PESSIMISTIC vs. OPTIMISTIC concurrency control

solidDB® offers two different concurrency control mechanisms, *pessimistic* and *optimistic*.

- **Pessimistic concurrency control** (or *pessimistic locking*) is called "pessimistic" because the system assumes the worst — it assumes that two or more users will want to update the same record at the same time, and then prevents that possibility by locking the record, no matter how unlikely conflicts actually are.

  The locks are placed as soon as any piece of the row is accessed, making it impossible for two or more users to update the row at the same time. Depending on the lock mode (*shared*, *exclusive*, or *update*), other users might be able to read the data even though a lock has been placed. For more details on the lock modes, see Lock modes: shared, exclusive, and update.

- **Optimistic concurrency control** (or *optimistic locking*) assumes that although conflicts are possible, they will be very rare. Instead of locking every record every time that it is used, the system merely looks for indications that two users actually did try to update the same record at the same time. If that evidence is found, then one user's updates are discarded and the user is informed.

  For example, if User1 updates a record and User2 only wants to read it, then User2 simply reads whatever data is on the disk and then proceeds, without checking whether the data is locked. User2 might see slightly out-of-date information if User1 has read the data and updated it, but has not yet committed the transaction.

  Optimistic locking is available on disk-based tables (D-tables) only.

  The solidDB® implementation of optimistic concurrency control uses multiversioning.