

Homework 8

PB17000297 罗晏宸

May 19 2020

1 证明：如果一个并发调度 S 中的所有事务都遵循 2PL，则该调度必定是可串化调度。

解 如果一个并发调度 S 中的所有事务都遵循 2PL，则加锁操作一定在解锁前。若在冲突图中有关于 A 的边 $T_1 \rightarrow T_2$ ，则 T_2 对 A 的锁的申请一定在 T_1 释放之后才进行，之后 T_1 对 A 不再进行读或写。假设冲突图中有环，则存在申请过 A 的锁的事务在另一事务之后再次申请，这是违背 2PL 的。因此调度是可串化的。

2 在锁的相容性矩阵中， $\langle S, U \rangle$ （即 T_1 先持有了某对象上的 S 锁， T_2 再申请同一对象上的 U 锁时）是相容的，而 $\langle U, S \rangle$ 是不相容的。请解释一下 DBMS 为什么要设计这样不对称的锁相容性规则。

解 若 $\langle S, U \rangle$ 不相容，则不论 T_1 是否对对象进行修改，都不允许任何 T_2 申请 U 锁，这会影响到并发性能。若 $\langle U, S \rangle$ 是相容的，则若 T_1 意图对对象进行修改，则不能阻止后续的事务申请同一对象上的 S 锁。

3 采用了两阶段锁协议的事务是否一定不会出现脏读问题？如果是，请解释理由；如果不是，请给出一个反例。

解 不是，反例如下：

T_1	T_2
1 LOCK(A);	
2 READ(A, t);	LOCK(A);
3 $t \leftarrow t + 100$;	WAIT;
4 WRITE(A, t);	WAIT;
5 UNLOCK(A);	WAIT;
6 ROLLBACK;	READ(A, t);

脏读

- 4 判断下面的并发调度是否冲突可串？如果是，请给出冲突等价的串行调度事务顺序；如果不是，请解释理由。

$w_3(D) \ r_1(A) \ w_2(A) \ r_4(A) \ r_1(C) \ w_2(B) \ r_3(B) \ r_3(A) \ w_1(D) \ w_3(B) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$

解 是冲突可串的，冲突等价的串行调度事务顺序如下可得：

$w_3(D) \ r_1(A) \ w_2(A) \ r_4(A) \ r_1(C) \ w_2(B) \ r_3(B) \ r_3(A) \ w_1(D) \ w_3(B) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$
 $w_3(D) \ r_1(A) \ w_2(A) \ r_4(A) \ r_1(C) \ w_2(B) \ r_3(B) \ r_3(A) \ w_3(B) \ w_1(D) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$
 $w_3(D) \ r_1(A) \ w_2(A) \ r_1(C) \ r_4(A) \ w_2(B) \ r_3(B) \ r_3(A) \ w_3(B) \ w_1(D) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$
 $w_3(D) \ r_1(A) \ w_2(A) \ r_1(C) \ w_1(D) \ w_2(B) \ r_3(B) \ r_3(A) \ w_3(B) \ r_4(A) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$
 $w_2(A) \ r_1(A) \ w_3(D) \ r_1(C) \ w_1(D) \ w_2(B) \ r_3(B) \ r_3(A) \ w_3(B) \ r_4(A) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$
 $w_2(A) \ r_1(A) \ w_2(B) \ r_1(C) \ w_1(D) \ w_3(D) \ r_3(B) \ r_3(A) \ w_3(B) \ r_4(A) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$
 $w_2(A) \ w_2(B) \ r_1(A) \ r_1(C) \ w_1(D) \ w_3(D) \ r_3(B) \ r_3(A) \ w_3(B) \ r_4(A) \ r_4(B) \ r_4(C) \ w_4(C) \ w_4(B)$

$\underbrace{\hspace{10em}}_{T_2} \quad \underbrace{\hspace{10em}}_{T_1} \quad \underbrace{\hspace{10em}}_{T_3} \quad \underbrace{\hspace{10em}}_{T_4}$

- 5 判断下面的并发调度能否由一个使用 2PL 的调度器生成。如果可以，给出一种可能的加锁解锁的顺序；如果不行，请解释理由。

$w_3(x) \ w_4(y) \ w_1(z) \ w_3(a) \ w_3(y) \ w_2(a) \ w_3(x) \ w_3(x) \ w_1(a) \ w_3(y) \ w_2(z) \ w_1(x)$

解 不可以，一个不满足 2PL 的加锁解锁顺序如下，注意到在释放对 a 的锁之后，事务 T_1 进入缩减阶段，不能再申请新锁，而事务最后的 $w_1(x)$ 要求对 x 增加排它锁，因此并发调度不能由一个使用 2PL 的调度器生成。

