

GESTION DES ACCES CONCURRENTS / ISOLATION

Scénarios sur la base Aero

T1 : UPDATE; **T2** : UPDATE; **T2** : INSERT;
T3 : SELECT,

L'exécution entrelacée des transactions doit donner le même résultat qu'une exécution séquentielle



Transaction 1

UPDATE ...
UPDATE ...
COMMIT

Transaction 2

UPDATE ...
INSERT...
COMMIT

Transaction 3

SELECT
SELECT
COMMIT

SCENARIO 1 / ISOLATION

Transaction 1

- 1°) ...
- 2°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 10;
- 3°) SELECT Salaire FROM Pilote
WHERE Matricule = 10;
- 4°) ...
- 5°) COMMIT;

Transaction 2

- 1°) SELECT Salaire FROM Pilote
WHERE Matricule = 10;
- 2°) ...
- 3°) ...
- 4°) SELECT Salaire FROM Pilote
WHERE Matricule = 10;
- 5°) ...
- 6°) SELECT Salaire FROM Pilote
WHERE Matricule = 10;
- 7°) COMMIT;

Conclusion ?

SCENARIO 2 / ISOLATION

Transaction 1

- 1°) ...
- 2°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 20;
- 4°) SELECT Salaire FROM Pilote;
- 5°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 10;
- 6°) ...
- 7°) COMMIT;

Transaction 2

- 1°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 10;
- 3°) SELECT Salaire FROM Pilote;
- 4°) ...
- 5°) ...
- 6°) COMMIT;

Conclusion ?

SCENARIO 3 / ISOLATION

Transaction 1

- 1°) ...
- 2°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 20;
- 3°) SELECT Salaire FROM Pilote;
- 5°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 10;
- 6°) ...
- ...

Transaction 2

- 1°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 10;
- 2°) ...
- 4°) SELECT Salaire FROM Pilote;
- 5°) ...
- 6°) UPDATE Pilote SET Salaire = Salaire + 100
WHERE Matricule = 20;
- ...

Conclusion ?