undo log和bin log一样都属于逻辑日志,其作用是用来实现事务的回滚和MVCC,undo log在5.6之前保存在共享表 空间的回滚段中,5.6之后undo表空间可以配置成独立的文件,但是提前需要在配置文件中配置,完成数据库初始化 后生效且不可改变undo log文件的个数 当我们对数据进行修改的时候,除了会产生redo log,还会产生undo log;当事务执行失败或者是客户端要求回滚事务 时,InnoDB就可以通过undo log逻辑性地将数据回滚到事务开始时候的样子 当事务回滚时,每个INSERT对应一个DELETE,每个DELETE会对应一个INSERT、对于UPDATE,InnoDB会执行一个相 基本概念 反的UPDATE。 另外,InnoDB通过undo log来实现MVCC,也就是把数据的历史版本也记录下来。只不过这些数据的历史版本是服 etcd同样也实现了MVCC,并且我们可以随时查 务于InnoDB自身事务执行需要的,并不会永久的持久化 看某个数据的历史版本 事务的原子性是通过 undo log 实现的 MVCC是有undo log实现的,事务的隔离性是由MVCC实现的 insert undo log 我们通常会更关注update undo log,这是因为 -般会有两种格式 insert undo log只会在插入数据时产生,我们不会 有获取"插入这条数据的历史版本"这种需求 update undo log 指向下一个 undo log, 单链表连接 next type_cmpl update undo log 类型 事务 ID *undo no 记录 undo log 对应的表对象 *table id undo log 格式 info_bits 通过 next 字段可知, undo log 是通过单向链表串起来的, 当版本链很长的 *DATA_TRX_ID 旧记录的事务 ID 时候,获取数据的历史版本将成为一个耗时操作 *DATA_ROLL_PTR 旧记录的事务回滚段 对于 update undo log 的 type_cmpl 来说,通 常会有 3 种枚举 unique key 也就是 update exist record 的简写,表示当前 - TRX_UNDO_UPD_EXIST_REC -*lenN i_colN undo log 修改是未被标记为删除的记录 type_cmpl 更新的列数 n_update_field undo log与MVCC TRX_UNDO_UPD_DEL_REC -- 将 delete 的记录标记为 not delete Old Column1 (field_id, len, value) - TRX_UNDO_UPD_DEL_MARK_REC ——— 将记录标记为 delete 所更新列的旧值, 如果是标记删除 总之,undo log 中记录了非常详细的信息,其中与 MVCC 有关的几个字 的话,不会有这些字段 段就是事务 ID、旧记录的事务 ID 以及原有的数据列 Old ColumnN (field_id, len, value) n_bytes_below Old Column (field_id, len, value) start 通过undo log中记录的值,我们就可以实现MVCC了,在MySQL中,MVCC主要作 用是实现事务的隔离性,这里尤其指Repeated Read(RR)可重复读事务隔离级 别,同时也给一致性非锁定读提供了实现的可能 MVCC,Multiversion Concurrency Control,即多版本并发控制,主要用来实现事务的 · 基本概念 - 隔离性以及一致性非锁定读,后这被称为快照读。通过快照读可以极大的提高数据库的并 发性能 包含了指向下一行数据的指针, 或者说文件偏移量(offset) 变长字段长度列表 NULL 标志位 record header 回滚指针 事务 ID column 数据 记录每一个 varchar 的长度) MVCC 用于记录值为 NULL 的列号,当 table 中 |所有字段均为 NOT NULL 时,该字段将缺省 | 实现 -上图即为 InnoDB Compact 行记录格式,与 MVCC 相关的字段就是事务 ID 和回滚指针,回滚指针指向 undo log 中的记录 另一个事务 当前事务的读取 进行的修改 当我们进行数据读取时,也会有一个事务 ID 产生,若某一行数 据存在多个版本,那么当前事务只会去读取小于该事务 ID 的最 新数据 Snapshot Data v2 Snapshot Data v3 Snapshot Data v1 MVCC 解决幻读也是同样的原理,在 RR 隔离级别下,读取范围数据时需要对事务 ID 进行 比较,只要是超过了当前事务 ID 的数据行,就不在返回结果中进行记录