

# 第43期-事务基础知识

2019年7月5日 13:35

## 事务的概念

事务 (Transaction) 是由一系列对系统中数据进行访问与更新的操作所组成的一个程序执行逻辑单元。

## 事务的特征

事务具有4个基本特征，分别是：原子性 (Atomicity)、一致性 (Consistency)、隔离性 (Isolation)、持久性 (Duration)，简称：ACID，面试经常问到。

## 事务的隔离级别

ACID这4个特征中，最难理解的是隔离性。在标准SQL规范中，定义了4个事务隔离级别，不同的隔离级别对事务的处理不同。4个隔离级别分别是：

- 读未提交 (READ\_UNCOMMITTED)
- 读已提交 (READ\_COMMITTED)
- 可重复读 (REPEATABLE\_READ)
- 顺序读 (SERIALIZABLE)

## 事务所引发的问题

数据库在不同的隔离性级别下，并发访问可能会出现以下几种问题：

- 脏读 (Dirty Read)
- 不可重复读 (Unrepeatable Read)
- 幻读 (Phantom Read)

## 事务的思维导图



## 事务的ACID详解

### 原子性 (Atomicity)

原子：化学反应不可再分的基本微粒，原子在化学反应中不可分割，原子构成一般物质的最小单位。

事务的原子性是指事务必须是一个原子的操作序列单元。事务中包含的各项操作在一次执行过程中，只允许出现两种状态之一。

- 全部执行成功
- 全部执行失败

任何一项操作都会导致整个事务的失败，同时其它已经被执行的操作都将被撤销并回滚，只有所有的操作全部成功，整个事务才算是成功完成。

### 一致性 (Consistency)

事务的一致性是指事务的执行不能破坏数据库数据的完整性和一致性，一个事务在执行之前和执行之后，数据库都必须处以一致性状态。

比如：如果从A账户转账到B账户，不可能因为A账户扣了钱，而B账户没有加钱，无论A和B怎么转账，系统中总额是固定的，不可能因为A和B转账导致系统总额缺斤少两。

## 隔离性 (Isolation)

事务的隔离性是指在并发环境中，并发的事务是互相隔离的，一个事务的执行不能被其它事务干扰。也就是说，不同的事务并发操作相同的数据时，每个事务都有各自完整的数据空间。

一个事务内部的操作及使用的数据对其它并发事务是隔离的，并发执行的各个事务是不能互相干扰的。隔离性分4个级别，下面会介绍。

### [IsolationLevel Enum](#)

## 持久性 (Duration)

事务的持久性是指事务一旦提交后，数据库中的数据必须被永久的保存下来。即使服务器系统崩溃或服务器宕机等故障。只要数据库重新启动，那么一定能够将其恢复到事务成功结束后的状态。

## 事务隔离级别详解

### 1. 读未提交 (READ\_UNCOMMITTED)

读未提交，该隔离级别允许脏读取，其隔离级别是最低的。换句话说，如果一个事务正在处理某一数据，并对其进行了更新，但同时尚未完成事务，因此还没有提交事务；而以此同时，允许另一个事务也能够访问该数据。

#### 脏读示例：

在事务A和事务B同时执行时可能会出现如下场景：

时间	事务A (存款)	事务B (取款)
T1	开始事务	——
T2	——	开始事务
T3	——	查询余额 (1000元)
T4	——	取出1000元 (余额0元)
T5	查询余额 (0元)	——
T6	——	撤销事务 (余额恢复1000元)
T7	存入500元 (余额500元)	——
T8	提交事务	——

余额应该为1500元才对。请看T5时间点，事务A此时查询的余额为0，这个数据就是**脏数据**，他是

事务B造成的，很明显是事务没有进行隔离造成的。

## 2. 读已提交 (READ\_COMMITTED)

读已提交是另外事务执行的时候只能获取到已经提交的数据。这样就不会出现上面的脏读。

### 不可重复读示例

可是解决了脏读问题，但是还是解决不了可重复读问题。

时间	事务A (存款)	事务B (取款)
T1	开始事务	——
T2	——	开始事务
T3	——	查询余额 (1000元)
T4	查询余额 (1000元)	——
T5	——	取出1000元 (余额0元)
T6	——	提交事务
T7	查询余额 (0元)	——
T8	提交事务	——

事务A其实除了查询两次以外，其它什么事情都没做，结果钱就从1000变成0了，这就是不可重复读的问题。

## 3. 可重复读 (REPEATABLE\_READ)

可重复读就是保证在事务处理过程中，多次读取同一个数据时，该数据的值和事务开始时刻是一致的。因此该事务级别解决了重复读取和脏读，但是有可能出现幻读的数据。

### 幻读

幻读就是指同样的事务操作，在前后两个时间段内执行对同一个数据项的读取，可能出现不一致的结果集。

时间	事务A (统计总存款)	事务B (存款)
T1	开始事务	——
T2	——	开始事务
T3	统计总存款 (1000元)	——
T4	——	存入100元

T5	——	提交事务
T6	提交总存款（1100）	——
T7	提交事务	——

银行工作人员在一个事务中多次统计总存款时看到结果不一样。如果要解决幻读，那只能使用顺序读了。

## 4. 顺序读（SERIALIZABLE）

顺序读是最严格的事务隔离级别。它要求所有的事务排队顺序执行，即事务只能一个接一个地处理，不能并发。

### 事务隔离级别对比

事务隔离级别	脏 读	不可重复读	幻 读
读未提交（READ_UNCOMMITTED）	允许	允许	允许
读已提交（READ_COMMITTED）	禁止	允许	允许
可重复读（REPEATABLE_READ）	禁止	禁止	允许
顺序读（SERIALIZABLE）	禁止	禁止	禁止

4种事务隔离级别从上往下，级别越高，并发性越差，安全性就越来越高。一般数据默认级别是读已提交或可重复读。

### 相关附加阅读（重要）

什么是悲观锁和乐观锁

<https://zhuanlan.zhihu.com/p/31537871>

何为脏读、不可重复读和幻读：

<https://www.jianshu.com/p/f7ac1b22e899>

数据库事务详解

<https://www.jianshu.com/p/aa35c8703d61>

脏读、不可重复读、重复读和幻读的区别：

<https://www.cnblogs.com/tudou1223/p/7954432.html>

深入事务管理内部原理：

<https://www.cnblogs.com/artech/tag/Transaction>

SqlServer锁机制与实践

<https://www.cnblogs.com/wangweitr/p/7158023.html>

SQL Server 中锁与事务隔离级别

<https://www.jianshu.com/p/9233e5f12388>