链滴

MySQL 学习笔记 ----- 事务隔离

作者：wky181

原文链接：https://ld246.com/article/1579077436300

来源网站：链滴

事务

首先，先说一下什么是事务？事务就是一组原子性的SQL查询，或者说一个独立的工作单元，如果数库引擎能够成功执行对数据库应用该组的全部SQL查询，那么就执行该组查询，如果有任何一条语句为奔溃或其它原因不能执行，那么所有的语句都不会执行。

事务的四个特性(ACID)

1. 原子性（ayomicity）

一个事务必须被视为一个不可分割的最小单元，整个事务中的所有操作要么全部提交成功，要么全部败回滚，这就是事务原子性。

2. 一致性（consistency）

数据库总是从一个一致性的状态转换到另外一个状态。

3. 隔离性（isolation）

通常来说，一个事务所做的修改在最终改变以前，对于其他事务是不可见的。

4. 持久性（durability）

一旦事务提交，则其所做的修改就会永久保存到数据库中，即使系统崩溃，修改数据也不会丢失。

隔离性与隔离级别

当数据库上有多个事务同时执行的时候，就有能出现脏读，不可重复读，幻读的问题，为了解决这些题，就有了**隔离级别**的概念。

SQL标准的事务隔离级别包括：读未提交（read uncommitted）、读提交（read committed）、重复读（repeatable read）和串行化（serializable ）。

- 读未提交是指，一个事务还没提交时，它做的变更就能被别的事务看到。

- 读提交是指，一个事务提交之后，它做的变更才会被其他事务看到。
- 可重复读是指，一个事务执行过程中看到的数据，总是跟这个事务在启动时看到的数据是一致的。然在可重复读隔离级别下，未提交变更对其他事务也是不可见的。
- 串行化，顾名思义是对于同一行记录，“写”会加“写锁”，“读”会加“读锁”。当出现读写锁突的时候，**后访问的事务必须等前一个事务执行完成**，才能继续执行。

用一个例子说明这几种隔离级别。假设数据表T中只有一列，其中一行的值为1，下面是按照时间顺序行两个事务的行为。

```
mysql> create table T(c int) engine=InnoDB;
insert into T(c) values(1);
```

事务A	事务B
启动事务 查询得到值1	启动事务
	查询得到值1
	将1改成2
查询得到值V1	
	提交事务B
查询得到值V2	
提交事务A	
查询得到值V3	

若隔离级别是“读未提交”，B在未提交前，A都可以看到B事务内的值，那么V1、V2、V3都是2。

若隔离级别是"读提交"，则V1是，因为事务B在未提交前对A是不可见的，事务B的更新在提交后才能A看到，所以V2、V3都是2。

若隔离级别是"可重复读”，则V1、V2都是1，因为遵循的就是这个要求：事务在执行期间看到的数

前后必须是一致的，当事务A提交后，V3看到的值就是2。

若隔离级别是“串行化”，由于事务A先开始时查询，先加读锁，所以V1是1，事务B执行“将1改成2的时候，会被锁住，因为B写锁一直加不上。直到事务A提交后，事务B才可以继续执行，所以V2是1 V3是2。

配置事务的方式是，将启动参数transaction-isolation的值设置成READ-COMMITTED。你可以用sh w variables来查看当前的值。

```
mysql> show variables like 'transaction_isolation';
+-----------------------+----------------+
| Variable_name         | Value          |
+-----------------------+----------------+
| transaction_isolation | REPEATABLE-READ |
+-----------------------+----------------+
1 row in set (0.00 sec)
```

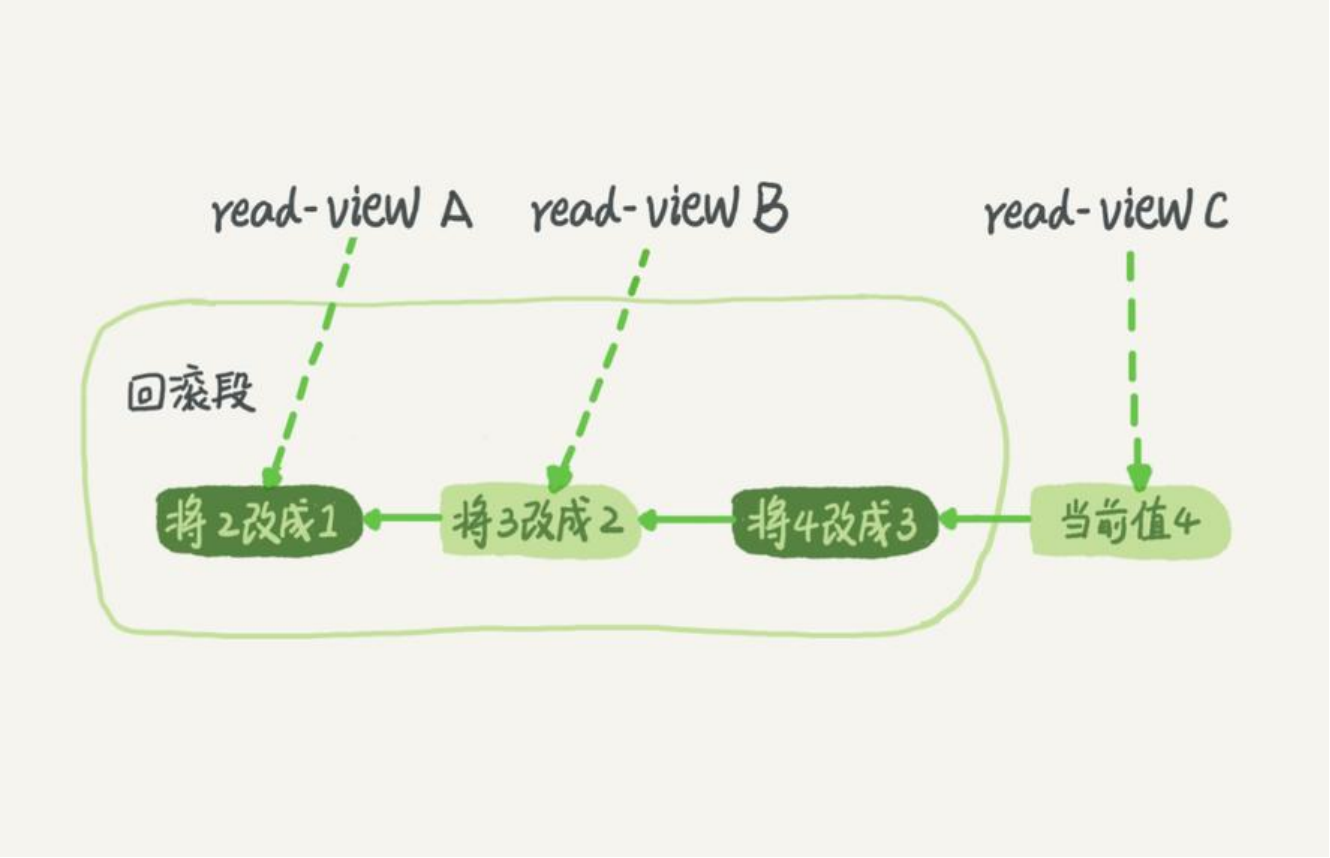
InnoDB默认的隔离级别是可重复读。通过SET　TRANSACTION ISOLATION LEVEL

总结来说，存在即合理，哪个隔离级别都有它自己的使用场景，要根据自己的业务情况来定。

事务隔离的实现

在MySQL中，实际上每条记录在更新的时候都会同时记录一条回滚操作。记录上的最新值，通过回操作，都可以得到前一个状态的值。这里我们展开说明“可重复读”。

假设一个值从1被按顺序改成了2、3、4，在回滚日志里面就会有类似下面的记录，这里的每次修改就一个事务。



由图看到，当前值是4，但是在查询这条记录的时候，不同时刻启动的事务会有不同的视图，对于read view A，要得到1，就必须将当前值依次执行图中所有的回滚操作得到。

回滚日志再不需要的时候会进行删除，系统会判断，当没有事务再需要用到这些回滚日志时，回滚日会被删除(就是当系统里没有比这个回滚日志更早的read-view的时候。)

上一篇：MySQL学习笔记-----日志模块

下一篇：MySQL学习笔记-----深入浅出索引