

技术分享 | undo 太大了怎么办

原创 王雨晨 爱可生开源社区 2022-07-04 16:30 发表于上海

收录于合集

#undo 1 #undo清理 1 #undo8.0 1

作者：王雨晨

爱可生数据库工程师，负责 MySQL 日常维护及 DMP 产品支持。

本文来源：原创投稿

* 爱可生开源社区出品，原创内容未经授权不得随意使用，转载请联系小编并注明来源。

问题背景

有用户在使用 MySQL 5.7 的数据库时，遇到 undo 暴涨情况，经排查存在一条慢 SQL 执行了上万秒仍没有结束，导致后续事务产生的 undo 不能清理，越来越多。

在线 truncate undo log 已开启，将慢 SQL kill 掉之后，undo 大小超过 innodb_max_undo_log_size 设置的大小，但 undo 文件没有立即收缩。

测试验证

测试参数如下，开启 innodb_undo_log_truncate

```
mysql> show variables like '%undo%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| innodb_max_undo_log_size | 104857600 |
| innodb_undo_directory | ./ |
| innodb_undo_log_truncate | ON |
| innodb_undo_logs | 128 |
| innodb_undo_tablespaces | 3 |
+-----+-----+
5 rows in set (0.00 sec)
```

模拟 undo 增长，超过 innodb_max_undo_log_size 设置大小

```
# du -sh ./undo*
152M      ./undo001
```

```
296M    ./undo002
15M     ./undo003
```

查看官方文档undo清理策略，简单概括为以下：

- 1、启用 `innodb_undo_log_truncate` 后，超过 `innodb_max_undo_log_size` 设置大小的undo表空间被标记为截断
- 2、被标记的undo表空间的回滚段被设置为不活跃的，不能分配给新的事务；
- 3、purge线程释放不需要的回滚段
- 4、释放回滚段后，undo表空间被截断为初始大小10M

可以看到在收缩undo大小前，需要purge线程先释放回滚段，这里涉及另一个参数

`innodb_purge_rseg_truncate_frequency`，默认值128，表示purge线程每调用128次，就释放回滚段一次

此次问题背景中，该参数设置的是默认值

```
mysql> show variables like 'innodb_purge_rseg_truncate_frequency';
+-----+-----+
| Variable_name                | Value |
+-----+-----+
| innodb_purge_rseg_truncate_frequency | 128   |
+-----+-----+
1 row in set (0.01 sec)
```

所以为了尽快收缩 undo 文件，我们可以将 `innodb_purge_rseg_truncate_frequency` 值调小，提高 purge 线程释放回滚段的频率

```
//调小该值
mysql> show variables like 'innodb_purge_rseg_truncate_frequency';
+-----+-----+
| Variable_name                | Value |
+-----+-----+
| innodb_purge_rseg_truncate_frequency | 16    |
+-----+-----+
1 row in set (0.01 sec)

//达到purge线程调用次数，释放回滚段，undo表空间被截断
# du -sh ./undo*
10M    ./undo001
10M    ./undo002
15M    ./undo003
```

MySQL8.0 新增 Manual Truncation

MySQL8.0 新增支持使用 SQL 语句来管理 undo 表空间

1、需要至少三个活跃的 undo 表空间，因为要保证有两个活跃的 undo 表空间来支持 Automated Truncation

手工创建一个 undo 表空间，必须以 .ibu 结尾

```
mysql> create undo tablespace undo_003 add datafile '/data/mysql/data/3307/undo_003.ibu';
Query OK, 0 rows affected (0.27 sec)
//三个处于 active 状态的 undo 表空间
mysql> SELECT NAME, STATE FROM INFORMATION_SCHEMA.INNODB_TABLESPACES WHERE NAME LIKE '%undo%';
+-----+-----+
| NAME           | STATE |
+-----+-----+
| innodb_undo_001 | active |
| innodb_undo_002 | active |
| undo_003        | active |
+-----+-----+
3 rows in set (0.00 sec)
```

2、手工截断 undo 表空间，需要先将 undo 表空间设置为 inactive

```
//模拟 undo 增长
# du -sh ./undo*
81M      ./undo_001
157M     ./undo_002
26M      ./undo_003.ibu

mysql> ALTER UNDO TABLESPACE innodb_undo_002 SET INACTIVE;
Query OK, 0 rows affected (0.01 sec)
```

3、手工设置 inactive 后，undo 表空间被标记为截断，purge 线程会增加返回频率，快速清空并最终截断 undo 表空间，状态变为 empty

```
mysql> SELECT NAME, STATE FROM INFORMATION_SCHEMA.INNODB_TABLESPACES WHERE NAME LIKE '%undo%';
+-----+-----+
| NAME           | STATE |
+-----+-----+
| innodb_undo_001 | active |
| innodb_undo_002 | empty  |
| undo_003        | active |
+-----+-----+
//undo 文件收缩
# du -sh ./undo*
81M      ./undo_001
2.1M     ./undo_002
26M      ./undo_003.ibu
```

4、empty 状态的 undo 表空间可以重新激活使用

```
mysql> ALTER UNDO TABLESPACE innodb_undo_002 SET ACTIVE;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT NAME, STATE FROM INFORMATION_SCHEMA.INNODB_TABLESPACES WHERE NAME LIKE '%undo%';
+-----+-----+
| NAME           | STATE |
+-----+-----+
| innodb_undo_001 | active |
| innodb_undo_002 | active |
| undo_003        | active |
+-----+-----+
3 rows in set (0.01 sec)
```

5、MySQL8.0 支持删除表空间，但前提是表空间为 empty 状态

```
mysql> ALTER UNDO TABLESPACE undo_003 SET INACTIVE;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT NAME, STATE FROM INFORMATION_SCHEMA.INNODB_TABLESPACES WHERE NAME LIKE '%undo%';
+-----+-----+
| NAME           | STATE |
+-----+-----+
| innodb_undo_001 | active |
| innodb_undo_002 | active |
| undo_003        | empty  |
+-----+-----+
3 rows in set (0.01 sec)

mysql> DROP UNDO TABLESPACE undo_003;
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT TABLESPACE_NAME, FILE_NAME FROM INFORMATION_SCHEMA.FILES WHERE FILE_TYPE LIKE 'UNDO';
+-----+-----+
| TABLESPACE_NAME | FILE_NAME |
+-----+-----+
| innodb_undo_001  | ./undo_001 |
| innodb_undo_002  | ./undo_002 |
+-----+-----+
2 rows in set (0.01 sec)
```

本文关键字：#undo# #undo清理# #undo8.0#

相关推荐：

技术分享 | kubernetes operator 简介

技术分享 | kubernetes 环境测试部署 MySQL 的随想

技术分享 | MongoDB 一次排序超过内存限制的排查

关于SQLE

爱可生开源社区的 SQLE 是一款面向数据库使用者和管理者，支持多场景审核，支持标准化上线流程，原生支持 MySQL 审核且数据库类型可扩展的 SQL 审核工具。

SQLE 获取

类型	地址
版本库	https://github.com/actiontech/sqlc
文档	https://actiontech.github.io/sqlc-docs-cn/
发布信息	https://github.com/actiontech/sqlc/releases
数据审核插件开发文档	https://actiontech.github.io/sqlc-docs-cn/3.modules/3.7_auditplugin/auditplugin_development.html

更多关于 SQLE 的信息和交流，请加入官方QQ交流群：[637150065...](#)



阅读原文

喜欢此内容的人还喜欢

故障分析 | xtrabackup 多表备份报错“ too many open files ”
爱可生开源社区

