

Java中数据库事务处理的实现 PDF转换可能丢失图片或格式，
建议阅读原文

https://www.100test.com/kao_ti2020/252/2021_2022_Java_E4_B8_AD_E6_95_B0_c104_252450.htm JavaBean JavaBean是用Java语言编写的与平台无关的组件。它是描述Java的软件组件模型，有点类似于Microsoft的COM组件的概念。在Java模型中，通过JavaBean可以无限扩充Java程序的功能，通过JavaBean的组合可以快速的生成新的应用程序。JavaBean可以实现代码的重复利用，对于程序的易维护性也有重大的意义。非可视化的JavaBean，在JSP程序中常用来封装事务逻辑、数据库操作等，可以很好的实现业务逻辑和前台程序的分离。JavaBean在服务器端的应用方面表现出了越来越强的生命力。EJB EJB技术定义了一组可重用的组件：Enterprise JavaBeans。你可以利用这些组件，像搭积木一样的建立你的分布式应用程序。当你把代码写好之后，这些组件就被组合到特定的文件中。每个文件有一个或多个Enterprise Beans，在加上一些配置参数。最后，这些Enterprise Beans被配置到一个装了EJB容器的平台上。客户能够通过这些Beans的home接口，定位到某个beans，并产生这个beans的一个实例。这样，客户就能够调用Beans的应用方法和远程接口。EJB技术简化了用Java语言编写的企业应用系统的开发、配置和执行。有三种类型的Enterprise beans: session beans、entity beans和Message-driven Beans。事务处理 信息是任何企事业单位的重要资产，任何企业部门都包含着信息的流入、流出，任何企业部门都控制着某些信息。同时，信息必须在适当的时机传播给需要的人。而且，信息还需要安全约束，通常根据信息的类型和内容

实施访问控制。为了保证数据的安全有效和正确可靠，数据库管理系统（DBMS）必须提供统一的数据保护功能。事务是现代数据库理论中的核心概念之一。如果一组处理步骤或者全部发生或者一步也不执行，我们称该组处理步骤为一个事务。当所有的步骤像一个操作一样被完整地执行，我们称该事务被提交。由于其中的一部分或多步执行失败，导致没有步骤被提交，则事务必须回滚（回到最初的系统状态）。事务必须服从ISO/IEC所制定的ACID原则。ACID是原子性（atomicity）、一致性（consistency）、隔离性（isolation）和持久性（durability）的缩写。事务的原子性表示事务执行过程中的任何失败都将导致事务所做的任何修改失效。一致性表示当事务执行失败时，所有被该事务影响的数据都应该恢复到事务执行前的状态。隔离性表示在事务执行过程中对数据的修改，在事务提交之前对其他事务不可见。持久性表示已提交的数据在事务执行失败时，数据的状态都应该正确。在下面我们列举一个使用SQL Server数据库进行事务处理的例子。主表是一个规章制度信息表（bylaw），主要字段有记录编号、标题、作者、书写日期等。两个子表分别是附件表（bylaw_affix）和文本信息表（bylaw_content）。表结构见图1所示。bylaw表的记录编号与bylaw_affix表的记录编号、bylaw_content表的记录编号是对应的，每次对规章制度信息的操作也就是对这三个表的联合操作。例如要删除规章制度中的一条记录，如果不使用事务，就可能会出现这样的情况：第一个表中成功删除后，数据库突然出现意外状况，而第二、三个表中的操作没有完成，这样，删除操作并没有完成，甚至已经破坏数据库中的数据。要避免这种情况，就应

该使用事务，它的作用是：要么三个表都操作成功，要么都失败。换句话说，就是保持数据的一致性。所以，为了确保对数据操作的完整和一致，在程序设计时要充分考虑到事务处理方面的问题。图1 示例表结构 Java中的事务处理 一般情况下，J2EE应用服务器支持JDBC事务、JTA（Java Transaction API）事务、容器管理事务。一般情况下，最好不要在程序中同时使用上述三种事务类型，比如在JTA事务中嵌套JDBC事务。第二方面，事务要在尽可能短的时间内完成，不要在不同方法中实现事务的使用。下面我们列举两种事务处理方式。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com