* 两阶段封锁协议为什么能保证冲突可串行化？

**Answer:** Suppose two-phase locking does not ensure serializability. Then

there exists a set of transactions *T*0, *T*1 ... *Tn*−1 which obey 2PL and which

produce a nonserializable schedule.A non-serializable schedule implies a

cycle in the precedence graph, andwe shall showthat 2PL cannot produce

such cycles. Without loss of generality, assume the following cycle exists

in the precedence graph: *T*0 → *T*1 → *T*2 → ... → *Tn*−1 → *T*0. Let a*i* be

the time at which *Ti* obtains its last lock (i.e. *Ti* ’s lock point). Then for all

transactions such that *Ti* → *Tj* , a*i* < a*j* . Then for the cycle we have

a0 < a1 < a2 < ... < a*n*−1 < a0

Since a0 < a0 is a contradiction, no such cycle can exist. Hence 2PL

cannot produce non-serializable schedules. Because of the property that

for all transactions such that *Ti* → *Tj* , a*i* < a*j* , the lock point ordering

of the transactions is also a topological sort ordering of the precedence

graph. Thus transactions can be serialized according to their lock points.

* 比较顺序索引和Hash索引，以及他们的优点缺点

Cost of periodic re-organization//定期重新组织的开销

Relative frequency of insertions and deletions//相对频繁的更新

Expected type of queries:

Hashing is generally better at retrieving records having a specified value of the key.

//Hashing：检索特定值

If range queries are common, ordered indices are to be preferred

//顺序索引：范围查询

* 描述归并连接

排序 归并-连接

* 描述块嵌套连接

every block of inner relation is paired with every block of outer relation.

* 证明是否为无损分解

若 A∩B 是 A或B的超码，则R分解成AB是无损分解

* 证明是否为保持依赖[dependency preserving]