

**Connolly et Begg - Chapitres 22 Gestion des transactions**

Marc Philippe Parent Database Systems 6e de Connolly et Begg

## Base de données

**IFT-2004**

**Département d’informatique et de génie logiciel**

## Chapitre 22 - Objectifs

* But du contrôle de simultanéité (concurrence)
* But de la restauration de BD
* Rôle et importance des transactions
* Propriétés d’une transaction
* Signification de la sérialisation et application au contrôle de simultanéité
* Utiliser des verrous pour garantir la sérialisation
* Comment fonctionne le protocole de verrou en deux phases (V2P)
* Signification des verrous indéfinis et comment les résoudre
* Utiliser l’estampillage pour garantir la sérialisation
* Fonctionnement des techniques optimistes de contrôle de simultanéité
* Certaines causes de défaillances des BD
* But du fichier journal de transactions

Connolly et Begg Database systems 6th 2

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Chapitre 22 - Objectifs

* + Rôle des points de contrôle pendant la journalisation des transactions
  + Récupérer un système à la suite d’une défaillance
  + Des modèles alternatifs pour les transactions de longue durée.

Connolly et Begg Database systems 6th 3

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Soutien des transactions

Une Transaction

* + - C’est une ou plusieurs Action(s)

demandée par un seul utilisateur ou programme d’application,

qui lit ou MAJ la BD

C’est une unité logique de travail (C’est un bloc ça forme un tout)

C’est un programme d'application

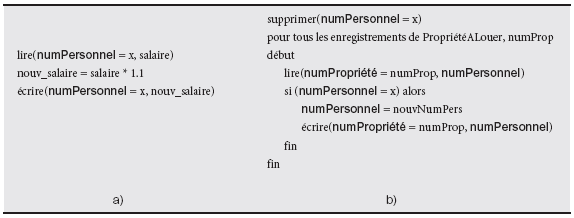
* + - Transactions avec traitement non BD entre plusieurs applications
  + Elle Transforme la BD d'un état cohérent à un autre état cohérent
    - Cette Cohérence peut être violée pendant transaction.

Connolly et Begg Database systems 6th 4

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Exemple Transaction

**(pseudo-code)**

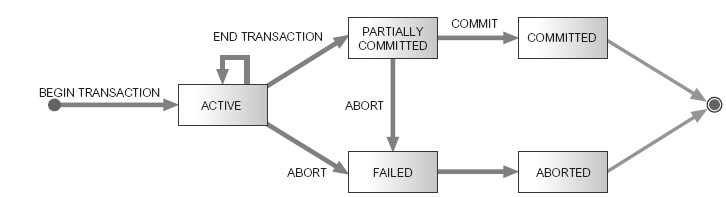


Connolly et Begg Database systems 6th 5

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Support aux transactions

## Partiellement valide / validé



**Echec Avorté (revenu a l’état initial)**

Connolly et Begg Database systems 6th 6

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Propriétés des transactions

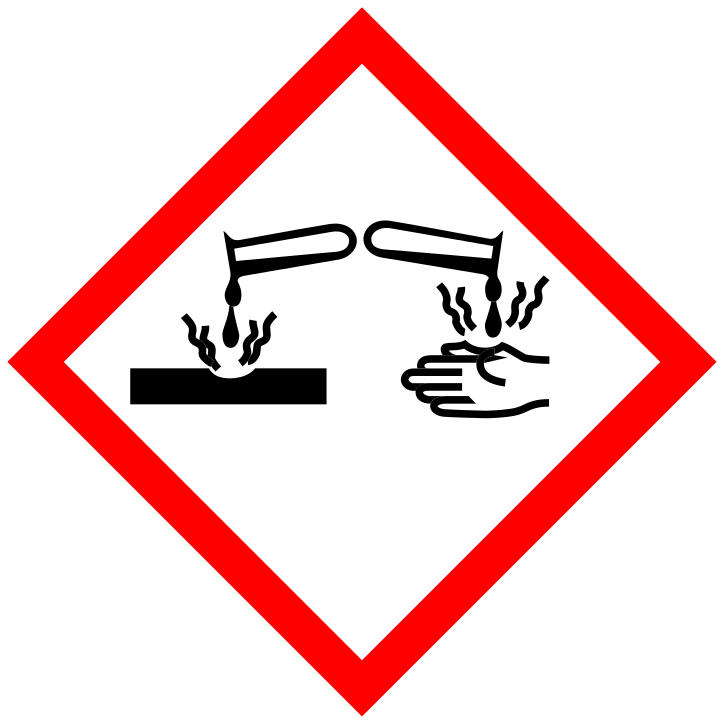
## (ACID) :

## Atomicité,

## Cohérence,

## Isolation (invisible),

## Durabilité (permanent)



Connolly et Begg Database systems 6th 7

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Le Contrôle de concurrence :

C’est la Gestion des opérations simultanées de la BD sans interférences

Elle prévient les interférences

Elle s’assure que les Transactions soient correctes

* Parce que l’Interaction peut produire un résultat incorrect

3 exemples problèmes concurrence.

Mise-a-jour Perdu

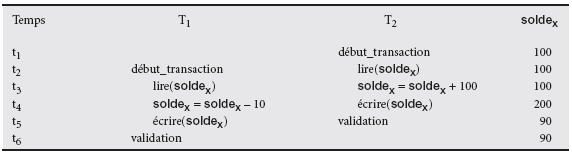
Problème de dépendance non validée

Problème d’analyse non-cohérente

Connolly et Begg Database systems 6th 8

Département d’informatique et de génie logiciel – Marc Philippe Parent

## MAJ perdue



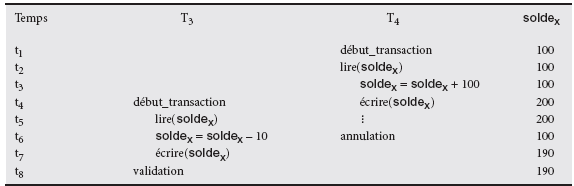
* + - Perte de la MAJ de T2 évitable qu’en interdisant T1

de lire le solde, tant que MAJ T2 pas achevée.

Connolly et Begg Database systems 6th 9

Département d’informatique et de génie logiciel – Marc Philippe Parent

# Problème de dépendance non validée



* Problème évité en empêchant T3 lire solde avant

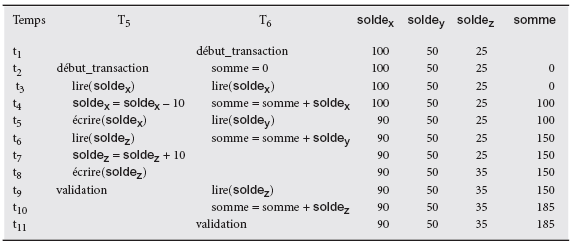
que T4

approuve ou annule.

Connolly et Begg Database systems 6th 10

Département d’informatique et de génie logiciel – Marc Philippe Parent

# Problème de l'analyse incohérente



* + Problème évité en empêchant T6 lire soldes de x

et z avant que T5 ait complété la MAJ.

Connolly et Begg Database systems 6th 11

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Capacité de sérialisation

* + L’Objectif du protocole de contrôle de concurrence
    - Est de planifier les transactions sans interférence
  + D’Exécuter les transactions en serie
  + La Sérialisation
    - Identifie les exécutions de transactions en assurant la cohérence
  + Une Planification c’est une
    - Séquence de lectures/écritures de transactions concurrentes préservant l’ordre des opérations de chacune

La Planification sérielle consiste à mettre les transactions une après l’autre

La Planification non sérielle : c’est d’essayer de faire les requêtes toutes en même temps.

Connolly et Begg Database systems 6th 12

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Capacité de sérialisation

* + Actions lectures/écritures important

1. Si 2 transactions font l’action de lire:

Il n’y a pas de probleme de conccurence

1. Si 2 transactions lisent ou écrivent des données différentes,

Il n’y a pas de problème de concurrence

1. Si des transactions écrit et autre lit ou écrit la même donnée.

Il y a un problème de concurence.

C’est la que la serialization est importante.

Connolly et Begg Database systems 6th 13

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Techniques de contrôle de concurrence

* + 2 techniques de base:
    - Le Verrouillage ou "Locking"
    - Estampillage ou "Timestamping" :

il va regarder quel transaction a commencé en premier

* + - Approches conservatrices (ou pessimistes)

Approche pessimiste :Retarder les transactions pour éviter tout problèmes. approche pessismiste

Technique optimiste : Laisser les requete se faire et si il y a un problème annuler la transaction.

Connolly et Begg Database systems 6th 14

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Verrouillage (technique conservatrice)

Le verrous consiste à :

* + - * Refuser les accès aux autres transactions
      * Pour Prévenir les MAJ incorrectes
    - C’est L’Approche la plus suivie dans les SGBD
    - La Transaction demande le verrou avant l’opération
      * Le Verrou partagé (lecture) empeche l’écriture
      * Le Verrou Exclusif (écriture) empeche la lecture et l’écriture.

Connolly et Begg Database systems 6th 15

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Verrouillage – Règles de base

* + Si la transaction possède unverrou partagé
    - Elle Peut lire mais pas modifier
  + Si une transaction possède un verrou exclusif
    - Elle Peut lire et modifier

Deux lectures ne sont jamais en conflit

* + - Plusieurs transactions peut avoir :
      * un verrous partagés sur la même donnée
  + Certains systèmes peuvent:

Élever le verrou de lecture en verrou exclusif

* + - Réduire le verrou exclusif en verrou partagé.

Connolly et Begg Database systems 6th 16

Département d’informatique et de génie logiciel – Marc Philippe Parent

**Exemple 22.5 – planification de verrouillage incorrect**

Planification valide, mais incorrecte

P = {verrou\_ecriture(T9, soldex), lecture(T9, soldex), écriture(T9, soldex), déverrouillage(T9, soldex),

verrou\_ecriture(T10, soldex), lecture(T10, soldex), écriture(T10, soldex), déverrouillage(T10, soldex),

verrou\_ecriture(T10, soldey), lecture(T10, soldey), écriture(T10, soldey), déverrouillage(T10, soldey),

validation(T10),

verrou\_ecriture(T9, soldey), lecture(T9, soldey), écriture(T9, soldey), déverrouillage(T9, soldey),

validation(T9) }.

Connolly et Begg Database systems 6th 17

Département d’informatique et de génie logiciel – Marc Philippe Parent

**Exemple 22.5 – planification de verrouillage incorrect**

* + Début

soldex = 100, soldey = 400

* + Résultat

soldex = 220, soldey = 330, si T9 avant

T10

soldex = 210, soldey = 340, si T10 avant T9

* + Mais donne soldex = 220 et soldey = 340

P pas planification sérialisable .

Connolly et Begg Database systems 6th 18

Département d’informatique et de génie logiciel – Marc Philippe Parent

**Exemple 22.5 – planification de verrouillage incorrect**

* + Problème
    - La Planification libère verrous trop tôt
    - Cela cause une perte d’isolation et d'atomicité
  + Solution: Verrouillage en 2 Phase (2PL)

Connolly et Begg Database systems 6th 19

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Verrouillage en 2 phases (2PL)

* + Protocole 2PL (V2P)
    - Toutes opérations de verrouillage précèdent la première opération de déverrouillage
  + 2 phases

La Croissance:

* + - * Acquière tous les verrous n'en libère aucun

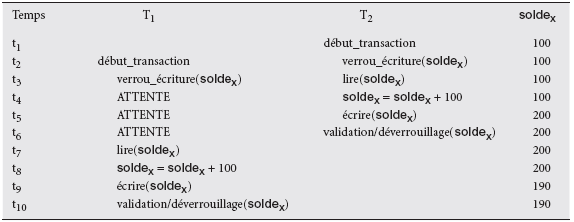
La Résorption :

* + - * Relâche verrous, ne peut en obtenir.

Connolly et Begg Database systems 6th 20

Département d’informatique et de génie logiciel – Marc Philippe Parent

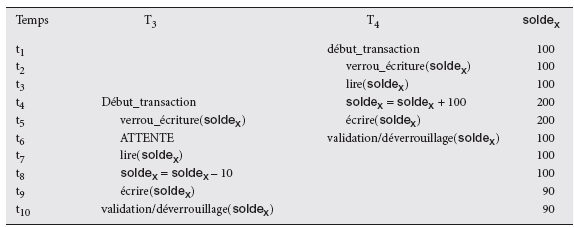
* 1. **- Éviter le problème de mise à jour perdue à l'aide de 2PL**



Connolly et Begg Database systems 6th 21

Département d’informatique et de génie logiciel – Marc Philippe Parent

* 1. **- Éviter le problème de dépendance non validée avec 2PL**

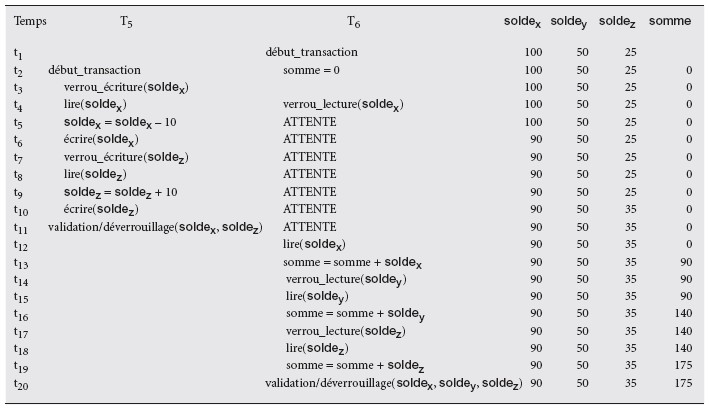


annulation

Connolly et Begg Database systems 6th 22

Département d’informatique et de génie logiciel – Marc Philippe Parent

* 1. **- Éviter le problème d'analyse incohérente avec 2PL**



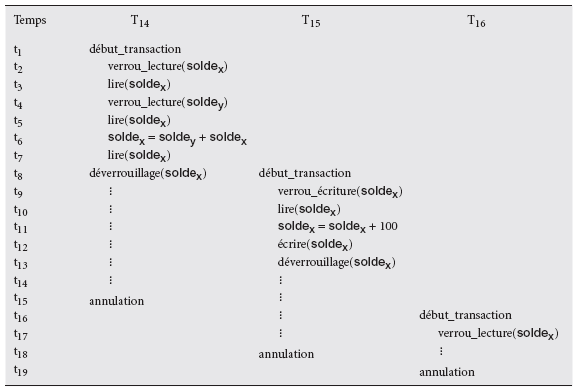
Connolly et Begg Database systems 6th 23

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Annulation en cascade

Transactions conformes 2PL

* + T14 -> échouée -> annulée



* + T15 dépend de T14

T15 aussi annulée

* + T16 dépend de T15
    - aussi annulée

Annulation en cascade

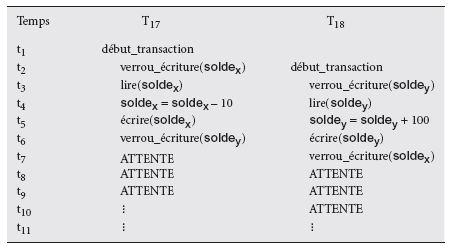
* + Solution avec 2PL
    - Aucun relâchements

avant fin transaction.

Connolly et Begg Database systems 6th 24

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Blocage indéfini (Deadlock)



Connolly et Begg Database systems 6th 25

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Blocage indéfini

* + Seul moyen de briser
    - 1+ transactions
  + Doit être transparent à l'utilisateur
    - SGBD doit transaction(s)
  + 3 techniques générales .

Connolly et Begg Database systems 6th 26

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Délai imparti

* + Transaction demande un verrou
    - Attendra que pour un temps
  + Si pas
    - Demande annulée
  + SGBD suppose transaction en blocage indéfini
    - Annule transaction
    - La

automatiquement.

Connolly et Begg Database systems 6th 27

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Prévention des blocages indéfinis

* + SGBD regarde en l’empêche

si blocage indéfini sera provoqué et

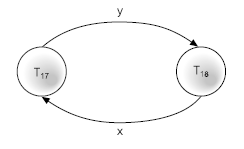
* + Ordonner transactions avec
    - Attendre -
      * Ancienne transaction attendre plus récente
    - - Attend
      * Transaction plus récente attend ancienne.

Connolly et Begg Database systems 6th 28

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Détection des blocages indéfinis

* + SGBD permet blocage indéfinis
    - Reconnaît et brise
  + Géré par graphe des
    - Créer nœud pour chaque
    - Créer Ti -> Tj, si Ti attend un verrou de Tj
  + Blocage indéfini ssi
  + Graphe créé à réguliers.



Connolly et Begg Database systems 6th 29

Département d’informatique et de génie logiciel – Marc Philippe Parent

**Récupération à la suite de la détection d'un blocage indéfini**

* + Plusieurs solutions
    - Choix de la

par moindre coût

* + - d’annulation de transaction
    - Évite la .

Connolly et Begg Database systems 6th 30

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Exercices

* + Les 4 propriétés des transactions?

A-Atrocité, Calculabilité, Indépendance, Développement B-Atomicité, Cohérence, Isolation, Développement

C-Atrocité, Calculabilité, Isolation, Durabilité D-Atomicité, Cohérence, Isolation, Durabilité

E-Atomicité, Calculabilité, Indépendance, Développement

* + Pour résoudre les blocages indéfinis, le serveur peut

A-Utiliser le délai imparti

B-Les prévenir en regardant en avant

C-Les détecter à l'aide d'un graphe des dépendances D-Aucune de ces réponses

E-Toutes ces réponses

Connolly et Begg Database systems 6th 31

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Méthodes d'estampillage

* + Transactions ordonnées globalement
  + Conflit  annuler et redémarrer transaction récente
  + Pas  pas verrou indéfini
  + Estampille - timestamp
    - Identificateur unique créé par le SGBD
    - Indique heure relative transaction
      * système ou
      * Compteur logique.

Connolly et Begg Database systems 6th 32

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Estampillage

* + Lecture/écriture autorisée seulement si
    - Dernière MAJ apportée par transaction plus
  + Sinon
    - Transaction qui demande
  + Estampilles de donnée

avec nouvelle estampille

* + - Estampille- Dernière transaction ayant lu
    - Estampille-

Dernière transaction ayant écrit.

Connolly et Begg Database systems 6th 33

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Demande lecture(x)

* + Transaction T avec l'estampille e(T)
    - Si e(T) < estampille\_ecriture(x)
      * x déjà par transaction plus récente
      * Transaction annulée et redémarrée avec
    - Sinon lecture permise

estampille

* + - * estampille\_lecture(x) = max(e(T), estampille\_lecture(x)).

Connolly et Begg Database systems 6th 34

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Demande écriture(x)

* + Si e(T) < estampille\_lecture(x)
    - x déjà par transaction plus jeune
    - T annulée et redémarrée avec
  + Sinon si e(T) < estampille\_écriture (x)

estampille

* + - x déjà par transaction plus jeune
    - Écriture
  + Sinon
    - Opération

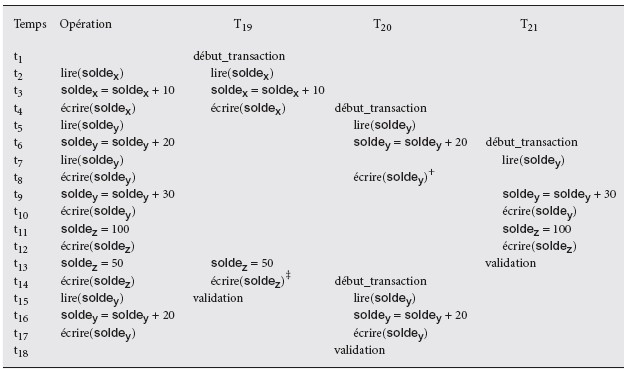
et exécutée

* + - estampille\_écriture(x) = e(T).

Connolly et Begg Database systems 6th 35

Département d’informatique et de génie logiciel – Marc Philippe Parent

**Exemple 22.10 – Classement élémentaire par ordre d'estampille**



Connolly et Begg Database systems 6th 36

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Tri des estampilles multiversions

* + Gestion données
    - Améliorer concurrence
  + Estampillage de base
    - Une de donnée
      * Une transaction peut y accéder
  + Plusieurs transactions lisent/écrivent différentes même donnée
  + Chaque transaction voit ensemble cohérent

de la

* + Écriture  Nouvelle de donnée
  + Lire  Sélectionne version = OK
  + Versions après.

Connolly et Begg Database systems 6th 37

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Techniques optimistes

* + Hypothèse
    - Conflit
    - Plus efficace laisser transactions s’exécuter sans délai
  + À la validation

    - Oui

pour déterminer si conflit

* + - Pas verrou
  + \_ phases.

annulée + redémarrée plus grande concurrence

Connolly et Begg Database systems 6th 38

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Techniques optimistes

* Phase 1 – Lecture
  + Début jusqu'à immédiatement

validation

* + - Transaction lit valeurs BD et stocke dans variables
  + Phase 2 – Validation
    - Transactions lecture  Vérifie données toujours valeurs
    - Transactions  Vérifier BD état cohérent
    - Pas d'interférence  Transaction
    - Sinon  Annulée + redémarrée
* Phase 3 – écritures - Transactions MAJ
  + Suit phase terminée avec succès
  + MAJ copies locales à la BD.

Connolly et Begg Database systems 6th 39

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Récupération d'une BD

* BD
* Besoin de contrôle de récupération
  + 3 types
    - Volatile (mémoire principale)
    - Non-volatile (disque dur)
    - Stockage hors .

Connolly et Begg Database systems 6th 40

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Types de pannes

* système
* Défaillances
* Erreurs
* naturelles
* Destruction données/équipements
* .

Connolly et Begg Database systems 6th 41

Département d’informatique et de génie logiciel – Marc Philippe Parent

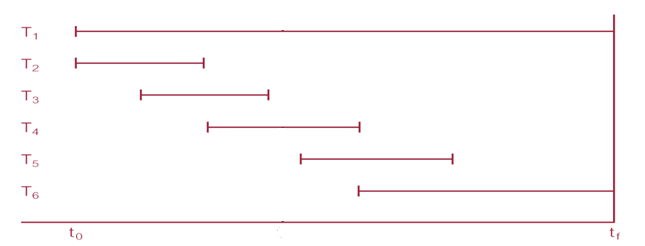
## Transactions et récupération

* Transactions
  + récupération
* Gestionnaire de récupération responsable
  + - Si transaction pas validée  Défaire effets
  + Durabilité
    - Si entre validation et écriture sur stockage secondaire
      * Refaire MAJ

Connolly et Begg Database systems 6th 42

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Transactions et récupération



Connolly et Begg Database systems 6th 43

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Gestion des tampons

* Gestionnaire tampons
  + Administre transferts pages mémoire  stockage
* Lire pages disque vers tampons jusqu’à
* Pas sur disque page déjà dans un tampon
* Stratégie de
* Stratégie
* Stratégie
* Plus simple en pratique.

Connolly et Begg Database systems 6th 44

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Utilitaires de récupérations

* SGBD fournit outils pour récupération
  + Mécanisme de
  + Points de
  + Gestionnaire de .

Connolly et Begg Database systems 6th 45

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Fichier journal

* Informations sur MAJ de BD

Aussi pour autres fonctions ( )

* Peut être double ou
* Parfois séparé en deux fichiers à accès
* Goulot d'étranglement potentiel.

Connolly et Begg Database systems 6th 46

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Fichier journal

* Enregistrements de transaction

de transaction

d'enregistrement

Image

Image

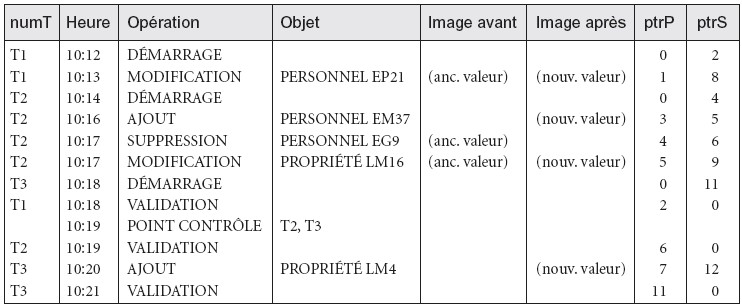
* + IInformation de

du journal.

Connolly et Begg Database systems 6th 47

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Segment de fichier journal



Connolly et Begg Database systems 6th 48

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Points de contrôle

* Point de

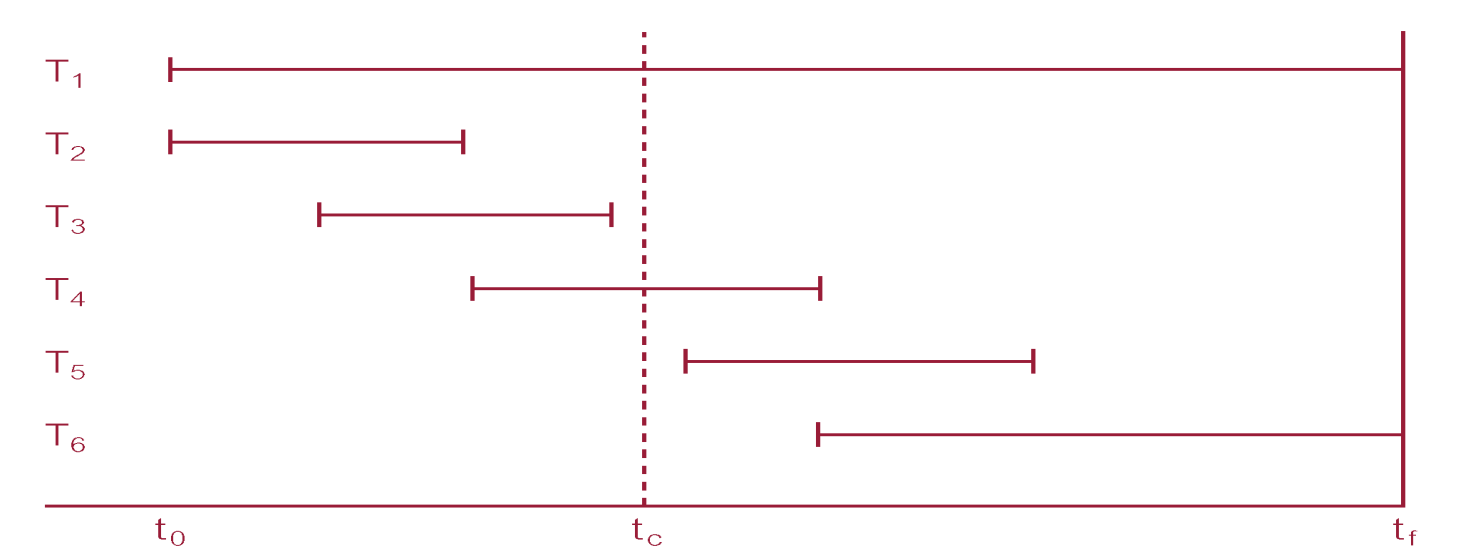
entre BD et journal

* Écriture forcée des tampons sur stockage secondaire
* Enregistrement point de contrôle créé
* Panne.

Connolly et Begg Database systems 6th 49

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Points de contrôle



Connolly et Begg Database systems 6th 50

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Techniques de récupération

* Si BD
  + Restaurer dernière
  + Refaire MAJ transactions validées avec journal
* Si DB
  + changements incohérence
  + Refaire transactions pour que MAJ soient sur stockage secondaire
  + Pas de
    - Restaurer BD avec images avant et après.

Connolly et Begg Database systems 6th 51

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Pagination à l'arrière-plan

* 2 tables de page pendant la transaction

* + Arrière-plan
* Début

2 pages

* Pendant
  + Arrière-plan
    - Jamais
    - Utilisée restauration BD si panne

Courante enregistre toutes

* Complétée 

devient Arrière-plan.

Connolly et Begg Database systems 6th 52

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Modèles avancés de transaction

Transactions de longue

* + Défaillance
    - Ne peut annuler/recommencer
  + Verrouille grand nombre données
    - Concurrence limitée si données

longtemps

* + - Verrou plus probable
  + Coopération utilisation données
* Nous voyons 2 modèles
  + Modèle de transactions imbriquées
  + Modèle de flux de travaux.

Connolly et Begg Database systems 6th 53

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Modèle de transactions imbriquées

Transaction



Haut niveau

de sous-transactions

Plusieurs transactions

* + Peut avoir transactions imbriquées

Niveau

* + Exécute opérations de BD
* Validées

Bas vers le .

Connolly et Begg Database systems 6th 54

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Modèle de transactions imbriquées

* Annulation à un niveau

Affecter pas transaction au niveau

* Parent peut récupérer

sous-transaction

pannes

* + - Sous-transaction non vitale
  + Exécuter sous-transaction

Tout

* MAJ sous-transactions niveaux intermédiaires visibles seulement

dans portée

immédiats.

Connolly et Begg Database systems 6th 55

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Modèle de transactions imbriquées

* Validation sous-transaction

* + ACID
* Avantages

à validation/annulation ses supérieurs

* + Décomposable en sous-transactions
    - Simultanéité et récupération

plus fine

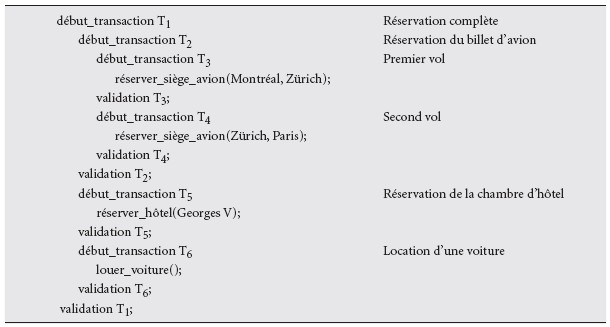
dans transaction

dans transaction

Connolly et Begg Database systems 6th 56

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Exemple de transactions imbriquées



Connolly et Begg Database systems 6th 57

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Modèles de flux d'activité

* Modèle précédent
  + Pas avec certaines activités industrielles et

commerciales

* Flux d'activité
  + Implique exécution de traitement

de tâches effectuées par entités

* + - Personnes ou systèmes logiciels
* 2 problèmes généraux
  + et exécution
* Compliqués par plusieurs organisations utilisent plusieurs

systèmes indépendants pour processus.

différentes parties du

Connolly et Begg Database systems 6th 58

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Exercices

* Dans l'estampillage multiversions, il existe

A- plusieurs versions des données

B- plusieurs versions des estampilles

C- plusieurs versions de page (avant-plan et arrière plan) D- plusieurs versions de verrous

E- Aucune de ces réponses

* Laquelle est fausse, dans le modèle imbriqué des transactions

A-Une sous-transaction est valide si supérieurs le sont B-les sous-transactions respectent les propriétés ACID

C-Il peut y avoir du parallèlisme entre les sous-transactions D-Aucune de ces réponses

E-A et C F-B et C G-A et B H-Toutes ces réponses

Connolly et Begg Database systems 6th 59

Département d’informatique et de génie logiciel – Marc Philippe Parent

## Références

* image ACID https://upload.wikimedia.org/wikipedia/commons/a/a1/GHS-pictogram-acid.svg

Connolly et Begg Database systems 6th 60

Département d’informatique et de génie logiciel – Marc Philippe Parent