

Control de concurrencia

Para lograr el objetivo de que se cumpla el aislamiento hay dos enfoques:

- 1. Optimista, deja hacer y luego validar, si falló hace rollback.*
- 2. Pesimista, busca garantizar que no se produzcan conflictos. Hay dos técnicas:*

- Control de concurrencia basada en locks.*
- Control de concurrencia basado en timestamps.*

Elegiremos esta técnica, dado que es la más utilizada en los SGBD

Control de concurrencia

*Los **locks** o candados son variables asociadas a determinados recursos, que permiten regular el acceso a los mismos en los sistemas concurrentes.*




Control de concurrencia con locks

Un lock debe disponer de dos primitivas de uso, que permiten tomar $\text{lock}(X)$ $L(X)$ y liberar el recurso $\text{unlock}(X)$ $U(X)$.

Cuando una transacción tiene un lock sobre un ítem X , ninguna otra transacción puede adquirir un lock sobre el mismo hasta tanto la primera no lo libere.

$\text{Lock}(X)$ requiere escribir y leer una variable, por lo tanto no puede estar solapada con una ejecución similar en otra transacción.

Control de concurrencia con locks



Los tipos de locks principales son de lectura o “compartidos” y de escritura o de “acceso exclusivo”.

Cuando una transacción tiene un lock de acceso exclusivo ninguna otra transacción puede tener un lock de ningún tipo sobre él.

Pero muchas transacciones pueden poseer locks de acceso compartido simultáneamente.

Pero con los locks solo no alcanza...

Una transacción podría adquirir un lock sobre un elemento para leerlo, liberarlo y luego volver a tomar un lock para modificarlo. Si en el medio otra transacción lo lee y escribe usando locks, seguramente se produciría alguna anomalía.

Entonces hace falta un protocolo que nos garantice la seriabilidad.