

Lecture 26 — Weak Consistency

Jeff Zarnett

Transaction Isolation

The SQL standard provides several isolation levels, and it's not necessary that we stick with serializability as the level of consistency we are willing to accept. We can weaken the rules a bit to get some more performance out of the database and that gives us what is called *weak consistency*: there are rules, but they are more like... guidelines?

Degree-Two Consistency. The idea of *degree-two consistency* is to prevent cascading rollbacks (aborts) without guaranteeing serializability. There are shared and exclusive locks, but two-phase behaviour is not required; in fact shared locks can be released at any time and locks can be acquired at any time, but exclusive locks must be held until the transaction commits or aborts [?]. This means that we might read out of date data, but uncommitted values can never be read, so the level of transaction isolation here is “read-committed”.

Cursor Stability. Cursor stability is a form of degree-two consistency for programs that iterate over some set of tuples using an iterator or cursor [?]. This means that the tuples are examined or processed one at a time, in some order. To make sure that this works alright, the tuple currently being examined needs to be locked: before processing it is locked in shared mode; if any processing changes that tuple then it must first be locked in exclusive mode.