

考试技术SQLServer和Oracle防止数据锁定的比较 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/143/2021_2022__E8_80_83_E8_AF_95_E6_8A_80_E6_c102_143173.htm 数据库并行访问，也就是两个或两以上用户同时访问同一数据，这也是数据库引擎如何设计和实现适度反应所面临的最大问题。设计优良、性能卓越的数据库引擎可以轻松地同时为成千上万的用户服务。而“底气不足”的数据库系统随着更多的用户同时访问系统将大大降低其性能。最糟糕的情况下甚至可能导致系统的崩溃。当然，并行访问是任何数据库解决方案都最为重视的问题了，为了解决并行访问方面的问题各类数据库系统提出了各种各样的方案。SQL Server和Oracle两大DBMS也分别采用了不同的并行处理方法。它们之间的实质差别在哪里呢？

并行访问的问题 并行访问出现问题存在若干种情况。在最简单的情形下，数量超过一个的用户可能同时查询同一数据。就这种情况而言数据库的操作目标很简单：尽可能地为用户们提供快速的数据访问。这对我们现在常见的数据库来说不成问题：SQL Server和 Oracle 都采用了多线程机制，它们当然能够一次处理多个请求。不过，在用户修改数据的情况下并行访问问题就变得复杂起来了。显然，数据库通常只允许唯一用户一次修改特定的数据。当某一用户开始修改某块数据时，SQL Server和 Oracle 都能很快地锁定数据，阻止其他用户对这块数据进行更新，直到修改该数据的第1位用户完成其操作并提交交易（commit transaction）。但是，当某一位用户正在修改某块数据时假设另一位用户又正想查询该数据的信息时会发生什么情况呢？在这种情况下数据库管理系统又该

如何动作呢？Oracle 和 SQL Server针对这一问题采取了不同的解决方案。SQL Server方法 现在不妨假设有人开始修改SQL Server上存储的数据，于是这块数据立即被数据库锁定。数据锁定操作阻塞其他任何访问该数据的连接连查询操作都不会放过。于是，这块被锁定的数据只有在交易被提交或者回滚之后才能接受其他访问操作。下面用SQL Server随带的pubs示例数据库做一个简单示范。在Query Analyzer内打开两个窗口。在第1个窗口中执行下列SQL操作语句，更新pubs数据库中某一图书的价格：`use pubs go begin tran 0update titles set price = price * 1.05 where title_id = BU2075` 由于代码中并没有执行commit语句，所以数据变动操作实际上还没有最终完成。接下来，在另一个窗口里执行下列语句查询titles数据表：`0select title_id,title,price from titles order by title_id.` 你什么结果也得不到。窗口底部的小地球图标会转个不停。尽管我在先前的操作中仅仅更新了一行，但是，0select语句的执行对象却恰好包含了其数据正被修改的一行。因此，上面的操作不会返回任何数据，除非回到第1个窗口提交交易或者回滚。SQL Server的数据锁定方案可能会降低系统的性能和效率。数据被锁定的时间越长，或者锁定的数据量越大，其他数据访问用户就越可能不得不等待其查询语句的执行。因此，从程序员的角度来看，对SQL Server编程的时候应该尽量地把交易代码设计得既小又快。在SQL Server的最近版本中，微软对SQL Server进行了某些修改，使其一次锁定的数据量大大减少，这是数据库设计中的一大重要改进。在6.5版及以前版本中，最少的数据锁定量是一页。哪怕你只在修改一行数据，而该行数据位于包含10行数据的一页上，则整页10行数据都会被锁

定。显然，这么大的数据锁定量增加了其他数据访问连接不得不等待数据修正完成的概率。在SQL Server 7中，微软引入了行锁定技术，这样，目前的SQL Server只锁定实际正被改变的数据行。SQL Server的解决方案听起来很简单，但实际上其幕后为提供足够的系统高性能而采取了很多措施。例如，如果你在同时修改多行数据，SQL Server则会把数据锁定范围提升到页级别乃至锁定整个数据表，从而不必针对每一记录跟踪和维护各自的数据锁。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