



MySQL 分配 Undo 段

爱可生开源社区

2025-01-06

👁 42

🕒 阅读5分钟

分配完回滚段，接下来该分享 Undo 段了。

作者：操盛春，爱可生技术专家，公众号『一树一溪』作者，专注于研究 MySQL 和 OceanBase 源码。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。本文基于 MySQL 8.0.32 源码，存储引擎为 InnoDB。

1. 分配缓存的 Undo 段

回滚段的内存结构中 有两个属性，都是链表：

- `insert_undo_cached`，缓存 Insert Undo 段。
- `update_undo_cached`，缓存 Update Undo 段。

缓存链表有两个作用：

- 加快分配 Undo 段的速度。
- 直接复用 Undo 段管理的 Undo 页的空间，提升空间使用率。

分配 Insert Undo 段时，先从 `insert_undo_cached` 链表头部获取一个 Undo 段。

如果获取到了，就从 `insert_undo_cached` 链表中移除这个 Undo 段。如果没有获取到，则创建新的 Undo 段。

分配 Update Undo 段时，先从 `update_undo_cached` 链表头部获取一个 Undo 段。

如果获取到了，就从 `update_undo_cached` 链表中移除这个 Undo 段。如果没有获取到，则创建新的 Undo 段。

2. 创建新的 Undo 段

如果缓存链表中没有获取到 Undo 段，就需要创建新的 Undo 段。Insert Undo 段、Update Undo 段的结构相同，怎么区分不同类型的 Undo 段呢？

Undo 段的内存结构中 有个 `type` 属性：

- `TRX_UNDO_INSERT`，表示 Insert Undo 段。
- `TRX_UNDO_UPDATE`，表示 Update Undo 段。

事务改变（插入、更新、删除）用户临时表或者用户普通表的数据，分配相应的回滚段之后，才能基于这个回滚段分配 Undo 段。

一个回滚段可以管理 1024 个 Undo 段。分配 Undo 段之前，需要先在回滚段首页的 1024 个小格子中找到一个空闲的小格子。分配 Undo 段之后，把 Undo 段首页的页号写入这个小格子，这个 Undo 段才能被这个回滚段纳入管理。

寻找空闲小格子的过程简单直接，流程如下。

第 1 步，从第一个小格子开始，遍历回滚段首页中 1024 小格子。

第 2 步，每轮循环读取一个小格子中存放的整数。

第 3 步，如果读取到的整数是 `4294967295`（代码中表示为 `FIL_NULL`），说明这个小格子是空闲的，可以用来存放即将分配的 Undo 段首页的页号。循环就此结束。

第 4 步，否则，说明这个小格子被其它 Undo 段占用，继续下一轮循环，重复第 2 ~ 3 步。

找到一个空闲的小格子之后，接下来就可以创建新的 Undo 段了，主要流程如下：

- 从回滚段所属的 Undo 表空间中分配一个页。这个页，既是 Undo 段的首页，会写入 Undo 段的相关信息；也是 Undo 页，可以写入 Undo 日志。
- 初始化 Undo 段首页中各种头信息的字段。
- Undo 段首页的页号，写入刚刚找到的空闲小格子（新创建的 Undo 段就把这个小格子占用了）。
- 创建 Undo 段的内存结构，并初始化各属性。

如果直到循环结束，都没有找到一个空闲的小格子怎么办？

这说明回滚段中 1024 个小格子都被其它 Undo 段占用，本次分配 Undo 段的操作就不能继续往下进行了。MySQL 会给客户端返回以下错误：

▼ CPP 代码解读 复制代码

```
1 (1637, 'Too many active concurrent transactions')
```

在 MySQL 日志文件中，我们会看到这样一条错误日志（经过了换行处理）：

 爱可生开源社区

LV.5

一个有深度的数据库社区 @上...

作者榜No.8 >

优秀作者

224

文章

237k

阅读

170

粉丝

关注

私信

目录 收起 ^

1. 分配缓存的 Undo 段

2. 创建新的 Undo 段

3. 加入 Undo 段链表

4. 怎么避免冲突？

5. 总结

关于 SQLE

相关推荐

docker快速搭建elasticsearch 环境

76阅读 · 3点赞

什么是时间轮？

896阅读 · 3点赞

Pandas 数据清理完全指南：从入门到精...

69阅读 · 1点赞

JAVA使用ND4J实现统计学归一化

233阅读 · 1点赞

Docker 使用技巧全解：提升容器化开发...

53阅读 · 1点赞

精选内容

JavaWeb(八)新增员工、修改员工信息...

张子栋 · 42阅读 · 1点赞

Python的暗影刺客之道

Huoya · 62阅读 · 0点赞

RocketMQ实战—9.营销系统代码初版

东阳马生架构 · 24阅读 · 0点赞

【设计模式】【行为型模式】职责链模式...

flzjkl · 38阅读 · 3点赞

这款正则表达式可视化神器，让复杂正...

LLLibra146 · 50阅读 · 1点赞

找对属于你的技术圈子

回复「进群」加入官方微信群





▼ CPP 代码解读 复制代码

```
1  2024-12-01T10:05:44.060517Z 8 [ERROR] [MY-013037] [InnoDB]
2    Cannot find a free slot for an undo log.
3    You may have too many active transactions running concurrently.
4    Please add more rollback segments or undo tablespaces.
```

3. 加入 Undo 段链表

不管是从 `insert_undo_cached`、`update_undo_cached` 两个缓存链表中获取到了 Undo 段，还是创建了新的 Undo 段，都会插入到回滚段的对应 Undo 段链表中。

Insert Undo 段插入回滚段的 `insert_undo_list` 链表头部。Update Undo 段插入回滚段的 `update_undo_list` 链表头部。

创建新的 Undo 段时，把 Undo 段首页的页号写入找到的回滚段中空闲的小格子，修改的是 Undo 表空间中回滚段的首页，这里把 Undo 段加入链表，是内存中的操作。

4. 怎么避免冲突？

用户普通表的 Undo 表空间数量由系统变量 `innodb_undo_tablespaces` 控制，默认为 2。每个 Undo 表空间中回滚段的数量由系统变量 `innodb_rollback_segments` 控制，默认为 128。2 个 Undo 表空间的回滚段数量为 256。

读写事务改变（插入、更新、删除）用户普通表的数据，如果有 256 个或者更多读写事务同时分配回滚段，就会出现 2 个或者更多读写事务从同一个回滚段分配 Undo 的情况。

为了避免出现这种情况，从回滚段的 `insert_undo_cached`、`update_undo_cached` 两个缓存链表中获取 Undo 段之前，就会申请获得这个回滚段的互斥量。

从缓存链表中获取到了 Undo 段，或者创建了新的 Undo 段，插入到回滚段的 `insert_undo_list`、`update_undo_list` 链表头部，然后再释放互斥量。

这样一来，一个读写事务从某个回滚段中分配 Undo 段的期间，其它事务想要从这个回滚段中分配 Undo 段，需要等待，也就避免了多个事务从同一个回滚段中分配 Undo 段出现冲突。

读写事务改变（插入、更新、删除）用户临时表的数据，分配 Undo 段时，也会用同样的流程避免冲突，不再赘述。

5. 总结

基于已经为用户普通表或者用户临时表分配的回滚段，分配 Undo 段的主要流程如下：

- 获得回滚段的互斥量。
- 从缓存链表（ `insert_undo_cached`、`update_undo_cached` ）的头部获取一个 Undo 段。
- 如果获取到了，从缓存链表中移除 Undo 段。
- 如果没有获取到，创建新的 Undo 段。
- Undo 段插入到回滚段的 `insert_undo_list` 或者 `update_undo_list` 链表头部。
- 释放回滚段的互斥量。


更多技术文章，请访问：opensource.actionsky.com/

关于 SQLE

SQLE 是一款全方位的 SQL 质量管理平台，覆盖开发至生产环境的 SQL 审核和管理。支持主流的开源、商业、国产数据库，为开发和运维提供流程自动化能力，提升上线效率，提高数据质量。

标签： 数据库

评论 0



登录 / 注册

 即可发布评论!



暂无评论数据

为你推荐

Undo 表空间分配回滚段

爱可生开源社区 | 1月前 | 👁 35 | 👍 点赞 | 💬 评论

数据库 后端

MySQL 核心模块揭秘 | 11 期 | InnoDB 提交事务，提交了什么？

爱可生开源社区 | 10月前 | 👁 707 | 👍 1 | 💬 评论

数据库

MySQL 核心模块揭秘 | 12 期 | 创建 savepoint

爱可生开源社区 | 10月前 | 👁 713 | 👍 点赞 | 💬 评论

数据库

MySQL 核心模块揭秘 | 13 期 | 回滚到 savepoint

爱可生开源社区 | 10月前 | 👁 580 | 👍 点赞 | 💬 评论

数据库

MySQL 核心模块揭秘 | 14 期 | 回滚整个事务

爱可生开源社区 | 10月前 | 👁 226 | 👍 点赞 | 💬 评论

数据库

学习 MySQL 必须了解的几个 Undo 概念

爱可生开源社区 | 2月前 | 👁 73 | 👍 点赞 | 💬 评论

数据库

MySQL 死锁日志详解

爱可生开源社区 | 6月前 | 👁 106 | 👍 1 | 💬 评论

数据库

MySQL 核心模块揭秘 | 04 期 | 终于要启动事务了

爱可生开源社区 | 1年前 | 👁 957 | 👍 1 | 💬 评论

数据库

MySQL 核心模块揭秘 | 08 期 | 二阶段提交 (2) commit 阶段

爱可生开源社区 | 11月前 | 👁 641 | 👍 点赞 | 💬 评论

数据库

MySQL 核心模块揭秘 | 21 期 | 行锁 (1) 快速加锁

爱可生开源社区 | 8月前 | 👁 1.5k | 👍 1 | 💬 评论

数据库

MySQL 如何插入记录的 Undo 日志？

爱可生开源社区 | 1月前 | 👁 50 | 👍 点赞 | 💬 评论

数据库

MySQL存储引擎如何完成一条更新语句的执行！

程序员飞鱼 | 4月前 | 👁 658 | 👍 29 | 💬 1

后端 面试 Java

InnoDB之UNDO LOG介绍

阿里云云栖号 | 2年前 | 👁 1.2k | 👍 2 | 💬 1

后端

MySQL 核心模块揭秘 | 01 期 | 事务的起源：事务池和管理器的初始化

爱可生开源社区 | 1年前 | 👁 1.6k | 👍 1 | 💬 评论

数据库

J. Cole 的 InnoDB 系列 - 3. InnoDB空间文件布局的基础

hashcon | 3年前 | 👁 13k | 👍 20 | 💬 1

MySQL 数据库