装percona及其依赖包（开源，可从官网下载）

存储引擎为innodb

innobackupex 完全备份

格式：

]#innobackupex 选项 存放路径

--user 用户名

--password 密码

--host 主机

--port 端口号

--no-timestamp 不用时间作为子目录文件名

--databases=库名

（存放路径会自行创建）

]#innobackupex --user root --password 123456 /allbak --no-timestamp

innobackupex 完全恢复

]#innobackupex --apply-log /root/allbak

#准备恢复日志（回滚日志）

]#innobackupex --copy-back /root/allbak

#拷贝数据

增量备份(先完全备份)

格式：]#innobackupex --user root -password 密码 --incremental 目录名 --incremental-basedir=目录名

#--incremental 备份数据存放路径 --incremental-basedir=目录名 前一次备份存放路径

例：

]#innobackupex --user root --password 123456 /fulllbak --no-timestamp

#周一完全备份

]#innobackupex --user root --password 123456 --incremental /newdir --incremental-basedir=/fulllbak --no-timestamp(周二增量备份)

[root@node22 ~]# cat /fulllbak/xtrabackup\_checkpoints

#查看备份文件配置文件

backup\_type = incremental

from\_lsn = 3322592 #开始备份位置

to\_lsn = 3327295

last\_lsn = 3327304 #结束备份位置

compact = 0

recover\_binlog\_info = 0

]#innobackupex --user root --password 123456 --incremental /new1dir --incremental-basedir=/newdir --no-timestamp（周三增量备份）

#上一次备份之后增加的新数据进行备份

]#cat /new1dir/xtrabackup\_checkpoints

backup\_type = incremental

from\_lsn = 3327295 #上一次备份结束位置开始备份

to\_lsn = 3330465

last\_lsn = 3330474

compact = 0

recover\_binlog\_info = 0

增量恢复

]#innobackupex --apply-log --redo-only 完全备份目录

]#innobackupex --apply-log --redo-only 完全备份目录 --incremental-dir=目录名

#合并日志

]#innobackupex --copy-back 完全备份目录 #拷贝数据

例：

]#innobackupex --apply-log --redo-only /root/fulllbak

]#innobackupex --apply-log --redo-only /root/fulllbak --incremental-dir=/root/newdir

]#innobackupex --apply-log --redo-only /root/fulllbak --incremental-dir=/root/new1dir

]#innobackupex --copy-back /root/fulllbak/

恢复完全备份中的单张表

1. 把删除的表按照原表结构创建出来（删掉原有表）

mysql> create table db6.t2(name char(10));

1. 删除表空间

mysql> alter table dab6.t2 import tablespace;

1. 在备份文件中导出表信息

]#innobackupex --apply-log --export /backup

1. 将导出的表信息文件拷贝到数据库目录下，并修改所有者和所属组为mysql

]#cp /backup/db6/t2.{ibd,cfg,exp} /var/lib/mysql/db6/

]#chown mysql.mysql /var/lib/mysql/db6/t2.\*

1. 导入表信息

]#mysql> alter table db6.t2 import tablespace;

1. 删除数据库目录下的表信息文件

]#rm -rf /var/lib/mysql/db6/t2.cfg

]#rm -rf /var/lib/mysql/db6/t2.exp

1. 查看表记录