

# 二进制日志（bin log）

## 什么是 bin log?

二进制日志文件，也叫变更日志（update log）。记录了数据库所有执行的DDL（Data Definition Language）和DML（Data Manipulation Language）等数据库更新事件的语句，但不包含没有修改任何数据的语句。

使用通用查询日志（general query log）可以记录所有语句

## 主要应用场景

- 1. 数据恢复 - 如果数据库意外停止，可以通过二进制文件来查看用户执行的操作，对数据库服务器文件做了哪些修改，然后根据二进制文件中的记录来回复数据库。
- 2. 数据复制 - master服务器把它的二进制日志传递给slaves来达到master-slave数据一致的目的。

## 查看默认情况

```
mysql> SHOW VARIABLES LIKE 'log_bin%';
```

Variable_name	Value
log_bin	ON
log_bin_basename	/var/lib/mysql/mysql-bin
log_bin_index	/var/lib/mysql/mysql-bin.index
log_bin_trust_function_creators	OFF
log_bin_use_v1_row_events	OFF

log\_bin - 是否开启日志

log\_bin\_basename - 日志文件存放路径（/var/lib/mysql文件夹下以mysql-bin.xxxxxx形式命名的文件）

log\_bin\_index - 日志的索引文件

log\_bin\_trust\_function\_creators - 是否限制 存储函数 的创建、修改、调用

```
$ pwd
/var/lib/mysql
$ ls -la | grep mysql-bin
-rw-r----- 1 mysql mysql 2290 May 24 09:45 mysql-bin.000001
-rw-r----- 1 mysql mysql 157 Jun 9 13:14 mysql-bin.000002
-rw-r----- 1 mysql mysql 38 Jun 9 13:14 mysql-bin.index
```

## 日志参数设置（永久性方式）

```
$ vi /etc/mysql/my.cnf
```

```
[mysqld]
```

```
# 替换文件名为binlog.xxxxxx
```

```
log-bin=binlog
```

```
# 文件保存时间（单位：秒）
```

```
binlog_expire_logs_seconds=60000
```

# binlog文件大小。最大和默认值是1GB。该设置不能严格控制Binlog的大小（Binlog比较靠近最大值而又遇到一个比较大的事务时，为了保证事务的完整性，可能不做换日志的动作，只将该事物的所有SQL都记录进当前日志，直到事务结束。一般采用默认值）。

```
max_binlog_size=100M
```

### 注意

数据库文件不要与日志文件放在同一个磁盘上。这样当数据库文件所在的磁盘发生故障时，可以使用日志文件恢复数据

## 修改日志文件目录

```
[mysqld]
```

```
log-bin="/var/lib/mysql/binlog/mysql-bin"
```

需要使用mysql用户创建存放日志文件的文件夹

```
$ chown -R -v mysql:mysql binlog
```

## 查看日志

查看当前的二进制日志文件列表及大小

```
mysql> SHOW BINARY LOGS;
```

Log_name	File_size	Encrypted
mysql-bin.000001	2290	No
mysql-bin.000002	157	No

使用 **mysqlbinlog** 工具查看内容（显示伪SQL）

```
$ mysqlbinlog -v /var/lib/mysql/mysql-bin.000001
```

更为方便的查询命令

```
mysql> show binlog events [IN 'log_name'] [FROM pos] [LIMIT [offset,] row_count];
```

```
mysql> show binlog events in 'mysql-bin.000001';
```

**IN 'log\_name'** - 指定要查询的binlog文件名（不指定就是第一个binlog文件）

**FROM pos** - 指定从哪个pos起始点开始查起（不指定就是从整个文件首个pos点开始算）

**LIMIT [offset]** - 偏移量（不指定就是0）

**row\_count** - 查询总条数（不指定就是所有行）

解析查询结果，以update语句为例，包含如下事件

**Query** - 负责开始一个事务（BEGIN）

**Table\_map** - 负责映射需要的表

**Update\_rows** - 负责写入数据

**Xid** - 负责结束事务

其他事件

TO BE CONTINUED...