



MySQL 如何插入记录的 Undo 日志？

爱可生开源社区

2025-01-06

 50

 阅读4分钟

Undo 模块的第二篇，聊聊插入记录产生的 Undo 日志格式。

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. 准备工作

创建测试表：

SQL

[代码解读](#) [复制代码](#)

```
1 CREATE TABLE `t1` (  
2   `id` int unsigned NOT NULL AUTO_INCREMENT,  
3   `i1` int DEFAULT '0',  
4   PRIMARY KEY (`id`) USING BTREE,  
5   KEY `idx_i1` (`i1`)  
6 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
```

示例 SQL：

SQL

[代码解读](#) [复制代码](#)

```
1 INSERT INTO `t1` (`id`, `i1`)  
2 VALUES (50, 501);
```

2. Insert Undo 日志格式

插入一条记录到表中，首先会插入记录到主键索引，然后遍历二级索引，把记录插入到各个二级索引中。

插入记录到主键索引之前，会生成 Undo 日志，并写入 Undo 页。插入记录到二级索引，不会生成 Undo 日志。插入记录的 Undo 日志格式比较简单，如下图所示。

各属性详细说明如下：

- next_record_offset**，占用 2 字节，表示下一条 Undo 日志在 Undo 页中的偏移量。
- undo_type**，占用 1 字节，表示这条 Undo 日志的类型。插入记录产生的 Undo 日志，类型为 **TRX_UNDO_INSERT_REC**。
- undo_no**，64 位整数，压缩之后占用 1~11 字节，表示这条 Undo 日志的编号。
- table_id**，64 位整数，压缩之后占用 1~11 字节，这个属性值是表 ID，表示事务插入记录到哪个表产生的这条 Undo 日志。
- len**，32 位整数，压缩之后占用 1~5 字节，表示主键字段值的长度。
- value**，占用多少字节的存储空间，取决于主键字段的数据类型和具体值，这个属性中存储的就是主键字段值，存储时不会压缩。
- current_record_offset**，这条 Undo 日志在 Undo 页中的偏移量。

如果主键是由多个字段组成的联合主键，插入记录产生的 Undo 日志中，会按照联合主键定义的字段顺序写入所有主键字段的长度和值：len_1、value_1、len_2、value_2、...、len_N、value_N。

3. Insert Undo 日志内容

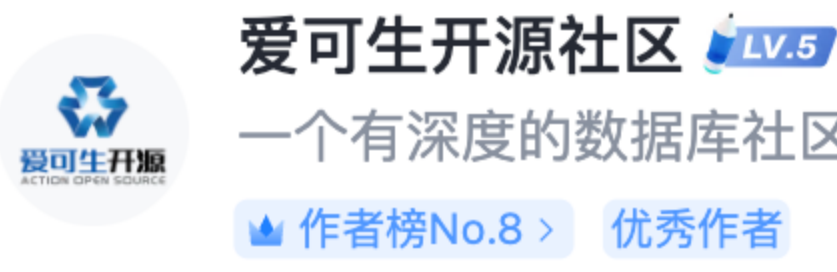
示例 SQL 插入记录到 t1 表中产生的 Undo 日志，如下图所示。

各属性值详细说明如下：

- 285**，下一条 Undo 日志在 Undo 页中的偏移量。这个值不会压缩，固定占用 2 字节。
- 11**，表示这条 Undo 日志是插入记录产生的，代码里定义为 **TRX_UNDO_INSERT_REC**。这个值不会压缩，固定占用 1 字节。
- 0**，这条 Undo 日志的编号。压缩之后占用 1 字节。
这个值来源于事务对象的 **undo_no** 属性。事务产生的第一条 Undo 日志编号为 0，第二条 Undo 日志编号为 1，依此类推。
- 1412**，这是 t1 表的 ID。压缩之后占用 2 字节。
- 4**，主键字段值的长度。压缩之后占用 1 字节。
- 50**，主键字段值。主键字段类型为 **int unsigned**，占用 4 字节。
- 272**，这条 Undo 日志在 Undo 页中的偏移量。这个值不会压缩，固定占用 2 字节。

4. Insert Undo 日志地址

InnoDB 存储引擎的表中，每条记录都有个隐藏字段 **DB_ROLL_PTR**，字段长度固定为 7 字节。通过这个字段值可以找到 Undo 日志（也是 MVCC 中记录的历史版本）。



爱可生开源社区

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](#)

- 准备工作
- Insert Undo 日志格式
- Insert Undo 日志内容
- Insert Undo 日志地址

[关于 SQLE](#)

相关推荐

- Linux部署.NET控制台应用程序
67阅读 · 1点赞
- 2024 NineData 年度总结关键字：国际...
61阅读 · 2点赞
- 修改 jmeter 内存配置 (win&mac&linux)
274阅读 · 1点赞
- 45分钟之内搞定jekyll博客系统并免费部...
105阅读 · 0点赞
- Shutil 库 2025 使用教程python在处理e...
56阅读 · 1点赞

精选内容

- 如何以低侵入方式获取业务系统使用的...
linyb极客之路 · 61阅读 · 1点赞
- Go Mongox：轻松实现 MongoDB 时间...
陈明勇 · 90阅读 · 2点赞
- 开源运营：一个多月，我是如何从 0 做到 ...
韩数 · 101阅读 · 3点赞
4. Docker镜像(详细讲解说明)
RainbowSea · 48阅读 · 0点赞
3. Docker 常用命令(详细说明附有截图)
RainbowSea · 59阅读 · 0点赞

找对属于你的技术圈子

回复「进群」加入官方微信群



从整体上来看，我们可以认为它是 Undo 日志的地址。但是，这个字段值实际上由 4 部分组成，如下图所示。

各属性详细说明如下：

- **is_insert**，表示这条 Undo 日志是否是插入记录产生的。
- **undo_space_id**，这条 Undo 日志所属 Undo 表空间的 ID。
InnoDB 最多支持 127 个 Undo 表空间，ID 范围是 0 ~ 127。7 bit 可以表示的最大数字正好是 127。
- **page_no**，这条 Undo 日志所属 Undo 页的页号。
- **offset**，这条 Undo 日志在 Undo 页中的偏移量。

DB_ROLL_PTR 的计算公式如下：

▼ CPP 代码解读 复制代码

```
1  is_insert << 55 | undo_space_id << 48 | page_no << 16 | offset
```

以示例 SQL 为例，插入记录时产生 Undo 日志得到的各属性值如下：

- is_insert = true, 转换成整数就是 1。
- undo_space_id = 2。
- page_no = 573。
- offset = 272。

用 Shell 按照以上公式计算得到 DB_ROLL_PTR，如下：

▼ SHELL 代码解读 复制代码

```
1  # 输出结果为 36591747009937680
2  echo $((1 << 55 | 2 << 48 | 573 << 16 | 272))
```

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

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

Undo 表空间分配回滚段

爱可生开源社区 | 1月前 | 34 点赞 | 评论 数据库 后端

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

MySQL 死锁日志详解

爱可生开源社区 | 6月前 | 106 点赞 | 评论 数据库

InnoDB之UNDO LOG介绍

阿里云云栖号 | 2年前 | 1.2k 点赞 | 评论 1 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 点赞 | 评论 数据库

（六）MySQL日志篇-2：MySQL的日志是做什么用的？（redo log, undo log, binlog）

奔跑的1000块 | 1年前 | 357 点赞 | 评论 5 MySQL 数据库

MySQL系列（8）— 事务原子性之UndoLog

bojiangzhou | 3年前 | 3.7k 点赞 | 评论 7 MySQL 后端

MySQL insert on duplicate 加锁分析（3）

爱可生开源社区 | 2月前 | 25 点赞 | 评论 数据库

MySQL insert on duplicate 加锁分析（2）

爱可生开源社区 | 2月前 | 30 点赞 | 评论 数据库

庖丁解InnoDB之UNDO LOG

阿里云云栖号 | 3年前 | 2.3k 点赞 | 评论 3 数据库 Java

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 点赞 | 评论 1 数据库

MySQL insert on duplicate 加锁分析（1）

爱可生开源社区 | 2月前 | 45 点赞 | 评论 数据库

MySQL Undo 工作机制历史演变

爱可生开源社区 | 5月前 | 55 点赞 | 评论 1 数据库

MySQL 在 RC 隔离级别插入记录，唯一索引冲突加什么锁？

爱可生开源社区 | 5月前 | 25 点赞 | 评论 数据库