

问答日志

杜庆奎:20:29:40

猫哥，在不手动开启事务的情况下，还会有死锁吗

答：和手动关系不大，crud 默认也是有事务的，现在只是手动模拟死锁的情况

soso-李卓:20:33:44

和 binlog 有啥区别？

soso-李卓:20:34:00

MS 是吧

答：undo log 保存了事务发生之前的数据的一个版本，可以用于回滚，同时可以提供多版本并发控制下的读（MVCC）

Binlog 用于复制，在主从复制中，从库利用主库上的 binlog 进行重播，实现主从同步。也可以用于数据库的基于时间点的还原。

攀登:20:40:59

系统怎么知道什么时候崩的呢？

攀登:20:41:09

查 redo log?

答：系统不用知道，直接走恢复的流程

soso-李卓:20:41:19

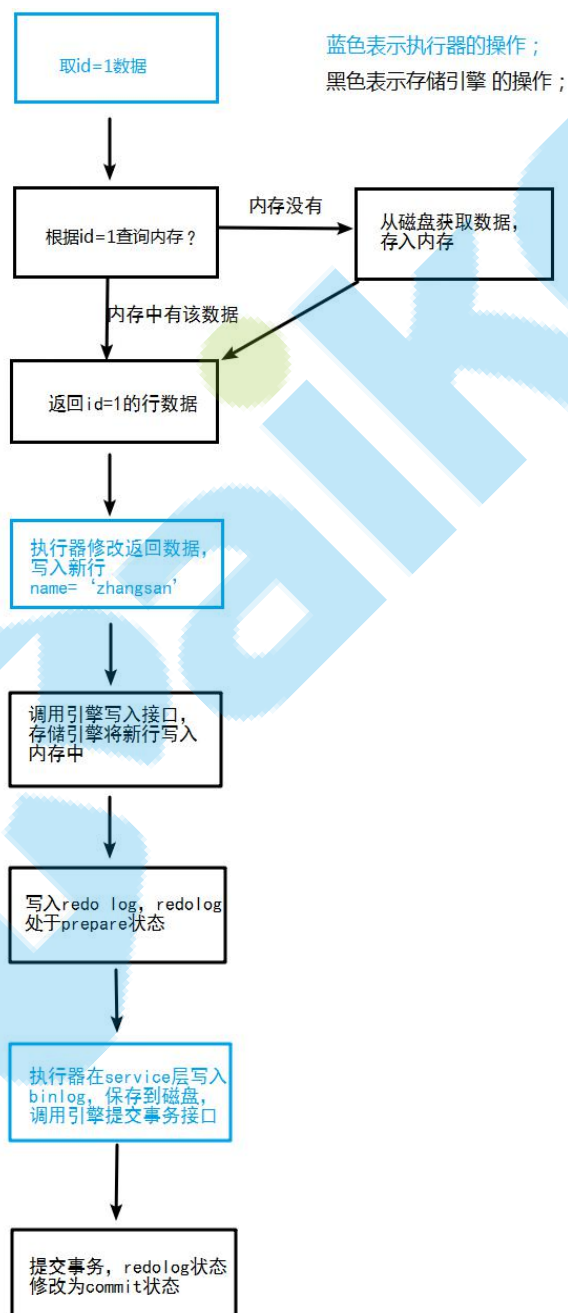
总之是 当前事务的不管了，恢复到上一次 commit 的数据 是吧

答：恢复完如果有 undo log，再根据 undo log 回滚

ERAQ:20:41:24

刷 binlog 的时机是什么时候呢？

答：



跑丢一只鞋-北京-2 年:20:41:31

commit 刷盘和 每秒刷盘 是同时存在吗??

答: 互不影响

余爽:20:41:45

undo log 存在就是为了解决不 commit 也刷盘的记录?

答: 不是的, undo log 存储历史记录

Ghost:20:42:33

redo log 恢复数据是恢复到缓存吗?

答: 用户表空间

soso-李卓:20:42:57

总之是 当前事务的不管了, 恢复到上一次 commit 的数据 上一次的数据就是 undolog 是吧

答: 不是, undolog 是没有 commit 的

ERAQ:20:43:09

事务提交之前不是也要刷 binlog 吗?

答: 对

大胜.彼得:20:43:13

前蹦的话，数据都没进表空间，回滚什么

答：如果按照每秒提交，redo log 里会有脏数据，根据 undolog 回滚到有效数据

强-上海:20:43:28

那意思是 commit 之后的 redo log 里面的 undo log 就会被清掉吗？

余爽:20:43:50

是不是 commit 了，undolog 就没有了？

答：不会马上被删，没用的时候会删，类似可达性分析

攀登:20:44:43

崩溃处理，其实就是确定要不要回滚。

大胜.彼得:20:45:06

回滚是回滚 relog 文件的数据吗？

答：undolog

湮汐:20:45:08

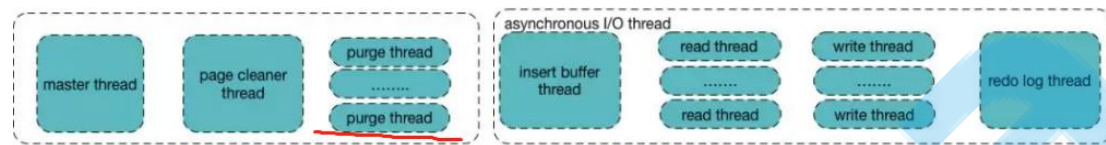
commit 之前，数据不是没有到表空间吗？为什么崩溃了还需要回滚？

答：可能每秒提交

soso-李卓:20:45:32

没听懂 什么线程?

答:



soso-李卓:20:46:15

每个事务 都有一个独立的 undolog ?

答: 不是的

雷池:20:46:51

没有活跃事务关联这个行的时候就可以删了?

答:是的

soso-李卓:20:46:58

事务和 undolog 的关系 通过事务 ID 联系?

答: 没有关联

ERAQ:20:46:59

写到 ibdata1 中

soso-李卓:20:50:53

RR 默认基本

xxzx_2655052:20:55:04

现在 RR 基本不在线上用了, 都是用 RC 吧

王助教:20:55:54

看你具体业务

黄海-成都:20:58:18

事务隔离级别，由客户端设置么？

黄海-成都:20:58:37

每个 session 需要单独设置？

答：对

大胜.彼得:20:59:40

脏读是读的 buffer pool 里数据吗

答：是

xxzx_7106658:21:01:22

select 不是不需要锁吗

答：不需要锁，但是会加锁

Ghost:21:02:15

加的还是共享读 和排他写吗

马崇-java-上海-1 年:21:03:59

这个是设置会话的隔离级别吗

马崇-java-上海-1 年:21:04:02

还是设置数据库的

答：会话

大胜.彼得:21:04:39

数据库的要改配置文件

吕伟明:21:11:47

隔离级别都是 session 的吗

答：表达有问题，是给 session 设的

觅长生:21:12:14

for update 为啥能看到当前数据

答：加锁了

Free3610:21:12:16

为啥还要 mvcc

直接用 RC 就好了？

答：解决不可重复读

杨天泽:21:12:24

隔离级别和我们上节课学的锁机制是什么关系

答：由锁实现的

马崇-java-上海-1 年:21:12:32

两个 session 要是隔离级别不一样呢

答：隔离级别主要针对读，看你读的那个是什么隔离级别

小孟-北京-4 年:21:12:39

RR 级别下，多久能读到新数据，还是一直都是快照的

答：跟多久没关系，跟你开始事物的时间有关

Ghost:21:21:34

RC 和 RR 都是 MVCC?

剩下的两个不是?

答：对

xxzx_2655052:21:22:30

sessionA 第一个查询在 sessionB 更新后呢?

答：一样的

觅长生:21:23:13

如果查询不开事务呢

答：什么查询不是在事务下的?

雷池:21:23:28

如果没有第一个 select， RC 查出来的是 3

答：你可以试下

Free3610:21:23:46

RC 就满足了

Free3610:21:24:01

性能呢

有区别吗

答：没什么区别

杨腾飞:21:24:32

不开事物，为啥就没并发？同时多个线程对一个数据修改呢？、

xxzx_2655052:21:24:43

RR 的性能会低点，因为支持间隙锁

余爽:21:24:49

不开事务是原子性操作

大胜.彼得:21:24:57

总有一个快

攀登:21:25:08

开事务，放大并发时间

吕伟明:21:24:43

update 时候不会快照吧只有 select 有，这个快照多久消失呢

答：跟多久没关系

RISE:21:27:58

oracle 是读写互相阻塞的吗?

老猫:21:28:11

不是

觅长生:21:28:31

当前读和快照都是针对的一个会话中的一个事物来说的吗

答:

在一个支持MVCC并发控制的系统中, 哪些读操作是快照读? 哪些操作又是当前读呢?

以MySQL InnoDB为例:

快照读: 简单的select操作, 属于快照读, 不加锁。(当然, 也有例外, 下面会分析) 不加读锁 读历史版本

当前读: 特殊的读操作, 插入/更新/删除操作, 属于当前读, 需要加锁。加行写锁 读当前版本

黄欣健-软件开发-东莞-4 个月:21:29:51

各位大佬, 如果 A 在快照读的时候, B 对这条记录 update 了 100, 然后提交, 这时候, A 又 update 了 100, 再读的时候, 值是多少

雷池:21:30:07

200

黄欣健-软件开发-东莞-4 个月:21:30:31

隔离级别是 可重复读

黄欣健-软件开发-东莞-4 个月:21:30:40

A 还没提交话, 也是读的 200 吗

雷池:21:30:44

update 有当前读，MVCC 对当前记录的 trxid 已经改成了 update 那个了

雷池:21:30:52

没提交是 100

黄欣健-软件开发-东莞-4 个月:21:31:15

好滴，谢谢雷哥

答：无论提交没有，都是 200

吕伟明:21:31:57

一直都是快照读吗，这个快照啥时失效呢

ERAQ:21:32:31

快照读的数据版本是基于事务开始的那个时刻的版本吧

王助教:21:32:49

是的

小冷:21:33:14

嗨起来

吕伟明:21:33:15

commit 完了事务不是已经结束了，那这个快照啥时结束，一直到下一个事务之前吗

答：快照没结束

雷池:21:33:27

RR 是开始的时候创建，RC 是每个 sql 创建一个

雷池:21:35:02

快照针对的是当前事务而言，每个事务都有一个 `transaction_id`，快照的结束指的应该是 `undo log` 的删除，也就是没有活跃事务指向操作的行吧

李永全:21:35:07

快照读在当前事务提交或加锁后读取最新版本的数据

马德成:21:47:45

高水位和低水位是怎么算出来的呢？

答: `low_limit_id` 是“高水位”，即当时活跃事务的最大 `id`，`up_limit_id` 是“低水位”，即当时活跃事务列表的最小事务 `id`

张中强:21:48:08

没有提交的才算

杜庆奎:21:48:32

事务 `id` 越大水位越高吗

Ghost:21:48:46

事务 `id` 怎么生成的呀？

跑丢一只鞋-北京-2 年:21:48:48

这两个 `id` 是 实时改变的吧

Ghost:21:48:51 +1？

答：是的

Ghost:21:49:50

那我一个事务操作两行数据 一行是 3 一行是 5 这我怎么标记在一个事务里呀？

答：不用标记在一个事务里

魏昌金:21:50:14

RC 下，每个语句指的是什么语句？

答：crud

雷池:21:52:12

undo log 是不是全部都是未提交的历史记录？

答：不是

xxzx_4684222:21:53:41

活跃的事务 id 可以不是连续的吗？比如活跃的 id 是 6 和 4，那么是不是代表 5 已经提交了？

答：对

Ghost:21:54:12

那一行数据没有事务操作的时候 这个值就一直是 0 了？

答：不操作没 log

马德成:21:54:32

所以 session2 的事务提交对于 session1 是没有影响的

张中强:21:54:39

对啊

觅长生:21:54:41

我的事务 ID 就是当前最大的事务 ID?

答：最晚开始最大

Ghost:21:55:31

我开启事务之后 操作一行数据，它会去 undo log 中看有没有这行数据先？

答：针对的是 select

觅长生:21:55:40

不是不能同时修改同一条记录吗

答：对

杨天泽:21:55:44

当前事物 id 从哪来

答：开始就有

杜庆奎:21:55:49

read view 是 undo log 的一部分吧

答：Read View 是事务开启时当前所有事务的一个集合

xxzx_2655052:21:56:42

简单来说，只能看见低于低水的数据吧？

答：不是

余爽:21:56:43

高水位是最新活跃事务，那为什么会有高于高水位的？

答：新开的事务

马德成:21:56:52

所以 session2 的事务提交对于 session1 是没有影响的么？所以 session2 提交了事务后，session1 还能读到历史版本？

答：对

攀登:21:59:06

undo log 中事务 ID 已经提交了，readview 是未提交的事务 ID

余爽:21:59:19

低于低水说明已经提交了？

ERAQ:22:02:37

read view 中的未提交事务可以看到记录的哪一个版本

答：看隔离级别，rc 最新，rr 看到开始事务是最新的

Ghost:22:03:29

RC 玩一次有一个 read view RR 下只有一个 read view

强-上海:22:03:54

RC 相当于会重新生成 read view

杜庆奎:22:03:55

以我启动的时刻为准, 如果一个数据版本是在我启动前生成的, 就认,
在我启动后生成的, 不认

黄海-成都:22:04:35

多条数据的事务 id 会一样吗?

答: 会

觅长生:22:04:41

这个是说当前事务对其他事务的可见性吗

答: 记录对事务

soso-李卓:22:05:01

同一条记录

任献良:22:05:21

记录 id 跟事务链表里的 id 比较

觅长生:22:05:26

我的 ID 是啥

雷池:22:05:30

一个事务 update 同一条记录两条呢

雷池:22:05:33

两次

雷池:22:06:01

update 一次有一个事务 id?

答：一次事务一个

强-上海:22:07:00

就是看不见比你大的，因为你比他旧

xiaoLee-hangzhou:22:07:19

隔离级别决定了从哪个记录读

跑丢一只鞋-北京-2 年:22:07:19

rc 下 这个 高水 id 和低水的 id 是实时的 根据 commit 改变的吗

答：在每个语句开始的时候，会将当前系统中的所有的活跃事务拷贝到一个列表中(read view)

xxzx_4684222:22:12:07

rc 下，当前事务，每执行一次语句，都刷新 red_view，但是事务 id 没变吧

答：对

陈冬亮-java-郑州-半年:22:16:20

Gap 是啥锁??

雷池:22:16:25 间隙锁

强-上海:22:16:33 间隙

雷池:22:18:41

排序变了。

陈冬亮-java-郑州-半年:22:19:56

它根据主键排序了吧

雷池:22:20:03

一条索引形成了索引覆盖

雷池:22:20:07

是这样吗。。

答：是

soso-李卓:22:23:39

说明 索引重排序了??

soso-李卓:22:24:21

嗯 索引应该不会变化才对

小孟-北京-4 年:22:24:50

只是看到的变了，索引树不变

马德成:22:25:18

如果同时存在多个索引，间隙是怎么算的呢?

答：后边讲

soso-李卓:22:25:45

对 上次讲过了

soso-李卓:22:27:25

细节 还是要再看一下

马德成:22:27:53

所以是锁住的是索引之间的间隙，而不是锁定索引对应的值之间的间隙？

答：对

小冷:22:28:43

觅长生:22:36:12

如果有条件 没有索引呢 怎么锁

答：锁全表

任献良:22:36:16

已经删除的数据是不是可以操作 undo log 做数据恢复？

任献良:22:38:03

我插入 5 条交易记录， 如果我删除了 也提交了。 这个时候还能查出来 ？

答：能看，提交无法恢复

陈冬亮-java-郑州-半年:22:38:08

锁住了索引与索引之间的间隙，插入的数据，索引列的值在这之间，

就插不进来了吧

答：对

马德成:22:39:44

哦哦，那么 RR 是不是比 RC 耗空间啊？

答：没有

杨天泽:22:40:00

rr 下，不同事务能看到的数据是不同的对吧

答：对

小冷:22:40:51

为啥哪个 news 表加了第三列就按照主键排序，删了就按照第二列索引排序了

答：跟那个没关系，navicate 怎么排都可以

杜庆奎:22:40:57

在我启动前启动的都能看见，在我之后启动的都看不见

马崇-java-上海-1 年:22:40:57

在 RR 下，事务开启后，什么时候可以看到新提交的更新

小孟-北京-4 年:22:41:26

只能下个 session 了

枣幸 home:22:41:36

下个 session 可以

李永全:22:41:41

加锁后可以读到是什么原因

余爽:22:41:52

当事务开始和结束期间，事务能看到的数据是一致的

小孟-北京-4 年:22:41:59

当前读

余爽:22:42:00

加锁是当前读

Ghost:22:43:25

soso-李卓:22:43:26

嗯 对 这个事务也没提交 所以看不见新数据