在Spring中数据库事务是通过PlatformTransactionManager进行管理，常用的实现类是DataSourceTransactionManager

在Spring中事务可以分为**声明式事务**或者**编程式事务**，常用的是声明式事务，声明式事务又分为**XML配置**和**注解配置**，主要是注解配置，使用@Transaction注解。

**配置事务管理器**

XML配置

定义数据库连接池（DBCP），使用DatasourceTransactionManager去定义事务管理器，并且注入数据库连接池。

要在配置文件中加入如下配置：

<tx:annotation-driven transaction-manager= ” transactionManager ” />

Java配置

用Java配置的方式来实现Spring数据库事务，需要在配置类中实现接口TransactionManagerConfigurer的AnnotationDrivenTransactionManager方法，**该方法返回的事务管理器作为程序中的事务管理器。**

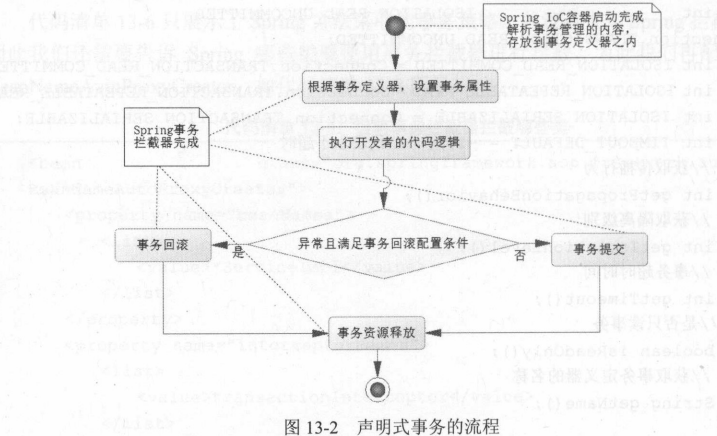
注意：要在实现类上加@EnableTransactionManagement，加上这个注解之后，就可以使用@Transaction注解

**使用事务管理器**

使用事务管理器可以分为XML配置和注解配置，Spring IoC容器初始化的时候，会读入这个注解或者XML的配置信息，并且保存到一个事务定义类中。当运行时Spring拦截XML的配置信息标注的类或者注解标注的某一个方法或者类的全部方法。

实现原理： AOP

1. XML配置需要一个事务拦截器Transactionincepter, 并且需要配置一个BeanNameAutoProxyCreator告诉Spring需要拦截哪些类
2. 注解配置直接使用@transaction 应用在**方法**或**类**



**丢失更新：**

第一类丢失更新：一个回滚、一个提交 数据库自身基本消除

第二类丢失更新：两个事务均提交

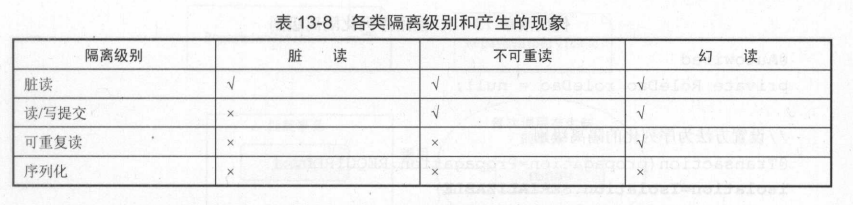
**隔离级别：**

脏读： 允许一个事务去读取另一个事务未提交的数据

读/写提交： 一个事务只能读取另一个事务已经提交的数据

可重复读（针对同一条数据库记录）

序列化 （针对不用数据库记录）



**传播行为：**

一个方法调度另一个方法，可以对事务的特性进行传播配置，称为传播行为

