

Hand-out

Advanced database technology

March 14, 2005

We consider the following three transactions A, B, and C operating on the relation `Primes(p INT)`, which has no indexes and is initially empty.

A	B	C
<code>INSERT INTO Primes VALUES (2);</code> <code>UPDATE Primes WHERE p=3 SET p=6;</code> <code>COMMIT;</code>	<code>INSERT INTO Primes VALUES (3);</code> <code>DELETE FROM Primes WHERE p=2;</code> <code>COMMIT;</code>	<code>SELECT count(*)</code> <code>FROM Primes</code> <code>GROUP BY p;</code> <code>COMMIT;</code>

1. What state would the database be in after each of the possible serial schedules for the transactions?
2. Consider the locking system with shared, exclusive, and update locks. What locks could be requested for the transactions? In each case, list the possible behaviors of the locking scheduler when the transactions are started at the same time.
3. Suppose timestamp-based concurrency control is used. In what situations would one of the transactions be forced to abort?
4. Explain what may be the result of running the transactions at each of the three SQL isolation levels `READ COMMITTED`, `REPEATABLE READ`, and `SERIALIZABLE`.