



11.4 并发调度的可串行性

👉 什么样的并发操作调度是正确的

计算机系统对并行事务中并行操作的调度是随机的，而不同的调度可能会产生不同的结果。

- * 如果一个事务运行过程中没有其它事务在同时运行，也就是说它没有受到其它事务的干扰，那么就可以认为该事务的运行结果是正常的或者预想的。
- * 将所有事务串行起来的调度策略一定是正确的调度策略。
- * 以不同的顺序串行执行事务也有可能产生不同的结果，但由于不会将数据库置于不一致状态，所以都认为是正确的。
- * 几个事务的并行执行是正确的，**当且仅当其结果与按某一次序串行地执行它们时的结果相同**。这种并行调度策略称为可串行化(Serializable)的调度。



11.4 并发调度的可串行性

例：现在有两个事务，分别包含下列操作：

事务1：读B；A=B+1；写回A；

事务2：读A；B=A+1；写回B；

假设A的初值为2，B的初值为2。

调度(a)

| T ₁ | T ₂ |
|---|---|
| Slock B Y=B=2 Unlock B Xlock A A=Y+1 写回A(=3) Unlock A | |
| | Slock A X=A=3 Unlock A Xlock B B=X+1 写回B(=4) Unlock B |

调度(b)

| T ₁ | T ₂ |
|---|--|
| | SlockA X=A=2 Unlock A Xlock B B=X+1 写回B(=3) Unlock B |
| Slock B Y=B=3 Unlock B Xlock A A=Y+1 写回A(=4) Unlock A | |

调度(a)和调度(b)
是串行调度，
这种调度是正
确的。

结论：可串行性
是并行事务正
确性的唯一准
则。



11.4 并发调度的可串行性

| 调度(c) | | 调度(d) | |
|-----------------------------|-----------------------------|------------------------------|---|
| T ₁ | T ₂ | T ₁ | T ₂ |
| Slock B Y=B=2 | | Slock B Y=B=2 | |
| | Slock A X=A=2 | Unlock B Xlock A | |
| Unlock B | | | Slock A 等待 等待 等待 |
| Xlock A A=Y+1 写回A(=3) | Unlock A | A=Y+1 写回A(=3) Unlock A | X=A=3 |
| | Xlock B B=X+1 写回B(=3) | | Unlock A |
| Unlock A | Unlock B | | Xlock B B=X+1 写回B(=4) Unlock B |

- (1) 调度(c)执行结果与(a)、(b)的结果都不同，是错误的调度
- (2) 调度(d)执行结果与串行调度(a)的执行结果相同，是正确的调度

注：有些情况下也可能在错误的调度下产生正确的结果，但这只能是巧合。

产生错误调度的原因在于调度过程中破坏了事务的隔离性，而其根源在于事务程序编写错误。