第三十七章：Mysql全量备份及增量备份

**一、数据库备份概述；**

**二、数据库备份的分类；**

**三、数据库全量备份及其恢复；**

**方式一：物理文件冷备份 方式二：mysqldump热备份**

**四、数据库增量备份及其恢复；**

**方式一：二进制文件备份、mysqlbinlog恢复**

**五、生产环境Mysql数据库备份策略；**

**六、扩展：Xtarbackup工具备份Mysql数据库；**

**一、数据库备份概述；**

**数据库备份概述：**将数据库中存在的现有数据，进行存放成为副本数据，可解决数据容灾；

提高系统的高可用性和灾难恢复性，数据崩溃时，以最小代价重新恢复数据；

**造成数据丢失的原因：**程序错误、人为错误、磁盘错误、天灾人祸

**二、数据库备份的分类；**

**物理备份：**指对数据库操作系统的物理文件（数据文件、日志文件）等的备份；

冷备份：必须在数据库关闭的状态下进行备份，能够更好的保证数据库的完整性；

热备份：能够在数据库处于正常运行的情况下备份，能够更高的保证服务的可用性；

**逻辑备份：**指对数据库的逻辑组件（数据库、表、数据对象）进行备份；

完全备份：将数据进行完整的备份，包含完整的库、表、索引、视图等，需要花费更长的时间

差异备份：备份自上次完全备份到现在发生改变的数据库内容，备份的文件比完整备份的文件小，备份的速度更快；

增量备份：备份至上次完全备份或增量备份后被修改的数据库内容；

**备份方式比较：**

|  |  |  |  |
| --- | --- | --- | --- |
| 备份方式 | 完全备份 | 差异备份 | 增量备份 |
| 完全备份的状态 | 表1，表2 | 表1，表2 | 表1，表2 |
| 第一次添加内容 | 创建表3 | 创建表3 | 创建表3 |
| 备份内容 | 表1，表2，表3 | 表3 | 表3 |
| 第二次添加内容 | 创建表4 | 创建表4 | 创建表4 |
| 备份内容 | 表1，表2，表3，表4 | 表3，表4 | 表4 |

**三、数据库全量备份及其恢复；**

**方式一：物理备份（冷备份）**

[root@my ~]# systemctl stop mysqld ##备份物理文件

[root@my ~]# mkdir /opt/backup

[root@my ~]# tar zcvf /opt/backup/mysql-backup-$(date +%F).tar.gz /usr/local/mysql/data/

[root@my ~]# ls /opt/backup/

mysql-backup-2018-08-02.tar.gz

[root@my ~]# systemctl start mysqld ##模拟数据库丢失

[root@my ~]# mysql -uroot -p123123

mysql> show databases;

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

| Database |

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

| information\_schema |

| mysql |

| performance\_schema |

| sys |

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

4 rows in set (0.00 sec)

mysql> drop database mysql;

Query OK, 31 rows affected, 2 warnings (0.06 sec)

mysql> show databases;

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

| Database |

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

| information\_schema |

| performance\_schema |

| sys |

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

3 rows in set (0.00 sec)

mysql> exit

[root@my ~]# systemctl stop mysql ##恢复数据库文件

[root@my ~]# rm -rf /root/\*

[root@my ~]# mkdir /root/backup

[root@my ~]# tar zxvf /opt/backup/mysql-backup-2018-08-02.tar.gz -C /root/backup/

[root@my ~]# cp /root/backup/usr/local/mysql/data/mysql/ /usr/local/mysql/data/ -rf

[root@my ~]# chown mysql:mysql -R /usr/local/mysql/data/mysql/

[root@my ~]# systemctl start mysqld

[root@my ~]# mysql -uroot -p123123

mysql> use mysql;

mysql> show tables;

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

| Tables\_in\_mysql |

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

| columns\_priv |

...

mysql> exit

**方式二：Mysqldump命令（热备份）**

**优点：使用简单、备份还原速度快、**

**缺点：**

mysql> create database linuxfan; ##准备备份的数据库数据

Query OK, 1 row affected (0.01 sec)

mysql> show databases;

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

| Database |

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

| information\_schema |

| linuxfan |

| mysql |

| performance\_schema |

| sys |

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

mysql> use linuxfan;

Database changed

mysql> create table it(岗位 char(16),姓名 char(16),身份证号 char(48),学历 char(16),工资 int);

Query OK, 0 rows affected (0.00 sec)

mysql> insert into it values('网络工程师','张三','150404199201091132','高中','5500');

Query OK, 1 row affected (0.00 sec)

mysql> insert into it values('网络工程师','李四','150406187805271432','专科','12000');

Query OK, 1 row affected (0.00 sec)

mysql> insert into it values('java工程师','王五','111376199609101456','初中','10000');

Query OK, 1 row affected (0.00 sec)

mysql> insert into it values('网络工程师','刘备','345646199209103256','高中','15000');

Query OK, 1 row affected (0.00 sec)

mysql> insert into it values('ui工程师','关羽','178645199909101189','本科','7000');

Query OK, 1 row affected (0.00 sec)

mysql> select \* from it;

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

| 岗位 | 姓名 | 身份证号 | 学历 | 工资 |

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

| 网络工程师 | 张三 | 150404199201091132 | 高中 | 5500 |

| 网络工程师 | 李四 | 150406187805271432 | 专科 | 12000 |

| java工程师 | 王五 | 111376199609101456 | 初中 | 10000 |

| 网络工程师 | 刘备 | 345646199209103256 | 高中 | 15000 |

| ui工程师 | 关羽 | 178645199909101189 | 本科 | 7000 |

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

mysql> exit

[root@my ~]# rm -rf /opt/\*

备份所有数据库：

[root@my ~]# mysqldump -uroot -p123123 --all-databases >/opt/all.sql

mysqldump: [Warning] Using a password on the command line interface can be insecure.

备份单个数据库的所有表：

[root@my ~]# mysqldump -uroot -p123123 --databases linuxfan>/opt/linuxfan.sql

mysqldump: [Warning] Using a password on the command line interface can be insecure.

备份数据库中的单个表的结构：

[root@my ~]# mysqldump -uroot -p123123 -d linuxfan it>/opt/linuxfan\_it\_jiegou.sql

mysqldump: [Warning] Using a password on the command line interface can be insecure.

备份数据库中的单个表的内容：

[root@my ~]# mysqldump -uroot -p123123 linuxfan it>/opt/linuxfan\_it\_neirong.sql

mysqldump: [Warning] Using a password on the command line interface can be insecure.

跨服务器进行导出导入，将服务器a的数据导入到服务器b，注：b服务器的test数据库必须已经存在，不然会报错，-C 代表压缩传输；

[root@my ~]# mysqldump --host=a -uroot -p123123 --databases linuxfan -C | mysql --host=b -uroot -p123123 test

压缩备份：

[root@my ~]# mysqldump -uroot -p123123 --databases linuxfan>/dev/null |gzip >/opt/linuxfan.sql.gz

mysqldump: [Warning] Using a password on the command line interface can be insecure.

还原压缩后的数据库文件:

[root@my ~]# gunzip -c /opt/linuxfan.sql.gz |mysql -uroot -p123123 linuxfan

mysql: [Warning] Using a password on the command line interface can be insecure.

[root@my ~]# mysql -uroot -p123123 ##模拟数据库丢失

mysql> drop database linuxfan;

Query OK, 1 row affected (0.01 sec)

mysql> show databases;

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

| Database |

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

| information\_schema |

| mysql |

| performance\_schema |

| sys |

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

mysql> create database test;

Query OK, 1 row affected (0.00 sec)

mysql> exit

恢复数据库文件方式一：

[root@my ~]# mysql -uroot -p123123 -f test </opt/linuxfan\_it\_neirong.sql ##-f忽略在还原中的sql错误

恢复数据库文件方式二：

[root@my ~]# mysql -uroot -p123123

mysql> create database text; ##准备导入数据的数据库

Query OK, 1 row affected (0.00 sec)

mysql> use text;

Database changed

mysql> source /opt/linuxfan\_it\_neirong.sql ##导入数据

...

mysql> select \* from it;

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

| 岗位 | 姓名 | 身份证号 | 学历 | 工资 |

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

| 网络工程师 | 张三 | 150404199201091132 | 高中 | 5500 |

| 网络工程师 | 李四 | 150406187805271432 | 专科 | 12000 |

| java工程师 | 王五 | 111376199609101456 | 初中 | 10000 |

| 网络工程师 | 刘备 | 345646199209103256 | 高中 | 15000 |

| ui工程师 | 关羽 | 178645199909101189 | 本科 | 7000 |

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

mysql> exit

**四、数据库增量备份及其恢复；**

**方式一：二进制日志文件增量备份**

1.开启二进制日志文件；

[root@my ~]# echo -e "log-bin = /usr/local/mysql/logs/mysql\_bin \nserver\_id = 1\n max\_binlog\_size = 100M " >>/etc/my.cnf ##添加到[mysql区域内]

[root@my ~]# tail -2 /etc/my.cnf

log-bin = /usr/local/mysql/logs/mysql\_bin ##开启二进制日志文件

server\_id = 1

max\_binlog\_size = 100M  ##指定单个二进制日志文件的最大大小

[root@my ~]# systemctl restart mysqld

[root@my ~]# ls /usr/local/mysql/logs/

mysql\_bin.000001 mysql\_bin.index mysqld.log mysqld.pid ##二进制文件、索引文件

[root@my ~]# cat /usr/local/mysql/logs/mysql\_bin.index

/usr/local/mysql/logs/mysql\_bin.000001

[root@my ~]# cat /usr/local/mysql/logs/mysql\_bin.000001

?bin

...

2.查看mysql的二进制日志情况；

[root@my ~]# mysql -uroot -p123123

mysql> show master status;

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

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

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

| mysql\_bin.000001 | 154 | | | |

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

1 row in set (0.00 sec)

mysql> show master logs;

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

| Log\_name | File\_size |

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

| mysql\_bin.000001 | 154 |

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

1 row in set (0.00 sec)

mysql> flush logs; ##重新建立二进制文件日志

mysql> reset master; ##清空所有二进制文件日志

mysql> insert into it values('运维工程师','吕布','897689199609101456','本科','18000');

Query OK, 1 row affected (0.00 sec)

mysql> insert into it values('net工程师','庞统','895434198909101456','小学','8000');

Query OK, 1 row affected (0.00 sec)

mysql> select \* from it;

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

| 岗位 | 姓名 | 身份证号 | 学历 | 工资 |

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

| 网络工程师 | 张三 | 150404199201091132 | 高中 | 5500 |

| 网络工程师 | 李四 | 150406187805271432 | 专科 | 12000 |

| java工程师 | 王五 | 111376199609101456 | 初中 | 10000 |

| 网络工程师 | 刘备 | 345646199209103256 | 高中 | 15000 |

| ui工程师 | 关羽 | 178645199909101189 | 本科 | 7000 |

| 运维工程师 | 吕布 | 897689199609101456 | 本科 | 18000 |

| net工程师 | 庞统 | 895434198909101456 | 小学 | 8000 |

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

mysql> show master logs;

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

| Log\_name | File\_size |

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

| mysql\_bin.000001 | 795 |

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

1 row in set (0.00 sec)

mysql> show binlog events in 'mysql\_bin.000001'; ##查看二进制文件的变化

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

| Log\_name | Pos | Event\_type | Server\_id | End\_log\_pos | Info |

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

| mysql\_bin.000001 | 4 | Format\_desc | 1 | 123 | Server ver: 5.7.12-log, Binlog ver: 4 |

| mysql\_bin.000001 | 123 | Previous\_gtids |

...

mysql> exit

3.模拟故障，删除数据库：

[root@my ~]# mysql -uroot -p123123

mysql> drop database linuxfan; ##模拟故障，删除数据库

Query OK, 1 row affected (0.08 sec)

mysql> show databases;

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

| Database |

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

| information\_schema |

| mysql |

| performance\_schema |

| sys |

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

7 rows in set (0.00 sec)

4.恢复数据库全量备份linuxfan；

mysql> create database linuxfan; ##创建数据库，进行恢复数据

Query OK, 1 row affected (0.01 sec)

mysql> use linuxfan;

Database changed

mysql> source /opt/linuxfan\_it\_neirong.sql ##首先进行全量恢复

Query OK, 0 rows affected (0.00 sec)

...

mysql> show tables;

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

| Tables\_in\_linuxfan |

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

| it |

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

1 row in set (0.00 sec)

mysql> select \* from it;

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

| 岗位 | 姓名 | 身份证号 | 学历 | 工资 |

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

| 网络工程师 | 张三 | 150404199201091132 | 高中 | 5500 |

| 网络工程师 | 李四 | 150406187805271432 | 专科 | 12000 |

| java工程师 | 王五 | 111376199609101456 | 初中 | 10000 |

| 网络工程师 | 刘备 | 345646199209103256 | 高中 | 15000 |

| ui工程师 | 关羽 | 178645199909101189 | 本科 | 7000 |

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

mysql> exit

5.恢复数据库增量备份linuxfan；

根据position值位置进行恢复；

[root@my ~]# mysqlbinlog --start-position=1 --stop-position=795 /usr/local/mysql/logs/mysql\_bin.000001 |mysql -uroot -p123123

mysql: [Warning] Using a password on the command line interface can be insecure.

mysqlbinlog: [Warning] option 'start-position': unsigned value 1 adjusted to 4

根据时间点位置进行恢复；-d指定数据库，选项-h指定主机

[root@my ~]# mysqlbinlog --start-date="2010-09-29 18:00:00" --stop-date="2010-09-29 23:00:00" -d text -h 127.0.0.1 /usr/local/mysql/logs/mysql\_bin.000001 |mysql -uroot -p123123

将二进制日志文件中所有的数据记录全部恢复；

[root@my ~]# mysqlbinlog /usr/local/mysql/logs/mysql\_bin.000001 |mysql -uroot -p123123

[root@my ~]# mysql -uroot -p123123

mysql> use linuxfan;

Database changed

mysql> select \* from it; ##验证增量恢复的数据

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

| 岗位 | 姓名 | 身份证号 | 学历 | 工资 |

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

| 网络工程师 | 张三 | 150404199201091132 | 高中 | 5500 |

| 网络工程师 | 李四 | 150406187805271432 | 专科 | 12000 |

| java工程师 | 王五 | 111376199609101456 | 初中 | 10000 |

| 网络工程师 | 刘备 | 345646199209103256 | 高中 | 15000 |

| ui工程师 | 关羽 | 178645199909101189 | 本科 | 7000 |

| 运维工程师 | 吕布 | 897689199609101456 | 本科 | 18000 |

| net工程师 | 庞统 | 895434198909101456 | 小学 | 8000 |

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

7 rows in set (0.00 sec)

mysql> show master logs;

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

| Log\_name | File\_size |

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

| mysql\_bin.000001 | 3242 |

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

mysql> exit

[root@my ~]# mysqladmin -uroot -p123123 flush-logs ##重新建立二进制日志文件

mysqladmin: [Warning] Using a password on the command line interface can be insecure.

[root@my ~]# ls /usr/local/mysql/logs/

mysql\_bin.000001 mysql\_bin.000002

6.查看二进制日志文件的内容；

[root@my ~]# mysqlbinlog --stop-position=795 /usr/local/mysql/logs/mysql\_bin.000001 >/root/mysql\_bin\_001.sql ##查看二进制文件内容

[root@my ~]# cat /root/mysql\_bin\_001.sql

# at 617 ##二进制日志文件中记录的position值及时间

#180802 3:52:15 server id 1 end\_log\_pos 678 CRC32 0x998a3925 Table\_map: `linuxfan`.`it` mapped to number 103

# at 678

#180802 3:52:15 server id 1 end\_log\_pos 764 CRC32 0x18fd7c87 Write\_rows: table id 103 flags: STMT\_END\_F

BINLOG '

bw9iWxMBAAAAPQAAAKYCAAAAAGcAAAAAAAEACGxpbnV4ZmFuAAJpdAAF/v7+/gMI/jD+MP6Q/jAf

JTmKmQ==

bw9iWx4BAAAAVgAAAPwCAAAAAGcAAAAAAAEAAgAF/+AMbmV05bel56iL5biIBuW6nue7nxI4OTU0

MzQxOTg5MDkxMDE0NTYG5bCP5a2mQB8AAId8/Rg=

'/\*!\*/;

...

7.查看恢复数据后二进制文件的变化；

[root@my ~]# mysql -uroot -p123123

mysql> show master logs;

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

| Log\_name | File\_size |

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

| mysql\_bin.000001 | 3581 |

| mysql\_bin.000002 | 154 |

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

2 rows in set (0.00 sec)

mysql> exit

[root@my ~]# ls /usr/local/mysql/logs/mysql\_bin.\*

/usr/local/mysql/logs/mysql\_bin.000001 /usr/local/mysql/logs/mysql\_bin.000002 /usr/local/mysql/logs/mysql\_bin.index

**五、生产环境Mysql数据库备份策略；**

**策略设计思路：**

1.数据更新频繁，则应该进行较为频繁的备份；

2.数据较为重要，则在有适当更新时进行备份；

3.在数据库压力小的时段进行全量备份；

**备份方案：**

1.在每周末的某个时段使用mysqldump进行重要数据库的全量备份（最好备份单个数据库，而不是all）；

2.在平时每天的晚上时段将二进制日志文件终止（前提需要根据数据记录的生成数量合理指定单个二进制文件的大小），这样每隔24小时会生成一个二进制日志文件；

3.每周的全量备份文件和每天的二进制日志文件相加，就是这一周内数据库的整体内容；

4.出现故障时，首先恢复全量备份，可以执行mysqlbinlog命令结合实际情况根据position值或者时间点进行恢复，确保万无一失；

**备份脚本：**

[root@my ~]# vi /opt/mysql\_quan\_backup.sh ##**全量备份**

#!/bin/bash

#############quan liang beifen ##########

mkdir /opt/quanliang

mkdir /opt/quanliang/backup-$(date +%F)/

/usr/local/mysql/bin/mysqldump -uroot -p123123 linuxfan>/opt/quanliang/backup-$(date +%F)/linuxfan.sql

if [ -f /opt/quanliang/backup-$(date +%F)/linuxfan.sql ];then

echo "mysql is quan backup success on time-$(date +%F)" >>/usr/local/mysql/logs/mysql\_quan\_backup.log

else

echo "mysql is quan backup fail on time-$(date +%F)" >>/usr/local/mysql/logs/mysql\_quan\_backup.log

fi

[root@my ~]# chmod +x /opt/mysql\_quan\_backup.sh

[root@my ~]# echo "0 0 \* \* 7 /opt/mysql\_quan\_backup.sh" >>/var/spool/cron/root

[root@my ~]# systemctl restart crond

[root@my ~]# crontab -l

0 0 \* \* 7 /opt/mysql\_quan\_backup.sh

[root@my ~]# echo -e "log-bin = /usr/local/mysql/logs/mysql\_bin \nserver\_id = 1\n max\_binlog\_size = 100M " >>/etc/my.cnf  **##增量备份**

[root@my ~]# systemctl restart mysqld

[root@my ~]# vi /opt/mysql\_zeng\_backup.sh

#!/bin/bash

#############zeng liang beifen ##########

mkdir /usr/local/mysql/logs/log-$(date + %F)

myadmin -uroot -p123123 flush-logs >/dev/null

sleep 5

find /usr/local/mysql/logs/ -mmin 1 -exec cp {} /usr/local/mysql/logs/log-$(date + %F)/ \;

if [ -f /usr/local/mysql/logs/log-$(date + %F)/mysql\_bin.\* ];then

echo "mysql is zeng backup success on time-$(date +%F)" >>/usr/local/mysql/logs/mysql\_zeng\_backup.log

else

echo "mysql is zeng backup fail on time-$(date +%F)" >>/usr/local/mysql/logs/mysql\_zeng\_backup.log

fi

[root@my ~]# chmod +x /opt/mysql\_zeng\_backup.sh

[root@my ~]# echo "0 23 \* \* \* /opt/mysql\_zeng\_backup.sh" >>/var/spool/cron/root

[root@my ~]# systemctl restart crond

[root@my ~]# crontab -l

0 0 \* \* 7 /opt/mysql\_quan\_backup.sh

0 23 \* \* \* /opt/mysql\_zeng\_backup.sh

**六、扩展：Xtarbackup工具备份Mysql数据库；**

