

## 数据库系统课程教案

## 第 22 单元

学时：2

教材内容	第 11 章 并发控制
基本知识 点	并发控制、封锁、封锁协议、活锁和死锁、并发调度的可串行性、封锁的粒度、两段锁协议
教学重点	并发操作可能产生数据不一致性的情况及其确切的含义；封锁的类型；不同性质封锁的性质和定义，相关的相容控制矩阵；封锁协议的概念；封锁粒度的概念；封锁协议与数据一致性的关系；并发调度的可串行性概念；两段锁协议与可串行性的关系；两段锁协议与死锁的关系
教学难点	两段锁协议与可串行性的关系；两段锁协议与死锁的关系；具有意向锁的多粒度封锁方法的封锁过程
要求掌握 内容	并发操作可能产生数据不一致性的情况及其确切的含义；封锁的类型；不同性质封锁的性质和定义，相关的相容控制矩阵；封锁协议的概念；封锁粒度的概念；封锁协议与数据一致性的关系；并发调度的可串行性概念；两段锁协议与可串行性的关系；两段锁协议与死锁的关系
教学思路， 采用的教学 方法和 辅助手段， 板书设计， 重点如何 突出，难点 如何解决， 师生互动 等	<p><b>教学思路：</b></p> <p><b>一、复习旧课，巩固上次授课主要内容</b></p> <ol style="list-style-type: none"> <li>1、简述事务的概念和性质。</li> <li>2、简述数据库恢复的技术和策略。</li> </ol> <p><b>二、导入新课，明确本次授课的目的与要求</b></p> <ol style="list-style-type: none"> <li>1、了解三类数据不一致问题。</li> <li>2、掌握封锁的概念和三级封锁协议。</li> <li>3、理解活锁、死锁与封锁粒度的概念。</li> <li>4、掌握两段锁协议和并发调度的可串行性。</li> </ol> <p><b>三、讲解本次授课的具体内容</b></p> <p><b>教学方法：</b>整合教学内容，强化基础训练；努力营造生动活泼的课堂气氛，搭建师生间良好的沟通渠道；采用多媒体教学与传统的板书设计相结合的方式，教学手段灵活多变。</p> <p><b>辅助手段：</b>通过 PPT 幻灯片演示结合板书设计和举例加以阐述。</p>
本章思考题 和作业	P280 第 3、6、8、9 题
主要教材 参考资料	<ol style="list-style-type: none"> <li>1. 《数据库系统概论》，萨师煊，王珊，高等教育出版社，2014.9</li> <li>2. 《数据库系统概论学习指导与习题解答》，王珊，张俊，高等教育出版社，2015.7</li> </ol>
备 注	

## 本次授课具体内容

### 第 11 章 并发控制

#### 11.1 概述

(一)并发操作的概念：多个用户同时对某个数据库对象进行的操作。

(二)并发操作可能引起的三类数据不一致问题

1、丢失修改：指两个事务 T1 和 T2 从数据库中读入同一数据并修改，T2 的提交结果破坏了 T1 提交的结果，导致 T1 的修改被丢失。

2、不可重复读：不可重复读是指事务 T1 读取数据后，事务 T2 执行更新操作，使 T1 无法再现前一次读取结果。不可重复读包括三种情况：

(1)事务 T1 读取某一数据后，事务 T2 对其做了修改，当 T1 再次读该数据时，得到了与前一次不同的值。

(2)事务 T1 按一定条件从数据库中读取了某些数据记录后，事务 T2 删除了其中部分记录，当 T1 再次按相同条件读取数据时，发现某些记录神秘地消失了。

(3)事务 T1 按一定条件从数据库中读取了某些数据记录后，事务 T2 插入了一些记录，当 T1 再次按相同条件读取数据时，发现多了一些记录。

3、读“脏”数据：事务 T1 修改某一数据，并将其写回磁盘，事务 T2 读取同一数据后，T1 由于某种原因被撤消，这时事务 T1 已修改过的数据恢复原值，T2 读到的数据就与数据库中的数据不一致，则 T2 读到的数据称为“脏”数据。

#### 11.2 封锁

(一)封锁：任何事务 T 在对某数据操作之前，先向系统发出请求对其加锁。加锁后事务 T 就对该数据拥有了一定的控制权，在事务 T 释放锁之前其它事务不能更新该数据。

1、排它锁(X 锁, 写锁)：若事务 T 对数据 A 加 X 锁，则只允许 T 读取和修改 A，其他事务不能再对 A 加任何锁，直到 T 释放 A 上的 X 锁。它保证了在 T 释放 A 上 X 锁之前其他事务不能再读取和修改 A。

2、共享锁(S 锁, 读锁)：若事务 T 对数据 A 加 S 锁，则事务 T 只可读 A，但不能修改 A，其他事务只能再对 A 加 S 锁，而不能加 X 锁，直到 T 释放 A 上的 S 锁。它保证了在 T 释放 A 上 S 锁之前其它事务可以读 A，但不能修改 A。

(二)X 锁和 S 锁的相容性

下表为 X 锁和 S 锁的相容矩阵，其中：Y 为相容，N 为不相容。若两个锁不相容，则后提出锁请求的事务必须等待。

## 本次授课具体内容（下续）

T1 \ T2	X	S	-
X	N	N	Y
S	N	Y	Y
-	Y	Y	Y

### 11.3 封锁协议

(一)封锁协议的概念：事务对数据封锁时，何时申请 X 锁或 S 锁、持锁时间、何时释放等有关封锁的规则。

(二)常用的封锁协议

1、一级封锁协议：事务 T 在修改数据 A 之前必须先对其加 X 锁，直到该事务结束才释放 X 锁。

2、二级封锁协议：一级封锁协议加上事务 T 在读取数据 A 前必须先对其加 S 锁，读完后立即释放 S 锁。

3、三级封锁协议：一级封锁协议加上事务 T 在读取数据 A 前必须先对其加 S 锁，直到该事务结束才释放 S 锁。

### 11.4 活锁与死锁

(一)活锁：某个事务永远处于等待封锁的状态。活锁的避免：采用“先来先服务”策略。(二)死锁：多个事务同时处于等待状态，每个事务都在等待其它事务释放锁使其能够继续执行，从而出现多个事务互相等待的僵局，每个事务永远不能结束。

#### 1、死锁的预防

(1)一次封锁法：要求每个事务必须一次将所有要使用的数据全部加锁，否则就不能继续执行。它存在两个问题：一是扩大了封锁范围；二是很难事先精确确定每个事务所要封锁的数据对象，只能将事务执行中可能要封锁的数据对象全部加锁，降低了并发度(2)顺序封锁法：预先对数据对象规定一个封锁顺序，所有事务都按这个顺序实行封锁。它也存在两个问题：一是数据库系统中可封锁的数据对象极多并且不断变化，要维护这些资源的封锁顺序非常困难；二是事务的封锁请求可以随着事务的执行而动态地决定，很难事先确定每一个事务要封锁哪些对象，也就很难按规定的顺序去施加封锁。如规定数据对象的封锁顺序为 A, B, C, D, E。事务 T3 起初要求封锁数据对象 B, C, E，但当它封锁了 B, C 后，才发现还需要封锁 A，这样就破坏了封锁顺序。

因此 DBMS 解决死锁更普遍采用诊断+解除的方法。

#### 2、死锁的诊断与解除

(1)诊断：DBMS 一般使用超时法或事务等待图法。

(2)解除：死锁发生后 DBMS 选择处理代价最小的事务将其回滚，并释放它所持有的所有锁。

### 11.5 并发调度的可串行性

(一)可串行化调度：若多个事务并发调度的结果与任一串行调度的结果相同，则该并发调度是可串行化调度，两段锁协议是保证并发调度可串行性的封锁协议。

(二)对于一个给定的并发调度，当且仅当它是可串行化的，才认为是正确的调度。例：设 A、B 的初值都是 2，现有两个事务：

事务 T1：读 B；A=B+1；写回 A。

事务 T2：读 A；B=A+1；写回 B。

对比这两个事务的三种不同的调度策略。

### 11.6 两段锁协议

(一)两段锁协议(Two Phase Locking, 2PL)的概念

事务在对任何数据进行读、写之前，首先要申请并获得该数据上的封锁；在释放一个封锁之后，事务不再申请和获得任何其他封锁。两段的含义是：

(1)扩展阶段：只获得锁，不释放锁。

(2)收缩阶段：只释放锁，不获得锁。

(二)2PL 协议与可串行化调度的关系：遵守 2PL 协议是可串行化调度的充分条件，但不是必要条件。

(三)2PL 协议与一次性封锁的关系：一次性封锁遵守 2PL 协议，但遵守 2PL 协议的事务可能发生死锁。

### 11.7 封锁的粒度

(一)封锁粒度：指封锁对象的大小，与并发度相关。

(二)多粒度封锁：一个系统可支持多种粒度供不同事务选择。对结点 P 封锁后，其后继结点也被施加相同锁。对结点 P 请求封锁时，必须判断 P 是否存在不相容锁：可能是 P 上的显式锁、可能是 P 的上层结点传递的隐式锁、可能是 P 的下层结点中已存在的某个显式锁。

(三)意向锁：对一个结点加意向锁，说明该结点的下层结点正拟加相应锁；对任一结点加锁时，必须先对其上层结点加意向锁。

1、意向排它锁(IX 锁)：若对一个数据对象加 IX 锁，表示它的后继结点正拟加 X 锁。如对某记录加 X 锁，则先对表和数据库加 IX 锁

2、意向共享锁(IS 锁)：若对一个数据对象加 IS 锁，表示它的后继结点正拟加 S 锁。如对某记录加 S 锁，则先对表和数据库加 IS 锁

3、共享意向排它锁(SIX 锁)：若对一个数据对象加 SIX 锁，表示对它先加 S 锁，再加 IX 锁，即  $SIX=S+IX$ 。如对某表加 SIX 锁，表示要读整个表(要加 S 锁)，同时会更新若干记录(要加 IX 锁)。具有意向锁的多粒度封锁方法中任意事务 T 要对一个数据对象加锁，必须先对它的上层结点加意向锁。申请封锁时应该按自上而下的次序进行；释放封锁时则应该按自下而上的次序进行。该封锁方法提高了系统的并发度，减少了加锁和解锁的开销，已在实际 DBMS 中得到了广泛应用，如 Oracle 中就采用了这种封锁方法。

## 本次授课小结

本次授课讲述了三类数据不一致问题、封锁的概念和三级封锁协议、活锁、死锁与封锁粒度的概念、两段锁协议和并发调度的可串行性。

学生课后复习时应着重于其中的第 2、4、5 点内容，为进一步学习后续章节打好基础。

实验

综合实验一