**事务的：原子性、一致性、分离性、持久性**

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | |  | | --- | | 原子性、一致性、分离性、持久性 **(1) 原子性**  事务的原子性指的是，事务中包含的程序作为数据库的逻辑工作单位，它所做的对数据修改操作要么全部执行，要么完全不执行。这种特性称为原子性。  事务的原子性要求，如果把一个事务可看作是一个程序，它要么完整的被执行，要么完全不执行。就是说事务的操纵序列或者完全应用到数据库或者完全不影响数据库。这种特性称为原子性。 　 假如用户在一个事务内完成了对数据库的更新，这时所有的更新对外部世界必须是可见的，或者完全没有更新。前者称事务已提交，后者称事务撤消（或流产）。DBMS必须确保由成功提交的事务完成的所有操纵在数据库内有完全的反映，而失败的事务对数据库完全没有影响。 **(2) 一致性**     事务的一致性指的是在一个事务执行之前和执行之后数据库都必须处于一致性状态。这种特性称为事务的一致性。假如数据库的状态满足所有的完整性约束，就说该数据库是一致的。 一致性处理数据库中对所有语义约束的保护。假如数据库的状态满足所有的完整性约束，就说该数据库是一致的。例如，当数据库处于一致性状态S1时，对数据库执行一个事务，在事务执行期间假定数据库的状态是不一致的，当事务执行结束时，数据库处在一致性状态S2。 **(3) 分离性** 分离性指并发的事务是相互隔离的。即一个事务内部的操作及正在操作的数据必须封锁起来，不被其它企图进行修改的事务看到。 　　分离性是DBMS针对并发事务间的冲突提供的安全保证。DBMS可以通过加锁在并发执行的事务间提供不同级别的分离。假如并发交叉执行的事务没有任何控制，操纵相同的共享对象的多个并发事务的执行可能引起异常情况。 DBMS可以在并发执行的事务间提供不同级别的分离。分离的级别和并发事务的吞吐量之间存在反比关系。较多事务的可分离性可能会带来较高的冲突和较多的事务流产。流产的事务要消耗资源，这些资源必须要重新被访问。因此，确保高分离级别的DBMS需要更多的开销。 **(4)持久性** 持久性意味着当系统或介质发生故障时，确保已提交事务的更新不能丢失。即一旦一个事务提交，DBMS保证它对数据库中数据的改变应该是永久性的，耐得住任何系统故障。持久性通过数据库备份和恢复来保证。 持久性意味着当系统或介质发生故障时，确保已提交事务的更新不能丢失。即对已提交事务的更新能恢复。一旦一个事务被提交，DBMS必须保证提供适当的冗余，使其耐得住系统的故障。所以，持久性主要在于DBMS的恢复性能。 | | |