

Protocolos de prevención de deadlocks

2PL conservador: *que cada transacción **bloquee con antelación** todos los elementos que necesita. Si alguno no puede bloquearlo, espere y reintenta. Limita la concurrencia.*

-

Ordena todos los elementos de la base y se asegura de que una transacción que necesite varios elementos los bloqueará en ese orden. Obliga a conocer este orden.

-

Ejemplo de 2PL conservador

T1	T2	A	B
l(A); l(B)		25	25
A = A + 100			
w(A); u(A)		125	
r(B); B = B + 100	l(A); l(B) DENIED		
w(B); u(B)			
	l(A); l(B)	250	
	r(A)		
	A = A * 2		
	w(A)		125
	u(A); r(B)		
	B = B * 2		
	w(B); u(B)		250

El resultado es el mismo que una ejecución serial.

Problemas...

*Aunque tengamos un solapamiento serializable de transacciones, si una transacción T_i es abortada, el SGBD debe mantener la consistencia de la base. Si las modificaciones hechas por T_i fueron leídas por otras transacciones, entonces será necesario hacer el rollback de todas las transacciones involucradas en una **cascada de transacciones**.*

**Necesitamos
un nuevo
protocolo...**