

封锁技术



前言

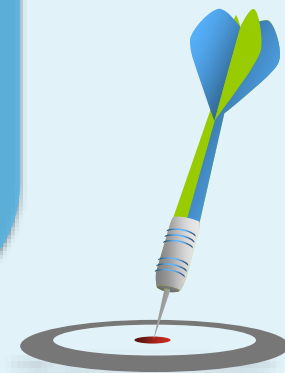
基于锁的并发控制的基本思想

- 事务对需要访问的数据库对象进行操作之前，先向系统发出封锁请求，获得所访问的数据库对象上的锁，限制并发的其他事务对这些数据库对象的访问。



讲授内容

- ◆ 1 封锁模式
 - 共享锁
 - 排他锁
- ◆ 2 两阶段封锁协议





封锁模式

- 共享锁（**S**hare Lock, 简称 **S** 锁, 又称读锁）
- 排他锁（e**X**clusive Lock, 简称 **X** 锁, 又称写锁）



封锁模式

共享锁

- 事务T想**读取**数据库对象A而**不更新**A，事务T必须申请获得A上的共享锁；
- 若申请成功，则事务在数据库对象A上加共享锁，事务T**可以读A但不能写A**；
- 其他事务**只能再对A加共享锁，而不能加排他锁**。



封锁模式

排他锁

- 事务T不仅要**读取**数据库对象A还要**更新**A，事务T必须申请获得A上的排他锁；
- 若申请成功，则事务T在数据库对象A上加排他锁，事务T**可以读A还能写A**；
- 其他事务**不能再对A加任何类型的锁**。



封锁模式

- 任何数据库对象上只能有一个**排他锁**，或者没有排他锁而有**多个共享锁**。
- 事务T想要读数据库对象A并可能更新A，应首先申请A上的一个共享锁，当事务T准备好为A写入新值时再申请将加在A上的共享锁**升级**为排他锁。



封锁模式

- 锁相容性矩阵描述在同一数据库对象上已经被加锁的情况下，能否同意其他封锁申请的策略。

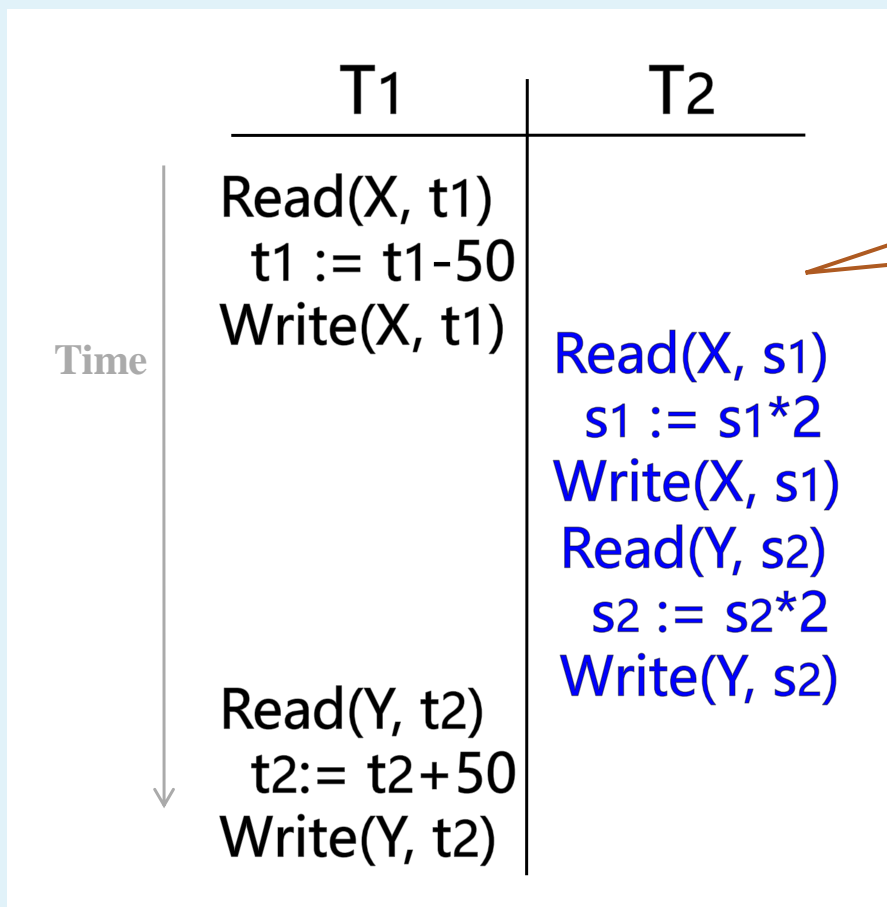
事务T2申请的锁

事务T1获得的锁	T2	S	X	—
	T1	S	X	—
	S	Y	N	Y
	X	N	N	Y
	—	Y	Y	Y

Y=Yes 相容
N=No 不相容



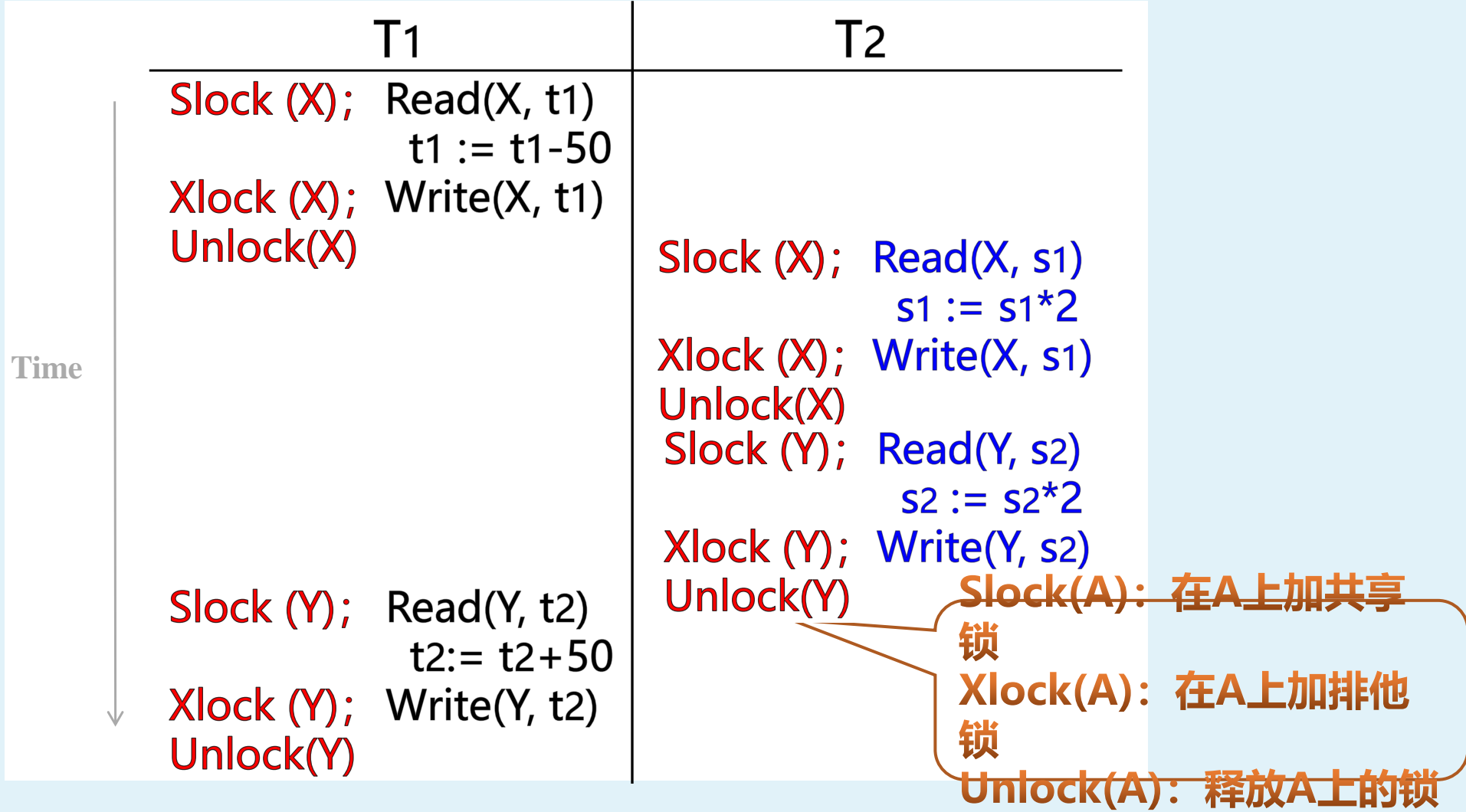
封锁模式



非可串行化的调度



封锁模式





封锁协议

- 封锁协议是对事务何时申请要访问的数据库对象上的锁、何时释放所获得的锁等约定的一些**规则**。
- 约定不同的**规则**就形成不同的**封锁协议**。
- **两阶段封锁协议**(Two-Phase Locking Protocol, 2PL)是最常用的一种实现**可串行化**的封锁协议。



两阶段封锁协议

- 获得锁阶段（扩展阶段）
 - 事务申请所需要的所有锁，不能释放锁。
- 释放锁阶段（收缩阶段）
 - 事务释放所获得的所有锁，不能再申请锁。

Time → Slock(A) ... Slock(B) ... Xlock(C) ... Unlock(B)... Unlock(A)... Unlock(C)



两阶段封锁协议

严格的两阶段封锁协议

- 事务T在读数据库对象前必须获得该数据库对象上的读锁。
- 事务T在更新数据库对象前必须获得该数据库对象上的写锁，或将读锁**升级**到写锁。
- 若事务B对数据库对象的封锁请求与事务A已获得的锁**不相容**，事务B将**等待**，直到事务A释放其所拥有的锁为止。
- 事务所获得的锁将一直**保持到事务结束**才释放。

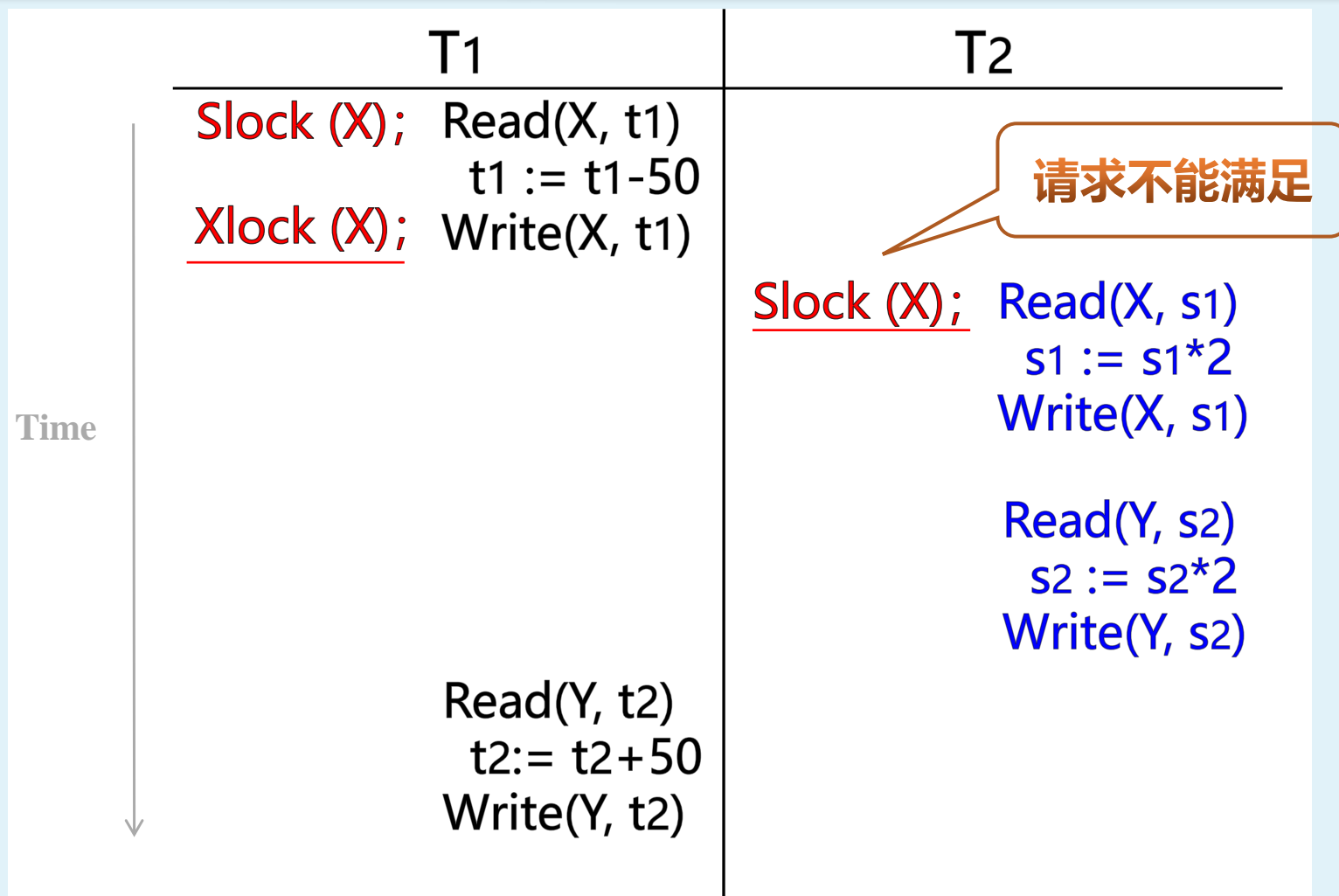


两阶段封锁协议

	T1	T2
Time ↓	Slock (X); Read(X, t1) t1 := t1-50	Slock (X); Read(X, s1) s1 := s1*2
	Xlock (X); Write(X, t1)	Xlock (X); Write(X, s1)
	Unlock(X)	Unlock(X)
	Slock (Y); Read(Y, t2) t2:= t2+50	Slock (Y); Read(Y, s2) s2 := s2*2
	Xlock (Y); Write(Y, t2)	Xlock (Y); Write(Y, s2)
	Unlock(Y)	Unlock(Y)

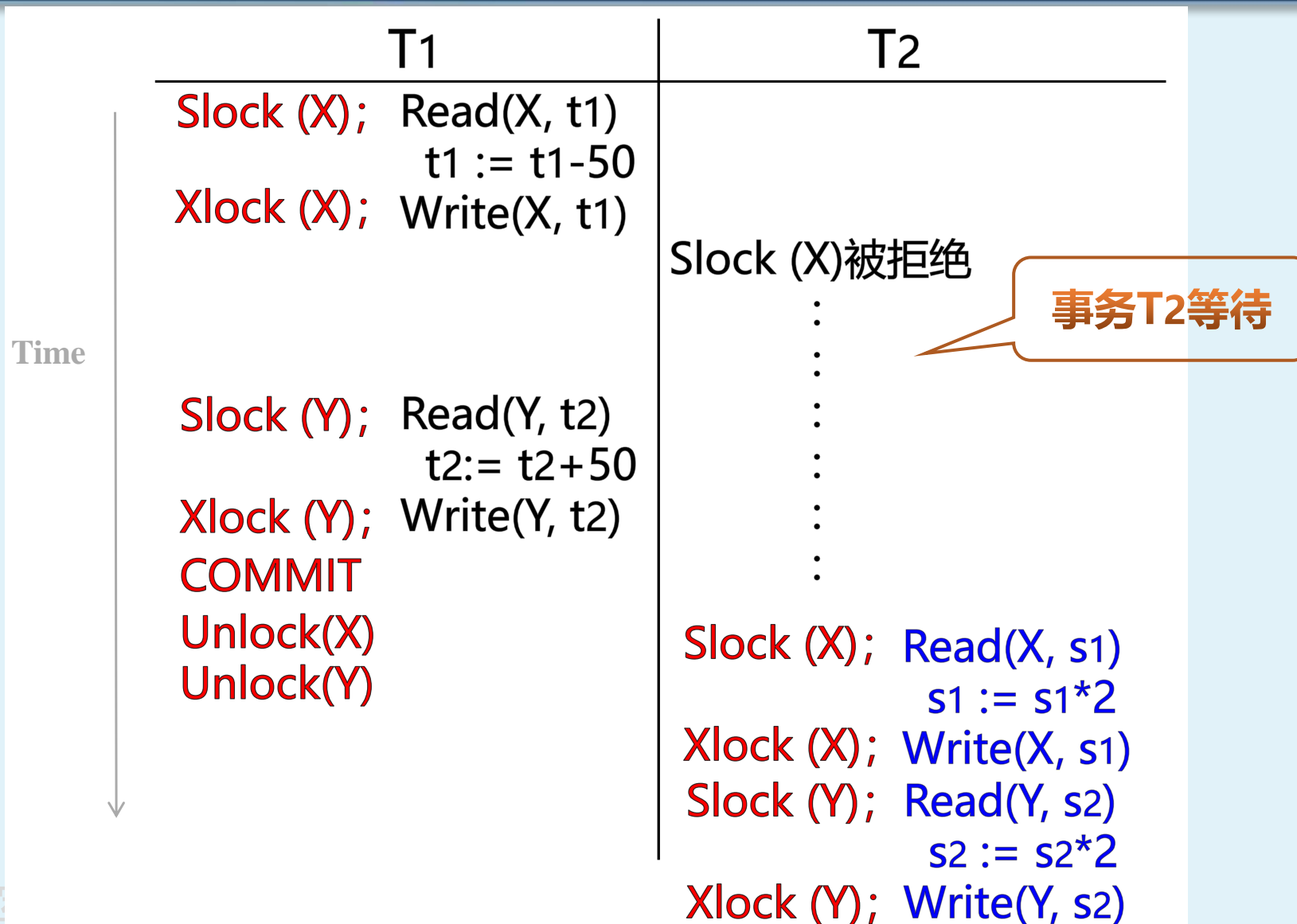


两阶段封锁协议





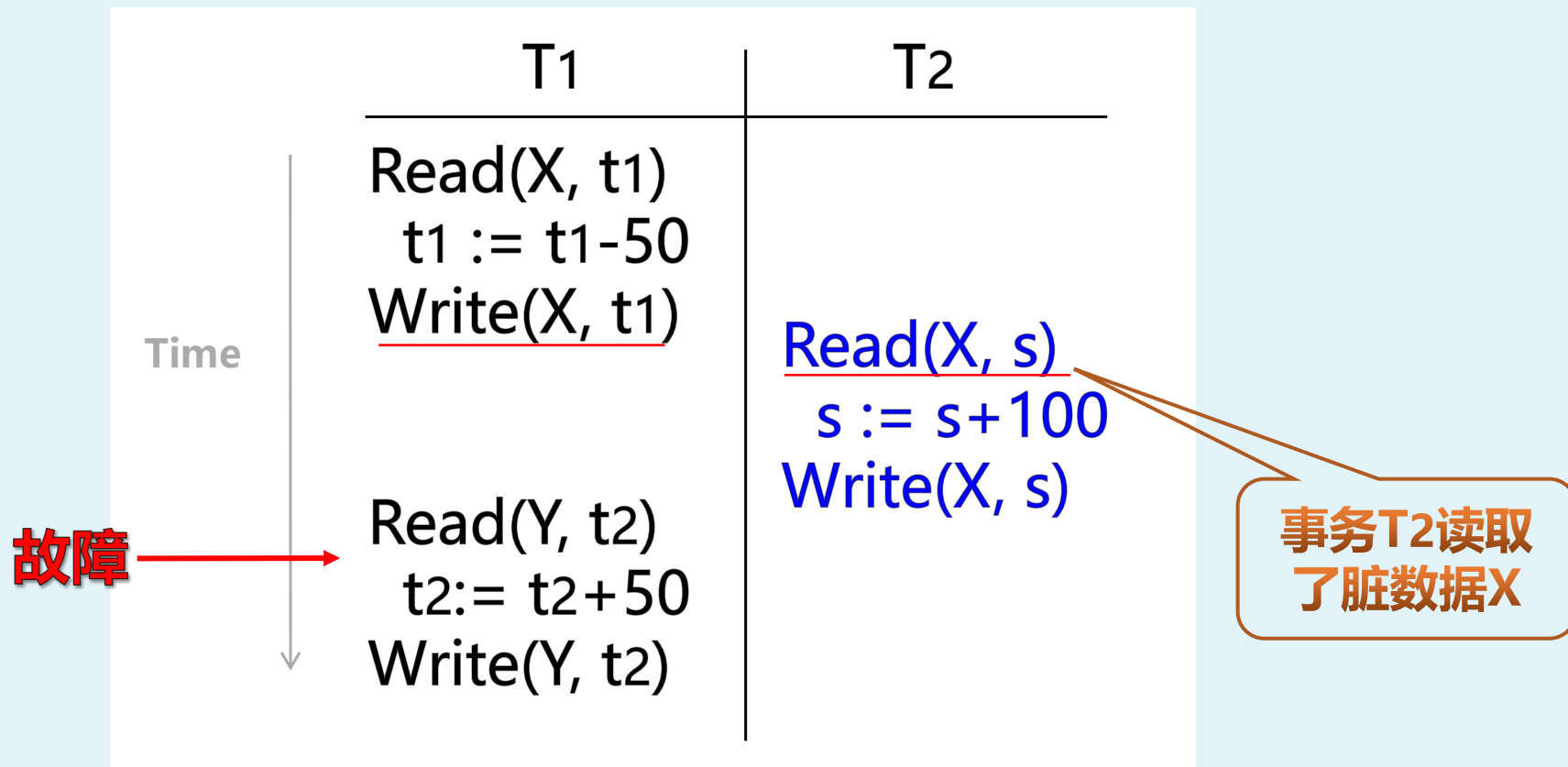
两阶段封锁协议





两阶段封锁协议

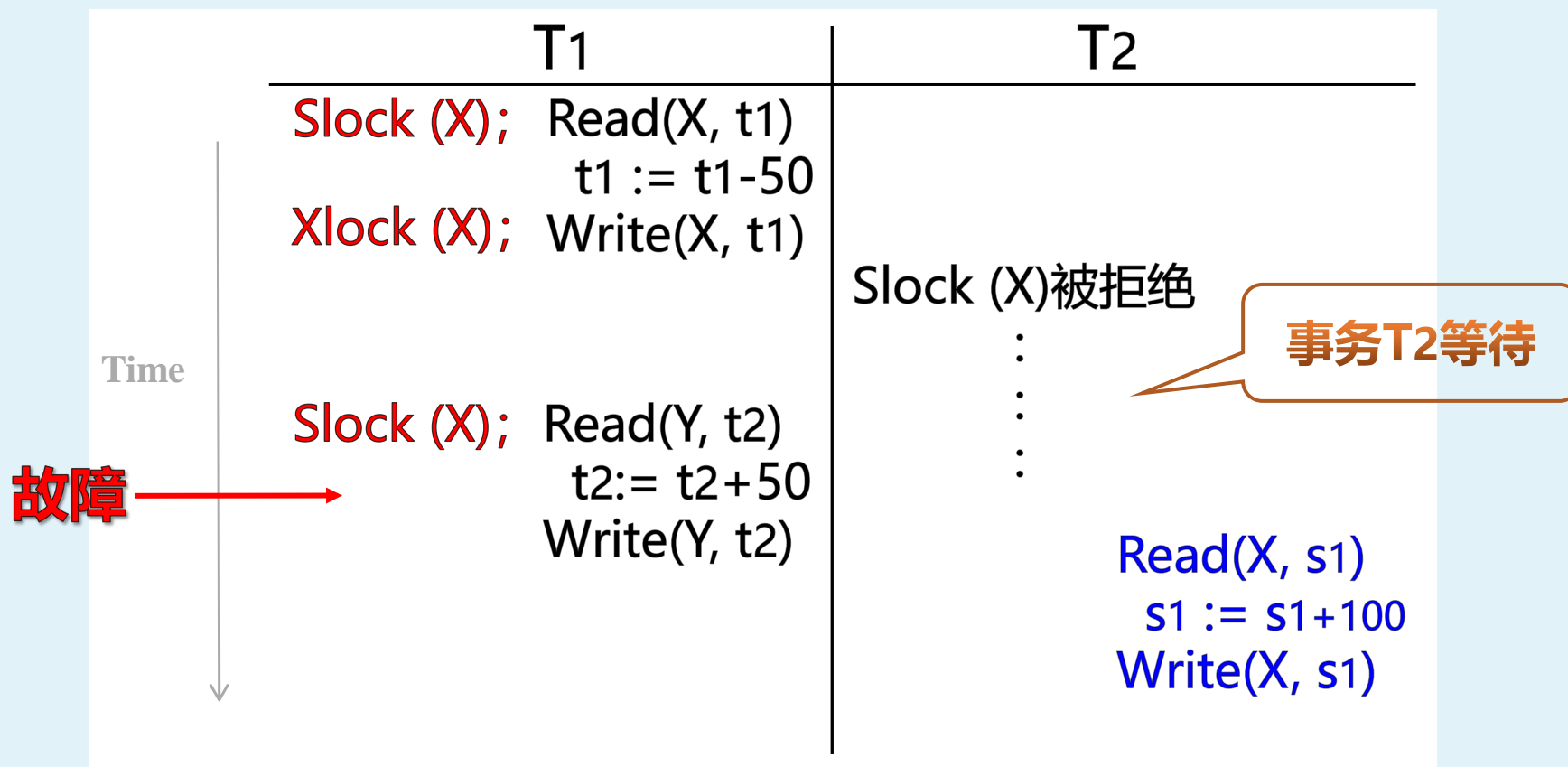
脏读





两阶段封锁协议

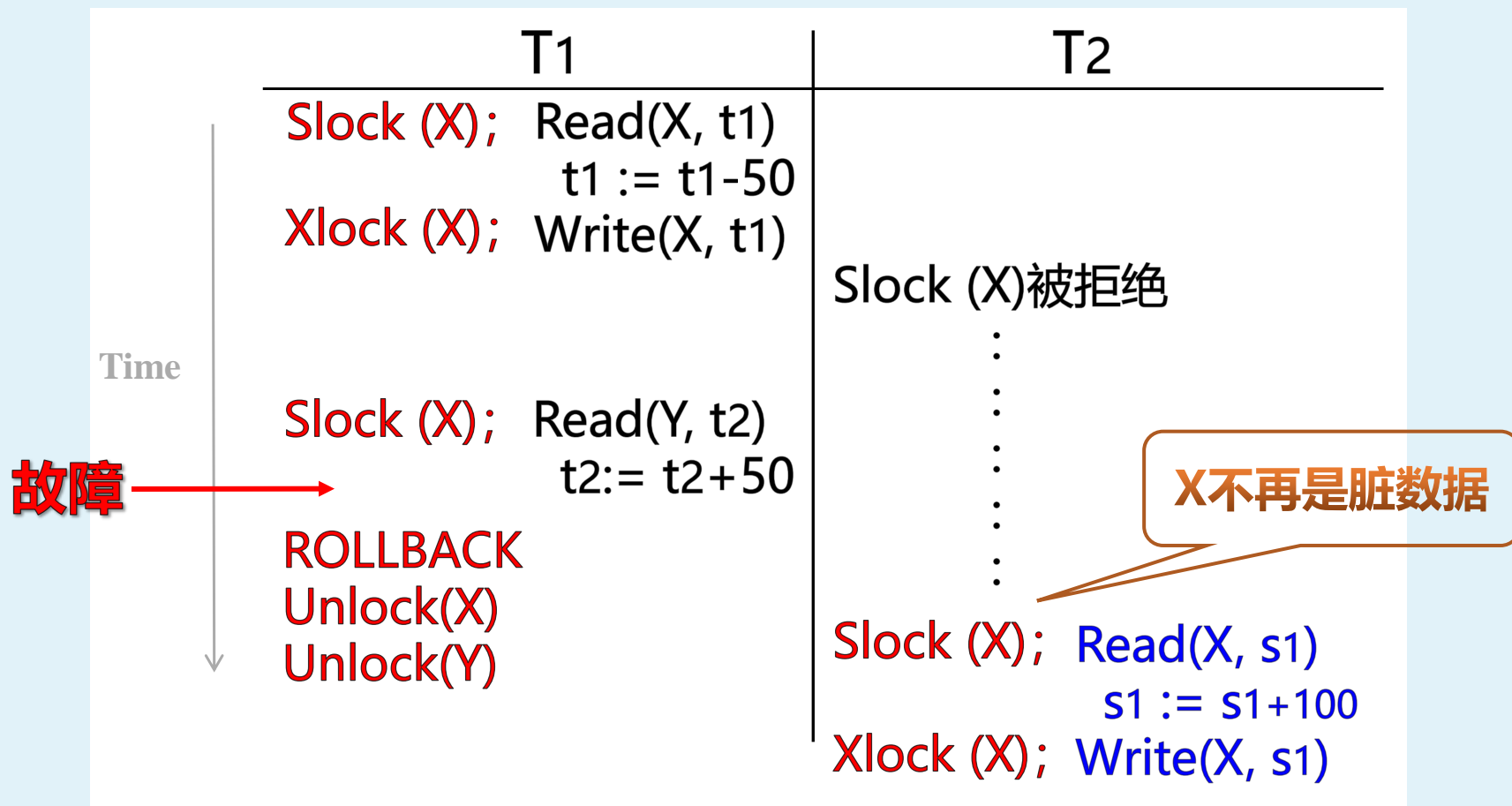
利用严格两阶段封锁协议解决“脏读”问题





两阶段封锁协议

利用严格两阶段封锁协议解决“脏读”问题





小结

- ❏ 封锁技术通过**共享锁**间的相容性，以及**排他锁**的排他性，使并发调度中的非冲突操作并发执行、冲突操作串行执行，实现了**冲突可串行化**。
- ❏ DBMS具体实现时，可能会使用共享锁和排他锁之外的其他类型的锁、相应的**封锁策略**及对应的**锁相容矩阵**，采用更能便于应用执行的**封锁协议**。