

【VIP直播课】

Spring事务传播原理及数据库事务操作原理

Tom





咕泡学院-Tom
前中电在线技术总监
前超星网架构师
现为咕泡学院联合创始人

10余年Java经验。精通java语言。开发过多套企业UI框架、ORM框架。热衷于分享经验，共同进步。

不只做一个技术者，更要做一个思考者。





书法爱好者、绘画爱好者 编程界字写得最好的 书法界编程最牛逼的

自幼开始练习书法。中学期间，曾获市级青少年杯书法竞赛一等奖，获校园杯美术竞赛工笔画一等奖，获校园征文比赛二等奖。大学担任学生会宣传部长，负责校园黑板报、校园刊物的编辑排版设计。参加工作后，担任过家具建模、平面设计等工作。

亲自设计咕泡学院Logo。



扫码加入书法兴趣小组



群名称:Tom老师书法兴趣小组
群 号:549209007

有兴趣的可以扫码加入书法兴趣小组



- 1、掌握Spring事务传播原理。
- 2、掌握基于Spring的数据库事务操作原理。
- 3、初步了解分布式事务。



- 1、适合有Spring开发经验并且希望深度掌握事务控制原理的人群。
- 2、通过本章内容的学习，能够掌握基于Spring AOP的事务设计原理。





从Spring事务配置说起

专注互联网IT教育，做技术人的指路明灯，职场生涯的精神导师。
咕泡学院官网：<http://www.gupaoedu.com>



专业互联网
IT教育服务平台

```
<tx:advice id="transactionAdvice" transaction-manager="transactionManager">
    <tx:attributes>
        <tx:method name="add*" propagation="REQUIRED" rollback-for="Exception,RuntimeException,SQLException"/>
        <tx:method name="remove*" propagation="REQUIRED" rollback-for="Exception,RuntimeException,SQLException"/>
        <tx:method name="modify*" propagation="REQUIRED" rollback-for="Exception,RuntimeException,SQLException"/>
        <tx:method name="login" propagation="NOT_SUPPORTED"/>
        <tx:method name="query*" read-only="true"/>
    </tx:attributes>
</tx:advice>

<aop:config>
    <aop:pointcut expression="execution(public * com.gupaoedu.vip..*.service..*Service.*(..))" id="transactionPointcut"/>
    <aop:advisor pointcut-ref="transactionPointcut" advice-ref="transactionAdvice"/>
</aop:config>
```





数据库事务原理详解



事务(Transaction)是访问并可能更新数据库中各种数据项的一个程序执行单元(unit)。

**特点：事务是恢复和并发控制的基本单位。事务应该具有4个属性：
原子性、
一致性、
隔离性、
持久性。
这四个属性通常称为ACID特性。**



常量名称	常量解释
PROPAGATION_REQUIRED	支持当前事务，如果当前没有事务，就新建一个事务。这是最常见的选择，也是 Spring 默认的事务的传播。
PROPAGATION_REQUIRES_NEW	新建事务，如果当前存在事务，把当前事务挂起。新建的事务将和被挂起的事务没有任何关系，是两个独立的事务，外层事务失败回滚之后，不能回滚内层事务执行的结果，内层事务失败抛出异常，外层事务捕获，也可以不处理回滚操作。
PROPAGATION_SUPPORTS	支持当前事务，如果当前没有事务，就以非事务方式执行。
PROPAGATION_MANDATORY	支持当前事务，如果当前没有事务，就抛出异常。
PROPAGATION_NOT_SUPPORTED	以非事务方式执行操作，如果当前存在事务，就把当前事务挂起。
PROPAGATION_NEVER	以非事务方式执行，如果当前存在事务，则抛出异常。
PROPAGATION_NESTED	如果一个活动的事务存在，则运行在一个嵌套的事务中。如果没有活动事务，则按REQUIRED属性执行。它使用了一个单独的事务，这个事务拥有多个可以回滚的保存点。内部事务的回滚不会对外部事务造成影响。它只对DataSourceTransactionManager事务管理器起效。



隔离级别	隔离级别的值	导致的问题
Read-Uncommitted	0	导致脏读
Read-Committed	1	避免脏读，允许不可重复读和幻读
Repeatable-Read	2	避免脏读，不可重复读，允许幻读
Serializable	3	串行化读，事务只能一个一个执行，避免了脏读、不可重复读、幻读。执行效率慢，使用时慎重



脏读：一事务对数据进行了增删改，但未提交，另一事务可以读取到未提交的数据。如果第一个事务这时候回滚了，那么第二个事务就读到了脏数据。

不可重复读：一个事务中发生了两次读操作，第一次读操作和第二次操作之间，另外一个事务对数据进行了修改，这时候两次读取的数据是不一致的。

幻读：第一个事务对一定范围的数据进行批量修改，第二个事务在这个范围增加一条数据，这时候第一个事务就会丢失对新增数据的修改。



隔离级别

解释

ISOLATION_DEFAULT

这是个 PlatformTransactionManager 默认的隔离级别，使用数据库默认的事务隔离级别。另外四个与 JDBC 的隔离级别相对应。

ISOLATION_READ_UNCOMMITTED

这是事务最低的隔离级别，它允许另外一个事务可以看到这个事务未提交的数据。这种隔离级别会产生脏读，不可重复读和幻像读。

ISOLATION_READ_COMMITTED

保证一个事务修改的数据提交后才能被另外一个事务读取。另外一个事务不能读取该事务未提交的数据。

ISOLATION_REPEATABLE_READ

这种事务隔离级别可以防止脏读，不可重复读。但是可能出现幻像读。

ISOLATION_SERIALIZABLE

这是花费最高代价但是最可靠的事务隔离级别。事务被处理为顺序执行。





浅谈分布式事务



Consistency 一致性

Availability 可用性

Partition Tolerance 分区容错性

DTS

通过日志还原，显然滞后，变成了异步操作



谢谢观看

Tom



Tom老师QQ号: 441221062

