

# Anomalie di concorrenza

**Lettura Sporca** : **READ COMMITTED** (*default*)

Una transazione legge dati non ancora confermati da un'altra transazione, che potrebbero quindi essere scartati.

```
-- T1

BEGIN;

UPDATE clienti SET saldo = 1000 WHERE id = 1;


-- T2

SELECT saldo FROM clienti WHERE id = 1; -- 1000


-- T1

ROLLBACK; -- valore torna come prima
```

---

**Perdita di Aggiornamento** : **REPEATABLE READ**

Due transazioni aggiornano lo stesso dato senza sapere l'una dell'altra, il secondo aggiornamento sovrascrive il primo.

```
-- T1

BEGIN;

SELECT saldo FROM conto WHERE id = 1; -- 100

UPDATE conto SET saldo = 110 WHERE id = 1;


-- UPDATE di T2

COMMIT;


-- T2

BEGIN;

SELECT saldo FROM conto WHERE id = 1; -- 100

UPDATE conto SET saldo = 120 WHERE id = 1;


-- COMMIT di T1

COMMIT;
```

---

### **Lettura Inconsistente/Non Ripetibile** : REPEATABLE READ

Una transazione rilegge i dati che ha letto in precedenza e rileva che i dati sono stati modificati da un'altra transazione.

```
-- T1  
  
BEGIN;  
  
SELECT prezzo FROM articoli WHERE id = 5; -- 10  
  
  
-- T2  
  
BEGIN;  
  
UPDATE articoli SET prezzo = 15 WHERE id = 5;  
  
COMMIT;  
  
  
-- T1  
  
SELECT prezzo FROM articoli WHERE id = 5; -- 15  
  
COMMIT;
```

---

### **Aggiornamento Fantasma** : REPEATABLE READ

Una transazione aggiorna un insieme di righe in base a una condizione, ma un'altra modifica righe esistenti facendole rientrare nella condizione iniziale.

```
--T1  
  
BEGIN;  
  
UPDATE CLIENTI SET premio = TRUE WHERE punti > 100;  
  
--COMMIT di T2  
  
COMMIT;  
  
  
--T2  
  
BEGIN;  
  
UPDATE clienti SET punti = 110 WHERE id = 1;  
  
COMMIT;
```

---

### **Inserimento Fantasma** : REPEATABLE READ

Una transazione esegue due SELECT con la stessa condizione, ma nel secondo risultato compaiono nuove righe a causa dell'inserimento da parte di un'altra transazione di righe che soddisfano la condizione.

```
-- T1

BEGIN;

SELECT * FROM eventi WHERE data > '2024-01-01';

-- T2

BEGIN;

INSERT INTO eventi VALUES ('2024-01-02', ...);

COMMIT;

-- T1

SELECT * FROM eventi WHERE data > '2024-01-01';

COMMIT;
```

---

### **Mancata Serializzazione** : SERIALIZABLE (*liv minimo solo per questa anomalia*)

Due transazioni si intersecano in modo tale da portare il sistema in uno stato impossibile da ottenere in un'esecuzione sequenziale.

```
--T1

BEGIN;

SELECT saldo FROM CONTO WHERE id = 1; --100

UPDATE CONTO SET saldo = saldo - 100 WHERE id = 1;

COMMIT;

-- T2 (contemporaneamente)

BEGIN;

SELECT saldo FROM CONTO WHERE id = 1; --100

UPDATE CONTO SET saldo = saldo - 100 WHERE id = 1;

COMMIT;

-- Entrambe prelevano 100$ dal conto ma non c'erano 200$ dentro!
```

