

数据库并发和并发异常

刘倩

白俄罗斯国立大学

Treebush302@gamail.com

摘要 (annotation) : 在使用数据库来支撑业务系统时, 随着用户量的增大, 经常会遇到同时读取相同数据的情况, 在没有对数据进行并发控制的情况下就会遇到各种各样的问题, 本文章对上述问题讲述了几种解决方法。

有两个方案可以解决读一致性问题: 基于锁的并发操作(LBCC)和基于多本版本的并发操作 (MVCC) 。

LBCC

既然需要保证前后两次读取数据一致, 那么在读取数据的时候, 给数据加个行锁, 锁定我要操作的数据, 不允许其他的事务来修改, 这种方案叫做基于锁的并发控制 (LBCC) 。

LBCC 是通过悲观锁来实现并发控制的, 如果事务 A 对数据进行加锁, 那么在锁被释放前, 其他事务就不可以对数据进行读写操作, 基于这种情况, 其实对大多数系统性能是不能满足要求的, 因为读操作是不会改变数据, 我们应该允许用户对数据进行读操作, 做到读写分离, 所以下面是第二种方法介绍关于数据库并发异常的解决方案。

MVCC

要让一个事务前后读取的数据保持一致, 我们可以在修改数据的时候给它建立一个备份, 后面来读取这个备份就可以了, 无论事务执行的时间有多长, 事务内部看到的数据是不受其他事务影响的, 根据事务开始时间不同, 同一时刻看到的数据是不一样的, 这种方案我们称之为多版本的控制方案 (MVCC) , MVCC 是基于乐观锁的。

LBCC 和 MVCC 都可以解决数据并发的问題, 具体使用哪种方式, 要结合具体的业务场景来选择, 比如涉及到交易金额, 像银行存款我们可以使用悲观锁 LBCC 来对数据进行并发控制, 像有一些非敏感信息且用户经常需要查看的我们可以使用 MVCC 来进行数据的并发的控制操作, 这样可以提升系统性能, 当然 MVCC 和锁也可以结合使用, 没有最好的保持数据并发异常的解决方案, 只有最合适的方式。