

【第42话：事务隔离级别和事务传播行为这块应该怎么回答】

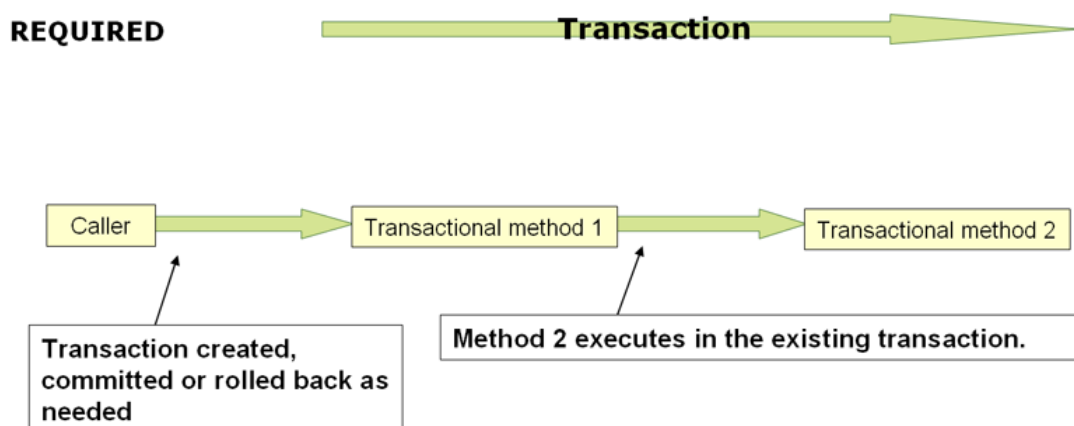
Hello 小伙伴们，这节课给大家说一下：“请说一下Spring如何配置事务隔离级别和事务传播行为”。

我们先来看看事务传播行为。

事务传播行为：当出现service的方法调用另一个service方法时（这些方法都被声明式事务管理），这些方法如何进行事务管理。

可以通过进行配置tx:method或@Transactional中的propagation属性来进行设置,propagation属性的可选值有：

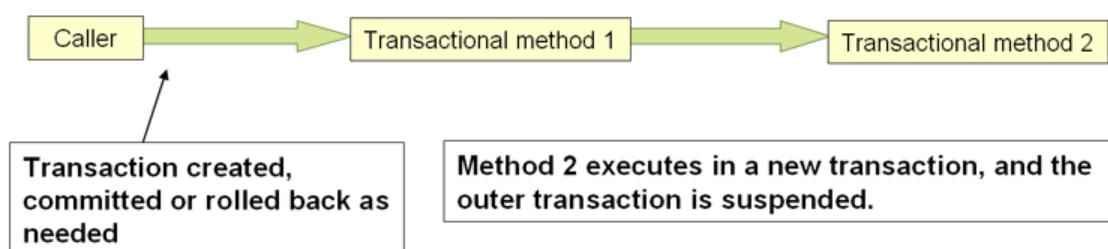
REQUIRED:默认值。如果当前有事务则加入到事务中。如果当前没有事务则新增事务。



NEVER:必须在非事务状态下执行,如果当前没有事务,正常执行,如果当前有事务,报错.

NESTED:必须在事务状态下执行.如果没有事务,新建事务,如果当前有事务,创建一个嵌套事务.

REQUIRES_NEW: 必须在事务中执行,如果当前没有事务,新建事务,如果当前有事务,把当前事务挂起.在重新建个事务。（调用者统一提交回滚）



SUPPORTS: 如果当前有事务就在事务中执行,如果当前没有事务,就在非事务状态下执行.

NOT_SUPPORTED: 必须在非事务下执行,如果当前没有事务,正常执行,如果当前有事务,把当前事务挂起.

MANDATORY: 必须在事务内部执行,如果当前有事务,就在事务中执行,如果没有事务,报错.（可以配置在入口方法）

然后在来看看事务隔离级别

多个事务同时操作数据库时，允许多个事务操作的方式就是事务隔离级别。事务隔离级别主要是通过添加锁操作实现的。事务隔离级别主要是解决高并发下脏读、幻读、不可重复读问题的。

事务隔离级别出现场景：高并发场景。

Java主要做的就是Web项目（服务端项目），每次客户端过来的都叫请求。运行多人同时请求。每个请求运行同一个方法实际上是多个事务。

脏读：

事务A没有提交事务，事务B读取到事务A未提交的数据，这个过程称为脏读。读取到的数据叫做脏数据。

不可重复读：

当事务A读取到表中一行数据时，同时另一个事务修改这行数据，事务A读取到的数据和表中真实数据不一致。

幻读：

事务A对表做查询全部操作，事务B向表中新增一条数据。事务A查询出来的数据和表中数据不一致，称为幻读。

我们可以在tx:method或@Transactional中设置属性isolation的值来进行配置。其中isolation可取值分别为 **DEFAULT:**

表示用数据库的隔离级别，MySQL8默认的事务隔离级别REPEATABLE_READ
(select @@transaction_isolation)

READ_UNCOMMITTED:

读未提交(脏读，幻读，不可重复读)

READ_COMMITTED:

读已提交(幻读，不可重复读)

REPEATABLE_READ:

可重复读（幻读）

SERIALIZABLE

串行读来通过牺牲性能解决脏读、不可重复度、幻读问题。