

MySQL事务之不可重复读问题

原创小异常于 2019-05-01 18:25:35 发布10716收藏32

分类专栏：数据库文章标签：mysql事务不可重复读数据库

版权

数据库专栏收录该内容

1 订阅20 篇文章

订阅专栏

版权声明：本文为 小异常 原创文章，非商用自由转载-保持署名-注明出处，谢谢！
本文网址：<https://blog.csdn.net/sun8112133/article/details/89739475>

文章目录

- 一、不可重复读概述
 - 1、事例
 - 2、分析
- 二、演示不可重复读
 - 1、新建一个数据库（bank库），并准备一张表（account表）
 - 2、打开两个窗口，并分别设置自动提交方式为off
 - 3、将A窗口的隔离级别设置成“读已提交”
 - 4、两个窗口分别开启事务
 - 5、在B窗口更改数据，并提交事务
 - 6、分别在数据库和A窗口中查看数据

在事务的并发操作中，也就是多个事务同时对同一组数据进行操作时，可能会出现脏读、不可重复读、幻读、丢失更新这四个问题，本篇博客就来为大家讲解不可重复读问题。

一、不可重复读 概述

不可重复读 就是一个事务读到另一个事务修改后并提交的数据（update）。在同一个事务中，对于同一组数据读取到的结果不一致。比如，事务B在事务A提交前读到的结果，和在事务A提交后读到的结果可能不同。不可重复读出现的原因就是由于事务并发修改记录而导致的。

隔离级别 有四种，分别是：读未提交、读已提交、可重复读、序列化。

读未提交：Read Uncommitted，顾名思义，就是一个事务可以读取另一个未提交事务的数据。最低级别，它存在4个常见问题（脏读、不可重复读、幻读、丢失更新）。

读已提交：Read Committed，顾名思义，就是一个事务要等另一个事务提交后才能读取数据。它解决了脏读问题，存在3个常见问题（不可重复读、幻读、丢失更新）。

可重复读：Repeatable Read，就是在开始读取数据（事务开启）时，不再允许修改操作。它解决了脏读和不可重复读，还存在2个常见问题（幻读、丢失更新）。

序列化：Serializable，序列化，或串行化。就是将每个事务按一定的顺序去执行，它将隔离问题全部解决，但是这种事务隔离级别效率低下，比较耗数据库性能，一般不使用。

小异常

关注

1

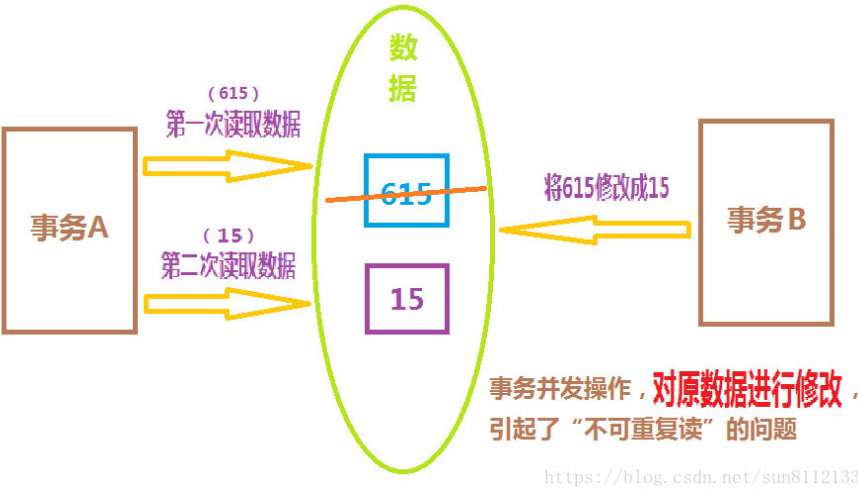
大多数数据库默认的事务隔离级别是 Read Committed，比如 SQL Server，Oracle。但MySQL 的默认隔离级别是 Repeatable Read。

1、事例

程序员拿着信用卡去享受生活（卡里当然是只有3.6万），当他买单时（程序员事务开启），收费系统事先检测到他的卡里有3.6万，就在这个时候！！程序员的妻子要把钱全部转出充当家用，并提交。当收费系统准备扣款时，再检测卡里的金额，发现已经没钱了（第二次检测金额当然要等待妻子转出金额事务并提交完）。程序员就会很郁闷，明明卡里是有钱的...

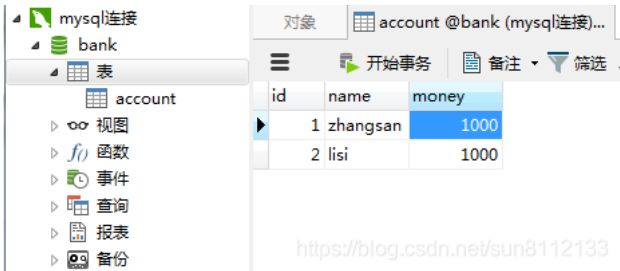
2、分析

在这个事例中，涉及到了两个事务（程序员事务和妻子事务），当程序员事务开启时，收费系统读取程序员卡里钱的操作还没完成，此时妻子这个事务就将卡里的钱进行了转账，即对数据进行了修改，导致收费系统两次读取到的数据不一样。出现了一个事务范围内两个相同的查询却返回了不同数据，这就是不可重复读，这是由于数据更新导致的，不能重复读取相同的数据。



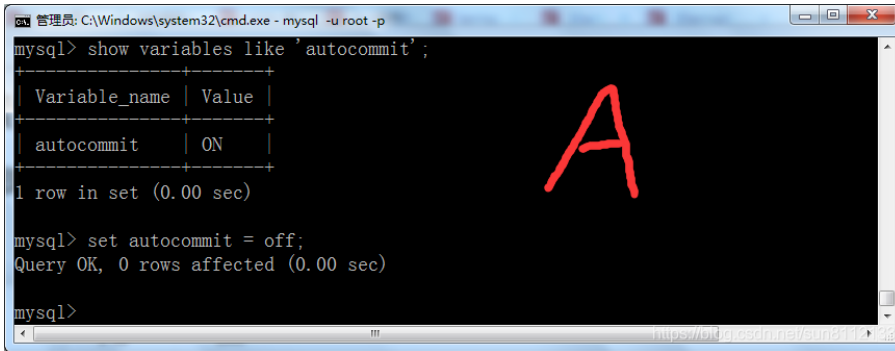
二、演示不可重复读

1、新建一个数据库（bank库），并准备一张表（account表）



2、打开两个窗口，并分别设置自动提交方式为off

```
show variables like 'autocommit'; -- 查看当前的自动提交是否开启
set autocommit = off; -- 将自动提交关闭
```



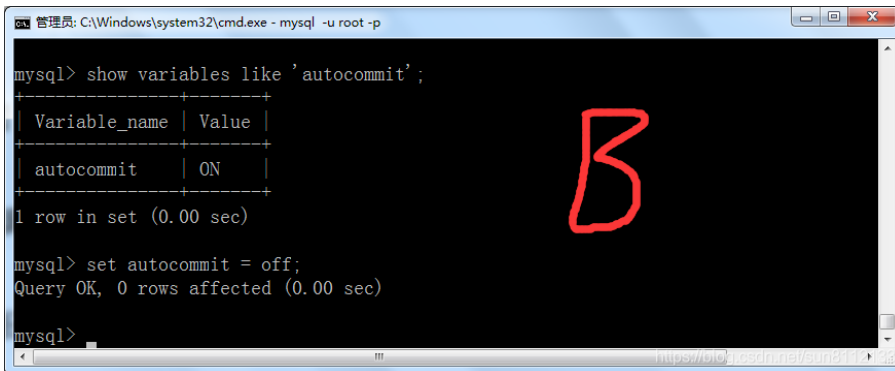
```
mysql> show variables like 'autocommit';
```

Variable_name	Value
autocommit	ON

```
mysql> set autocommit = off;
```

Query OK, 0 rows affected (0.00 sec)

mysql>



```
mysql> show variables like 'autocommit';
```

Variable_name	Value
autocommit	ON

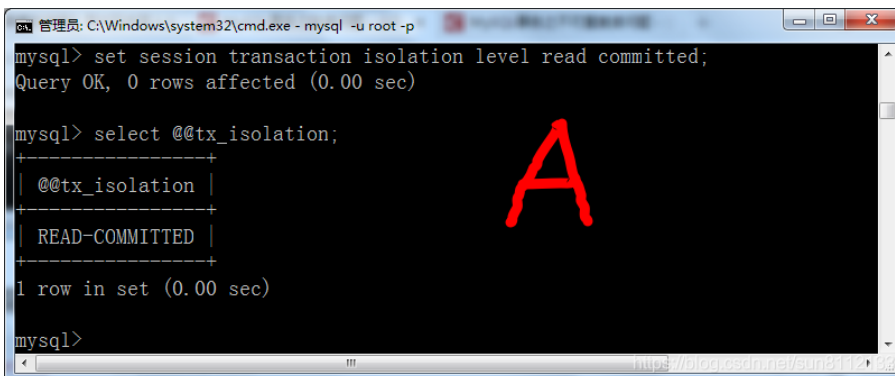
```
mysql> set autocommit = off;
```

Query OK, 0 rows affected (0.00 sec)

mysql>

3、将A窗口的隔离级别设置成“读已提交”

`select @@tx_isolation;` – 查询当前的隔离级别
`set session transaction isolation level read committed;` – 设置当前会话隔离级别为“读已提交”



```
mysql> set session transaction isolation level read committed;
```

Query OK, 0 rows affected (0.00 sec)

```
mysql> select @@tx_isolation;
```

@@tx_isolation
READ-COMMITTED

```
mysql>
```

4、两个窗口分别开启事务

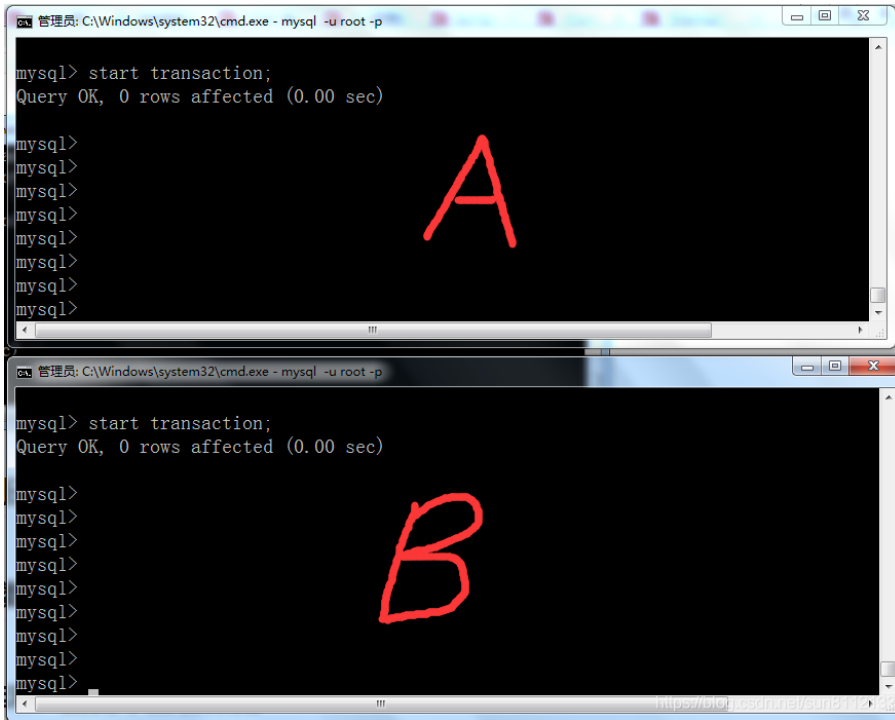
`start transaction;` – 开启事务 或 `begin;` 也可以显式开启事务



小异常

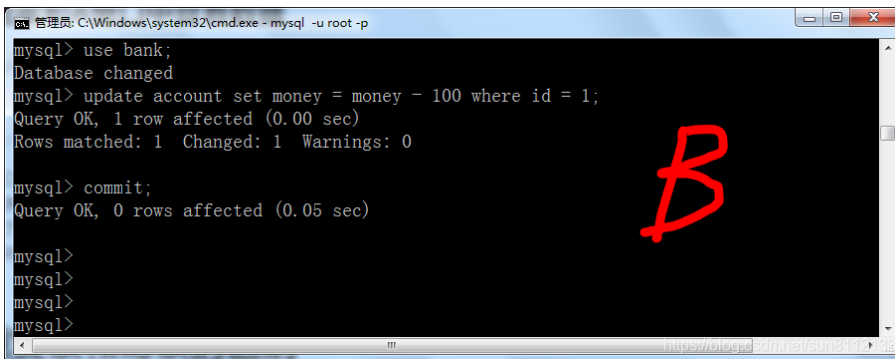
关注

1



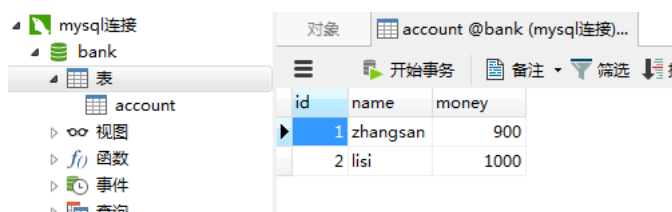
5、在B窗口更改数据，并提交事务

```
use bank      -- 切换到bank数据库
update account set money = money - 100 where id = 1;    -- 修改account表中
id为1的money字段数据
commit;        -- 提交事务
```



6、分别在数据库和A窗口中查看数据

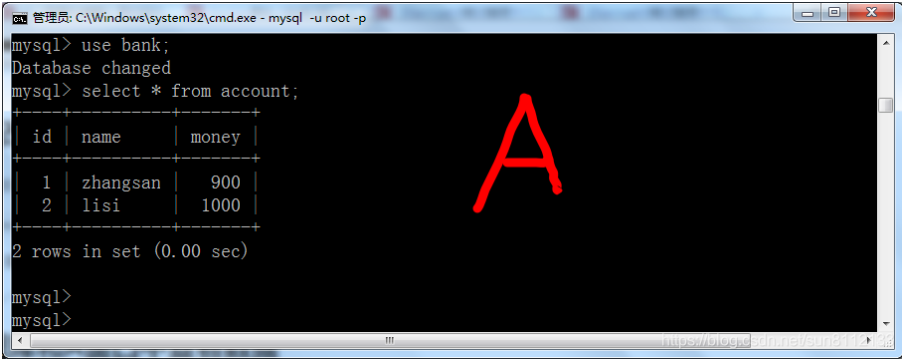
```
select * from account;    -- 查看account中的所有数据
```



小异常

关注

1



大家会发现数据库和A窗口中的数据都发生了改变，因为B窗口已经提交了事务，所以数据库中的数据发生改变，是属于正常现象。但是这种事务的隔离性似乎不是太好（事务的隔离性是一个事务的执行，不受其他事务的干扰），你看，B窗口提交了事务，影响到了A窗口中数据，这种隔离级别虽然解决了“脏读”问题，但是还会引发“不可重复读”、“幻读”及“丢失更新”问题，有关“幻读”及“丢失更新”的问题请参考后续博客。

有关事务的知识可以参考我之前写的博客《【Spring4.0笔记整理十七】Spring事务详解》及【Spring4.0笔记整理十八】Spring事务管理详解。

博客中若有不恰当的地方，请您一定要告诉我。前路崎岖，望我们可以互相帮助，并肩前行！

mysql不可重复读和重复读_mysql怎么解决不可重复读

weixin_35453718的博客 196

我就废话不多说了，大家还是直接看代码吧~create or replace function aa1(a1 integer[],a2 bigint) returns vo...

不可重复读有什么影响_MySQL可重复读的实际意义

weixin_39615808的博客 703

1.可重复读的疑惑事务的隔离级别定义了数据库系统中一个事务中操作的结果在何时以何种方式对其他并发...

评论 5

请发表有价值的评论，博客评论不欢迎灌水，良好的社区氛围需大家一起维护

评论

BINGINSL 2021.04.24

可重复读还会有丢失更新的问题吗？不会了吧

悦诗风吟 2021.02.21

错了，我设置有问题

悦诗风吟 2021.02.21

但是在mysql 8.0下你这种情况会发现事务A并没有读取到事务B将账户余额减去了100变为900并已提交的记录、所以我觉得神奇了，是不是mysql 8.0已经将innodb下读已提交和可重复读在同一事务里的相同条件读取都变为快照读而非当前读

悦诗风吟 回复 小异常 2021.02.23

对的，你的没问题，我自己搞错了，8.0依旧出现

查看全部 2 条回复

MySQL的不可重复读和幻读_AilsAlive的博客

2-18

如果Bob重复读取post_id为1的数据,他将发现结果集的记录数量发生了变化;幻读的重点在于新增或者删除 ...

mysql幻读和不可重复读的区别_不可重复读和幻读的区别...

3-5

所以说不可重复读和幻读最大的区别,就在于如何通过锁机制来解决他们产生的问题。上文说的,是使用悲观...

MySQL事务的不可重复读

天高任鸟飞 4202

不可重复读——NON-REPEATABLE READ 是指事务中，两次查

小异常

关注

1