

# Accès concurrents aux bases de données

Olivier Cailloux

LAMSADE, Université Paris-Dauphine

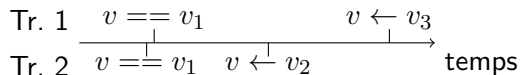
Version du 4 février 2020

# Nécessité des transactions atomiques

- Accès concurrents à DB : risques
- Si une transaction par statement ?

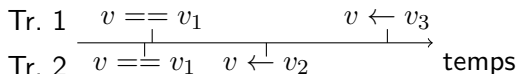
# Nécessité des transactions atomiques

- Accès concurrents à DB : risques
- Si une transaction par statement ?



# Nécessité des transactions atomiques

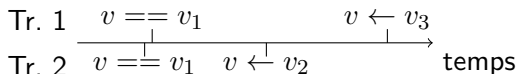
- Accès concurrents à DB : risques
- Si une transaction par statement ?



- Transaction atomique non triviale (couvrant un ensemble de statements) permet de lire-puis-écrire sans interruption
- Limiter : transaction terminée par *commit* ou *rollback*
- Implémentation naïve : DB verrouillée pour un utilisateur pendant le temps de la transaction
- Problème ?

# Nécessité des transactions atomiques

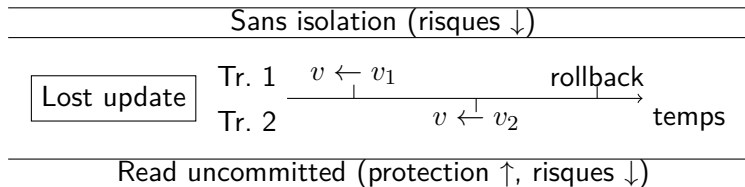
- Accès concurrents à DB : risques
- Si une transaction par statement ?



- Transaction atomique non triviale (couvrant un ensemble de statements) permet de lire-puis-écrire sans interruption
- Limiter : transaction terminée par *commit* ou *rollback*
- Implémentation naïve : DB verrouillée pour un utilisateur pendant le temps de la transaction
- Problème ? Souvent trop peu efficace

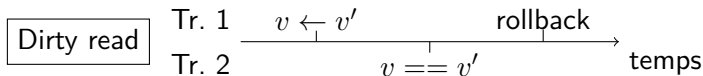
# Limites de transactions, et puis ?

- Les limites de transactions permettent l'isolation
- Isolation fournie par le SGBD
- Selon niveau d'isolation demandé
- Sans isolation : risque que rollback écrase des modifications indépendantes

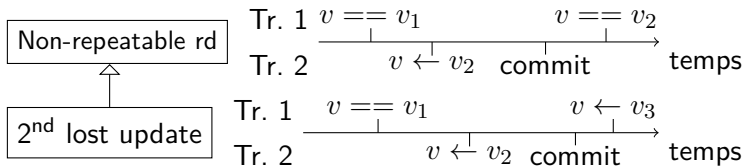


# Niveaux d'isolation

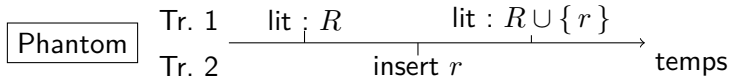
Read uncommitted (protection  $\uparrow$ , risques  $\downarrow$ )



Read committed (protection  $\uparrow$ , risques  $\downarrow$ )



Repeatable read (protection  $\uparrow$ , risques  $\downarrow$ )



Serializable (protection  $\uparrow$ )

# Niveaux d'isolation

- Quatre niveaux d'isolation standards (ANSI ; JDBC ; JTA)  
(critiqués)
- Définis comme protection contre catégories de risques
- Risque défini comme : phénomène problématique
- SGBD configuré pour un niveau d'isolation donné
- Typiquement : Read committed
- Possible de se protéger contre certains risques au cas par cas

## Protection contre 2<sup>nd</sup> lost update

- Optimiste : lire version lors lecture, check version lors écriture
- Pessimiste : verrouiller lors lecture



# Licence

Cette présentation, et le code LaTeX associé, sont sous [licence MIT](#). Vous êtes libres de réutiliser des éléments de cette présentation, sous réserve de citer l'auteur.

Le travail réutilisé est à attribuer à [Olivier Cailloux](#), Université Paris-Dauphine.